FRA Milestone 2 Business Project Report

TEJAS PADEKAR
PGP-DSBA Online
Date: 11/12/2022

CONTENTS:

List of Figures

List of Tables

PROBLEM 1

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.

Problem Statement 1:

Accurately predict the defaulters using Statsmodel approach in Logistic Regression, Random Forest and Linear Discriminant Analysis and compare the three to determine the best model for this dataset. 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 is to be used to drive the label field.

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 the necessary label field 'default' using the projected company networth next year for 1 year in future ie 2016 as requested.

Data Dictionary:

	#	Field Name	Description	New Field Name
0	1	Co_Code	Company Code	Co_Code
1	2	Co_Name	Company Name	Co_Name
2	3	Networth Next Year	Value of a company as on 2016 - Next Year(difference between the value of total assets and total liabilities)	Networth_Next_Year
3	4	Equity Paid Up	Amount that has been received by the company through the issue of shares to the shareholders	Equity_Paid_Up
4	5	Networth	Value of a company as on 2015 - Current Year	Networth
5	6	Capital Employed	Total amount of capital used for the acquisition of profits by a company	Capital_Employed
6	7	Total Debt	The sum of money borrowed by the company and is due to be paid	Total_Debt
7	8	Gross Block	Total value of all of the assets that a company owns	Gross_Block
8	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).	Net_Working_Capital
9	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.	Curr_Assets
10	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)	Curr_Liab_and_Prov
11	12	Total Assets/Liabilities	Ratio of total assets to liabailities of the company	Total_Assets_to_Liab
12	13	Gross Sales	The grand total of sale transactions within the accounting period	Gross_Sales
13	14	Net Sales	Gross sales minus returns, allowances, and discounts	Net_Sales
14	15	Other Income	Income realized from non-business activities (e.g. sale of long term asset)	Other_Income
15	16	Value Of Output	Product of physical output of goods and services produced by company and its market price	Value_Of_Output
16	17	Cost of Production	Costs incurred by a business from manufacturing a product or providing a service	Cost_of_Prod
17	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)	Selling_Cost
18	19	PBIDT	Profit Before Interest, Depreciation & Taxes	PBIDT
19	20	PBDT	Profit Before Depreciation and Tax	PBDT
20	21	PBIT	Profit before interest and taxes	PBIT
21	22	PBT	Profit before tax	PBT
22	23	PAT	Profit After Tax	PAT
23	24	Adjusted PAT	Adjusted profit is the best estimate of the true profit	Adjusted_PAT
24	26	CP	Commercial paper , a short-term debt instrument to meet short-term liabilities.	CP
25	27	Revenue earnings in forex	Revenue earned in foreign currency	Rev_earn_in_forex
26	28	Revenue expenses in forex	Expenses due to foreign currency transactions	Rev_exp_in_forex
27	29	Capital expenses in forex	Long term investment in forex	Capital_exp_in_forex
28	30	Book Value (Unit Curr)	Net asset value	Book_Value_Unit_Curr
29	31	Book Value (Adj.) (Unit Curr)	Book value adjusted to reflect asset's true fair market value	Book_Value_Adj_Unit_Curr
30	32	Market Capitalisation	Product of the total number of a company's outstanding shares and the current market price of one share	Market_Capitalisation
31	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	CEPS_annualised_Unit_Curr
32	34	Cash Flow From Operating Activities	Use of cash from ongoing regular business activities	Cash_Flow_From_Opr
33	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	Cash_Flow_From_Inv
34	36	Cash Flow From Financing Activities	Net flows of cash that are used to fund the company (transactions involving debt, equity, and dividends)	Cash_Flow_From_Fin
35	37	ROG-Net Worth (%)	Rate of Growth - Networth	ROG_Net_Worth_perc
36	38	ROG-Capital Employed (%)	Rate of Growth - Capital Employed	ROG_Capital_Employed_perc
37	39	ROG-Gross Block (%)	Rate of Growth - Gross Block	ROG_Gross_Block_perc
38	40	ROG-Gross Sales (%)	Rate of Growth - Gross Sales	ROG_Gross_Sales_perc
39	41	ROG-Net Sales (%)	Rate of Growth - Net Sales	ROG_Net_Sales_perc
40	42	ROG-Cost of Production (%)	Rate of Growth - Cost of Production	ROG_Cost_of_Prod_perc
41	43	ROG-Total Assets (%)	Rate of Growth - Total Assets	ROG_Total_Assets_perc
42	44	ROG-PBIDT (%)	Rate of Growth- PBIDT	ROG_PBIDT_perc
43	45	ROG-PBDT (%)	Rate of Growth- PBDT	ROG_PBDT_perc

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

Data Description in brief.

Dataset Head (Snapshot of Top 5 rows):

	Co_Code	Co_Name	Networth Next Year	Equity Paid Up	Networth	Capital Employed	Total Debt	Gross Block	Net Working Capital	Current Assets	...	PBIDTM (%) [Latest]	PBITM (%) [Latest]	PBDTM (%) [Latest]	CPM (%) [Latest]	APATM (%) [Latest]	Debt Velo (Da
0	16974	Hind.Cables	-8021.60	419.36	-7027.48	-1007.24	5936.03	474.30	-1076.34	40.50	...	0.00	0.00	0.00	0.00	0.00	
1	21214	Tata Tele. Mah.	-3986.19	1954.93	-2968.08	4458.20	7410.18	9070.86	-1098.88	486.86	...	-10.30	-39.74	-57.74	-57.74	-87.18	
2	14852	ABG Shipyard	-3192.58	53.84	506.86	7714.68	6944.54	1281.54	4496.25	9097.64	...	-5279.14	-5516.98	-7780.25	-7723.67	-7961.51	
3	2439	GTL	-3054.51	157.30	-623.49	2353.88	2326.05	1033.69	-2612.42	1034.12	...	-3.33	-7.21	-48.13	-47.70	-51.58	
4	23505	Bharati Defence	-2967.36	50.30	-1070.83	4675.33	5740.90	1084.20	1836.23	4685.81	...	-295.55	-400.55	-845.88	379.79	274.79	3

Above is a small snippet of the dataset. There are a total of 3586 rows and 67 columns in the dataset. The dataset contains mixed information across columns ranging from ratios to figures to percentages of a company's financial record.

Variables of 'Co_Code' & 'Co_Name' will not add any significance to our model and hence, we will be dropping these variables.

Target field 'default' is to be created separately using the features for the variable 'Networth_Next_Year'. Once we create the 'default' field, we will also drop 'Networth_Next_Year' as it will create a bias in our model and pose a high weightage amongst the predictors.

Data Info:

```
29  Book_Value_Adj_Unit_Curr           3582 non-null   float64
51  Curr_Ratio_Latest                  3585 non-null   float64
52  Fixed_Assets_Ratio_Latest          3585 non-null   float64
53  Inventory_Ratio_Latest             3585 non-null   float64
54  Debtors_Ratio_Latest               3585 non-null   float64
55  Total_Asset_Turnover_Ratio_Latest  3585 non-null   float64
56  Interest_Cover_Ratio_Latest        3585 non-null   float64
57  PBIDTM_perc_Latest                 3585 non-null   float64
58  PBITM_perc_Latest                  3585 non-null   float64
59  PBDTM_perc_Latest                  3585 non-null   float64
60  CPM_perc_Latest                    3585 non-null   float64
61  APATM_perc_Latest                  3585 non-null   float64
64  Inventory_Vel_Days                 3483 non-null   float64
dtypes: float64(63), int64(3), object(1)
```

There are 66 continuous and 1 categorical variable. 63 variables are float type, 3 are integers and 1 is an object. There are null values present in a few variables which are mentioned above. Let us check these missing values data.

Missing Value Count:

```
118
```

```
df.size
```

```
240262
```

```
(118/240262)*100
```

```
0.04911305158535265
```

```
Inventory_Vel_Days                 103
Book_Value_Adj_Unit_Curr             4
Inventory_Ratio_Latest               1
Interest_Cover_Ratio_Latest          1
Curr_Ratio_Latest                    1
Fixed_Assets_Ratio_Latest            1
Debtors_Ratio_Latest                 1
Total_Asset_Turnover_Ratio_Latest    1
PBIDTM_perc_Latest                   1
PBITM_perc_Latest                    1
PBDTM_perc_Latest                    1
CPM_perc_Latest                      1
APATM_perc_Latest                    1
```

There are in total 118 missing points of the total 240262 features in our dataset which is very insignificant at 0.05% . Also, there are no duplicate records in the dataset. Hence, the data can be said to be relatively good and we shall impute the missing data appropriately as part of our data preprocessing.

Treating missing values appropriately is an important exercise for regression and hence, these would need to be imputed appropriately. We will not be dropping them as there are not large in numbers compared to the overall data and the other features in the same rows could be valuable to our analysis and we must avoid losing them.

Dataset Description (Continuous Variables):

	count	mean	std	min	25%	50%	75%	max
Co_Code	3586.00	16065.39	19776.82	4.00	3029.25	6077.50	24269.50	72493.00
Networth_Next_Year	3586.00	725.05	4769.68	-8021.60	3.98	19.02	123.80	111729.10
Equity_Paid_Up	3586.00	62.97	778.76	0.00	3.75	8.29	19.52	42263.46
Networth	3586.00	649.75	4091.99	-7027.48	3.89	18.58	117.30	81657.35
Capital_Employed	3586.00	2799.61	26975.14	-1824.75	7.60	39.09	226.61	714001.25
Total_Debt	3586.00	1994.82	23652.84	-0.72	0.03	7.49	72.35	652823.81
Gross_Block	3586.00	594.18	4871.55	-41.19	0.57	15.87	131.90	128477.59

Net_Working_Capital	3586.00	410.81	6301.22	-13162.42	0.94	10.14	61.17	223257.56
Curr_Assets	3586.00	1960.35	22577.57	-0.91	4.00	24.54	135.28	721166.00
Curr_Liab_and_Prov	3586.00	391.99	2675.00	-0.23	0.73	9.23	65.65	83232.98
Total_Assets_to_Liab	3586.00	1778.45	11437.57	-4.51	10.55	52.01	310.54	254737.22
Gross_Sales	3586.00	1123.74	10603.70	-62.59	1.44	31.21	242.25	474182.94
Net_Sales	3586.00	1079.70	9996.57	-62.59	1.44	30.44	234.44	443775.16
Other_Income	3586.00	48.73	426.04	-448.72	0.02	0.45	3.63	14143.40
Value_Of_Output	3586.00	1077.19	9843.88	-119.10	1.41	30.89	235.84	435559.09
Cost_of_Prod	3586.00	798.54	9076.70	-22.65	0.94	25.99	189.55	419913.50
Selling_Cost	3586.00	25.55	194.24	0.00	0.00	0.16	3.88	5283.91
PBIDT	3586.00	248.18	1949.59	-4655.14	0.04	2.04	23.52	42059.26
PBDT	3586.00	116.27	956.20	-5874.53	0.00	0.80	12.95	23215.00
PBIT	3586.00	217.66	1850.97	-4812.95	0.00	1.15	16.67	41402.96
PBT	3586.00	85.75	799.93	-6032.34	-0.06	0.31	7.42	16798.00
PAT	3586.00	61.22	620.30	-6032.34	-0.06	0.26	5.54	13383.39
Adjusted_PAT	3586.00	60.06	580.43	-4418.72	-0.09	0.21	5.34	13384.11
CP	3586.00	91.73	780.79	-5874.53	0.00	0.74	10.91	20760.20
Rev_earn_in_forex	3586.00	131.17	1150.73	0.00	0.00	0.00	7.20	46158.00
Rev_exp_in_forex	3586.00	256.33	4132.34	0.00	0.00	0.00	6.99	193979.73
Capital_exp_in_forex	3586.00	7.66	111.43	0.00	0.00	0.00	0.00	3722.10
Book_Value_Unit_Curr	3586.00	157.24	1622.66	-3371.57	7.96	21.66	71.67	75790.00
Book_Value_Adj_Unit_Curr	3582.00	2243.15	128283.73	-33715.70	7.06	18.93	60.01	7677600.29
Market_Capitalisation	3586.00	1664.09	12805.17	0.00	0.00	8.37	111.46	260865.08
CEPS_annualised_Unit_Curr	3586.00	36.02	828.42	-1808.00	0.00	1.15	8.77	45438.44
Cash_Flow_From_Opr	3586.00	65.77	1455.05	-25469.23	-0.31	0.45	12.65	44529.40
Cash_Flow_From_Inv	3586.00	-60.87	701.97	-23843.45	-5.12	-0.12	0.12	3732.98
Cash_Flow_From_Fin	3586.00	11.44	1272.26	-38374.04	-5.85	0.00	0.46	28846.00
ROG_Net_Worth_perc	3586.00	1237.62	41041.93	-14485.71	-1.49	1.84	11.36	2144020.00
ROG_Capital_Employed_perc	3586.00	2988.88	126472.87	-8614.63	-3.83	1.38	12.59	7412700.00
ROG_Gross_Block_perc	3586.00	37.55	893.62	-116.12	0.00	0.25	6.72	47400.00
ROG_Gross_Sales_perc	3586.00	242.67	6103.53	-5503.70	-8.08	3.31	21.53	320200.00
ROG_Net_Sales_perc	3586.00	242.59	6103.49	-5503.70	-8.12	3.21	21.57	320200.00
ROG_Cost_of_Prod_perc	3586.00	310.49	5573.22	-2130.23	-7.24	4.42	23.12	267150.00

ROG_PBIDT_perc	3586.00	375.85	23278.40	-52200.00	-23.36	4.57	47.88	1386200.00
ROG_PBDT_perc	3586.00	336.38	20353.40	-52200.00	-30.60	3.37	52.91	1208700.00
ROG_PBIT_perc	3586.00	374.70	22462.79	-58500.00	-31.35	2.13	50.14	1338000.00
ROG_PBT_perc	3586.00	224.07	19659.23	-78900.00	-41.23	0.03	61.96	1160500.00
ROG_PAT_perc	3586.00	112.23	13480.52	-114500.00	-43.73	0.00	65.35	774200.00
ROG_CP_perc	3586.00	221.09	13980.20	-52200.00	-29.50	4.62	52.91	822400.00
ROG_Rev_earn_in_forex_perc	3586.00	37.23	658.67	-100.00	0.00	0.00	0.00	29084.77
ROG_Rev_exp_in_forex_perc	3586.00	364.86	15233.64	-100.00	0.00	0.00	0.00	894591.69
ROG_Market_Capitalisation_perc	3586.00	63.68	1047.93	-98.05	0.00	0.00	47.52	61865.26
Curr_Ratio_Latest	3585.00	12.06	108.41	0.00	0.88	1.36	2.77	4813.00
Fixed_Assets_Ratio_Latest	3585.00	51.54	681.15	0.00	0.27	1.56	4.74	22172.00
Inventory_Ratio_Latest	3585.00	37.80	458.19	0.00	0.00	3.56	8.94	15472.00
Debtors_Ratio_Latest	3585.00	33.03	489.56	0.00	0.42	3.82	8.52	22992.67
Total_Asset_Turnover_Ratio_Latest	3585.00	1.24	2.67	0.00	0.07	0.60	1.55	57.75
Interest_Cover_Ratio_Latest	3585.00	16.39	351.74	-5450.00	0.00	1.08	3.71	18639.40
PBIDTM_perc_Latest	3585.00	-51.16	1795.13	-78870.45	0.00	8.07	18.99	19233.33
PBITM_perc_Latest	3585.00	-109.21	3057.64	-141600.00	0.00	5.23	14.29	19195.70
PBDTM_perc_Latest	3585.00	-311.57	10921.59	-590500.00	0.00	4.69	14.11	15640.00
CPM_perc_Latest	3585.00	-307.01	10676.15	-572000.00	0.00	3.89	11.39	15640.00
APATM_perc_Latest	3585.00	-365.06	12500.05	-688600.00	0.00	1.59	7.41	15266.67
Debtors_Vel_Days	3586.00	603.89	10636.76	0.00	8.00	49.00	106.00	514721.00
Creditors_Vel_Days	3586.00	2057.85	54169.48	0.00	8.00	39.00	89.00	2034145.00
Inventory_Vel_Days	3483.00	79.64	137.85	-199.00	0.00	35.00	96.00	996.00
Value_of_Output_to_Total_Assets	3586.00	0.82	1.20	-0.33	0.07	0.48	1.16	17.63
Value_of_Output_to_Gross_Block	3586.00	61.88	976.82	-61.00	0.27	1.53	4.91	43404.00

We have been provided with a very comprehensive list of very important financial ratios and calculations. However, the data has good amount of outliers and zero or close to zero values. Zero values could also be present as the particular ratio may not be relevant to the specific business or industry.

The mean and median figures for almost all variables have a huge difference. The std deviation for a majority of the financial records is high. By the range of Market Capitalization, we can see that the portfolio of clientele is diverse catering to large as well as mid and small businesses.

Outlier Treatment:

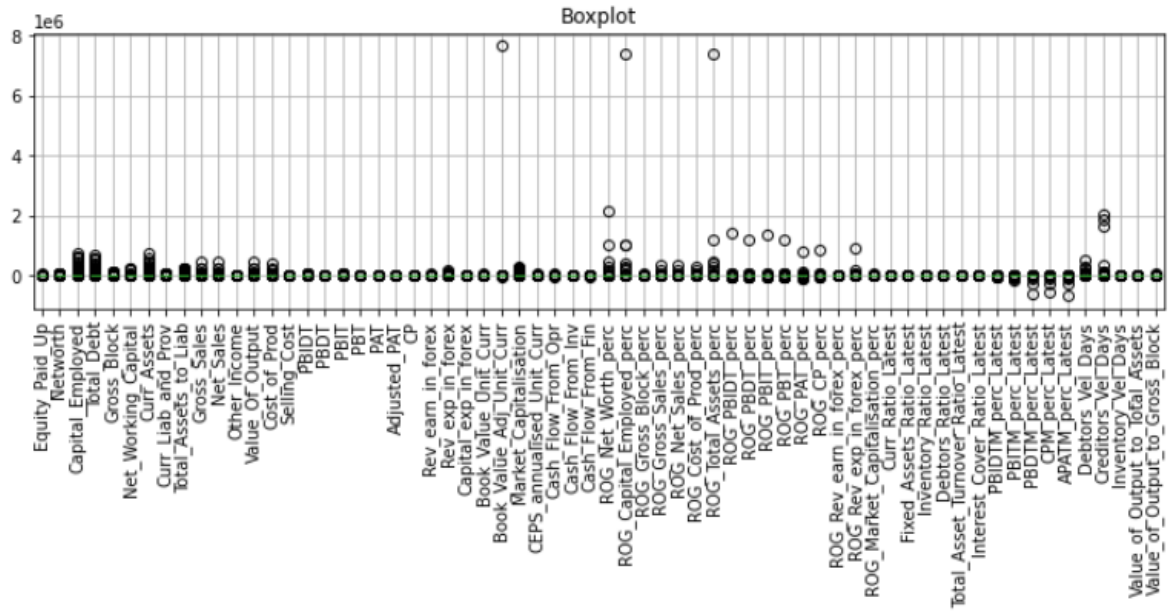

Figure 1 – Outliers Pre Treatment

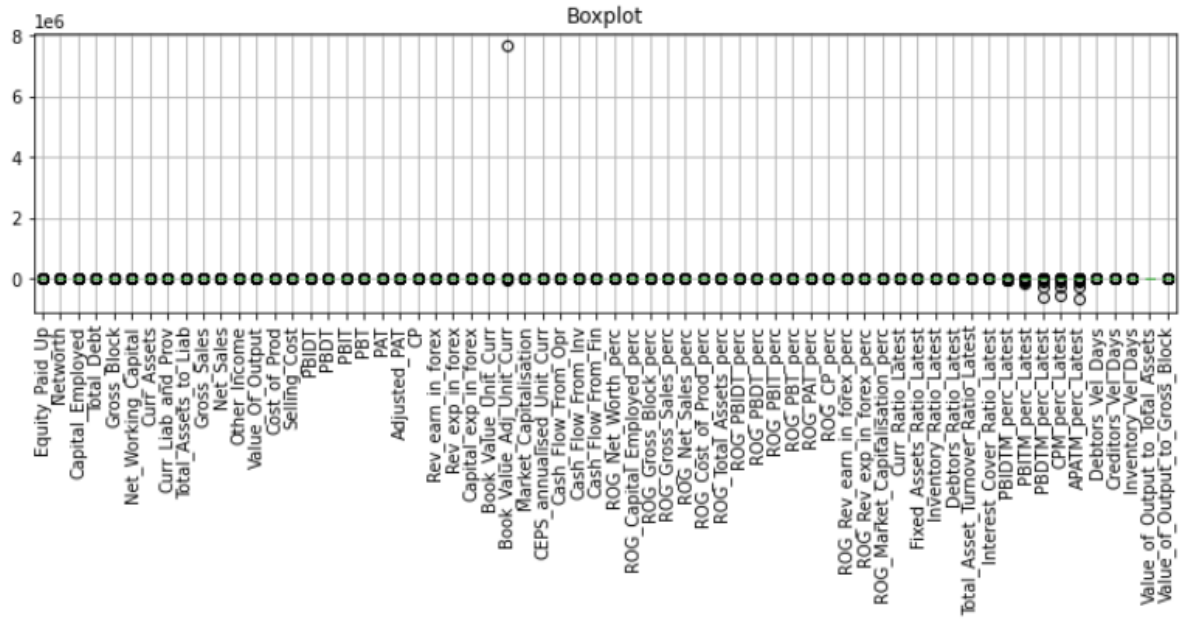

Figure 2 – Outliers Post Treatment

We used the capping and treating technique to treat the outliers where any outlier < Lower Limit (q25-(1.5*IQR) is capped at 5 percentile and any outlier > Upper Limit (q75+(1.5*IQR) per is capped at 95 percentile. In the above image, we can see the outliers have been eliminated as per our capping command.

Missing Value Treatment:

Imputing Missing Values:

```
default                        0
Cash_Flow_From_Fin             0
Cash_Flow_From_Opr             0
CEPS_annualised_Unit_Curr      0
Market_Capitalisation          0
                              ..
ROG_Cost_of_Prod_perc          0
ROG_Net_Sales_perc             0
ROG_Gross_Sales_perc           0
ROG_Gross_Block_perc           0
Equity_Paid_Up                 0
Length: 65, dtype: int64
```

We had 118 missing values as per mentioned earlier and have imputed these missing Nan values with KNN imputer having 5 nearest neighbours as parameter for imputation. We chose this method as it is considered as a rather basic and more effective approach than using imputation through mean or median values.

Transform Target variable into 0 and 1

Target Variable Description:

```
0    3198
1     388
Name: default, dtype: int64

0   0.89
1   0.11
Name: default, dtype: float64
```

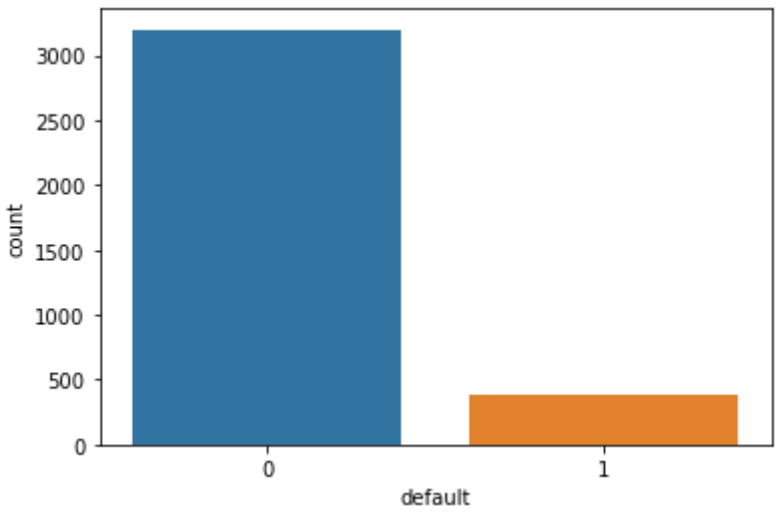

Figure 3: Proportion of Defaulters and Non Defaulters

We have 388 defaulters and 3198 non defaulters in the dataset which means around 11% of the total 3586 clients are defaulters.

Elimination of Variables with VIF >5:

As multicollinearity needs to be dealt with before applying Logistic Regression as otherwise it effects the significance of variables used in the prediction, we would be eliminating variables with high vif ie; Vif >5 using backward elimination approach.

Post dropping variables with VIF >5, we noticed that of 64 total continuous variables, 21 variables had VIF >= 5, which we can say that 1/3rd of the variables or financial parameters had multicollinearity present. Below is the list of balance 43 variables.

Variables with VIF <5 (in descending order)

	variables	VIF
2	Gross_Block	4.65
4	Curr_Liab_and_Prov	4.39
15	Cash_Flow_From_Opr	3.96
14	CEPS_annualised_Unit_Curr	3.76
19	ROG_Capital_Employed_perc	3.59
1	Total_Debt	3.37
23	ROG_Total_Assets_perc	3.35
24	ROG_PBIT_perc	3.24
11	Book_Value_Unit_Curr	3.23
25	ROG_PBT_perc	3.19
7	Adjusted_PAT	3.17

26	ROG_CP_perc	3.14
13	Market_Capitalisation	3.05
41	Value_of_Output_to_Total_Assets	2.98
9	Rev_exp_in_forex	2.92
3	Net_Working_Capital	2.79
16	Cash_Flow_From_Inv	2.76
6	Selling_Cost	2.70
5	Other_Income	2.64
10	Capital_exp_in_forex	2.57
8	Rev_earn_in_forex	2.45
0	Equity_Paid_Up	2.36
17	Cash_Flow_From_Fin	2.35
18	ROG_Net_Worth_perc	2.09
21	ROG_Net_Sales_perc	1.96
34	Total_Asset_Turnover_Ratio_Latest	1.89
22	ROG_Cost_of_Prod_perc	1.87
42	Value_of_Output_to_Gross_Block	1.58
38	Debtors_Vel_Days	1.51
39	Creditors_Vel_Days	1.50
29	ROG_Market_Capitalisation_perc	1.47
37	CPM_perc_Latest	1.34
36	PBIDTM_perc_Latest	1.33
20	ROG_Gross_Block_perc	1.30
40	Inventory_Vel_Days	1.29
28	ROG_Rev_exp_in_forex_perc	1.17
31	Fixed_Assets_Ratio_Latest	1.10
27	ROG_Rev_earn_in_forex_perc	1.10
35	Interest_Cover_Ratio_Latest	1.04
32	Inventory_Ratio_Latest	1.03
30	Curr_Ratio_Latest	1.02
33	Debtors_Ratio_Latest	1.02
12	Book_Value_Adj_Unit_Curr	1.01

Figure 4 – Variables with VIF <5

Train Test Split:

We have split the data into Train and Test dataset in a ratio of 67:33 and used random state =42 as informed. However, we have also used 'Statify' to maintain the same proportion of target variable 'default' in both Train as well as Test datasets.

Train Dataset (Post Split)

```
(2402, 65)
```

Proportion of Default Variable in Train Dataset (Post Split)

```
0.00    2142
1.00     260
```

Test Dataset (Post Split)

```
(1184, 65)
```

Proportion of Default Variable in Test Dataset (Post Split)

```
0.00    1056
1.00     128
```

We can see that the split has divided the Train and Test dataset appropriately in 67:33 and also the proportion of target variable 'default' similarly.

P Value Significance and Logistic Regression Result:

We will now check the significance of the balance variables based on the p value to determine the final set of predictors. We start by fitting these variables in the logistic regression equation and basis on the P value we will further narrow down on the significant predictors.

Significant variables are those with P value < 0.05 (alpha). Hence, we will further eliminate variables with P value > 0.05 using backward elimination approach.

This being an iterative process, it had to be ran 31 times until we were left with 14 variables having P value < 0.05 and which form our Model 31 given below.

Logit Regression Results

Dep. Variable:	default	**No. Observations:**	2402
Model:	Logit	**Df Residuals:**	2387
Method:	MLE	**Df Model:**	14
Date:	Sun, 04 Dec 2022	**Pseudo R-squ.:**	0.6089
Time:	18:05:07	**Log-Likelihood:**	-322.07
converged:	True	**LL-Null:**	-823.47
Covariance Type:	nonrobust	**LLR p-value:**	3.913e-205

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-0.9924	0.136	-7.280	0.000	-1.260	-0.725
Total_Debt	0.0009	0.000	2.117	0.034	6.46e-05	0.002
Curr_Liab_and_Prov	0.0021	0.001	2.510	0.012	0.000	0.004
Selling_Cost	-0.0224	0.010	-2.179	0.029	-0.043	-0.002
Rev_exp_in_forex	0.0033	0.002	2.016	0.044	9.34e-05	0.007
Book_Value_Unit_Curr	-0.1521	0.011	-13.241	0.000	-0.175	-0.130
Market_Capitalisation	-0.0008	0.000	-3.358	0.001	-0.001	-0.000
CEPS_annualised_Unit_Curr	-0.0911	0.035	-2.613	0.009	-0.159	-0.023
ROG_Net_Worth_perc	-0.0139	0.004	-3.398	0.001	-0.022	-0.006
ROG_Capital_Employed_perc	0.0145	0.006	2.419	0.016	0.003	0.026
ROG_Net_Sales_perc	-0.0031	0.001	-2.157	0.031	-0.006	-0.000
ROG_Total_Assets_perc	-0.0146	0.007	-2.028	0.043	-0.029	-0.000
Interest_Cover_Ratio_Latest	-0.0019	0.001	-2.164	0.030	-0.004	-0.000
Debtors_Vel_Days	-0.0011	0.000	-2.356	0.018	-0.002	-0.000
Value_of_Output_to_Gross_Block	-0.0221	0.009	-2.419	0.016	-0.040	-0.004

Figure 5 – Logistic Regression Result for Model 31

We can see that all the above variables are having p value < 0.05 and hence, these 14 variables can be determined as significant predictors & may be useful to discriminate cases of default.

We see that adjusted pseudo R square of 0.591 is now close to pseudo R square of 0.608, thus suggesting presence of lesser insignificant variables in the model no 31 as well.

We now check the VIF for these variables below.

	variables	VIF
1	Curr_Liab_and_Prov	3.36
8	ROG_Capital_Employed_perc	3.35
10	ROG_Total_Assets_perc	3.20
6	CEPS_annualised_Unit_Curr	3.19
4	Book_Value_Unit_Curr	3.07
0	Total_Debt	2.51
2	Selling_Cost	2.34
5	Market_Capitalisation	2.19
3	Rev_exp_in_forex	2.16
7	ROG_Net_Worth_perc	1.90
13	Value_of_Output_to_Gross_Block	1.26
9	ROG_Net_Sales_perc	1.21
12	Debtors_Vel_Days	1.10
11	Interest_Cover_Ratio_Latest	1.03

The VIF is high for a few first variables but they are not > 5. Let us now plot the default variable on these variables.

Plotting Default Variable on Predicted Probability Values from Model 31:

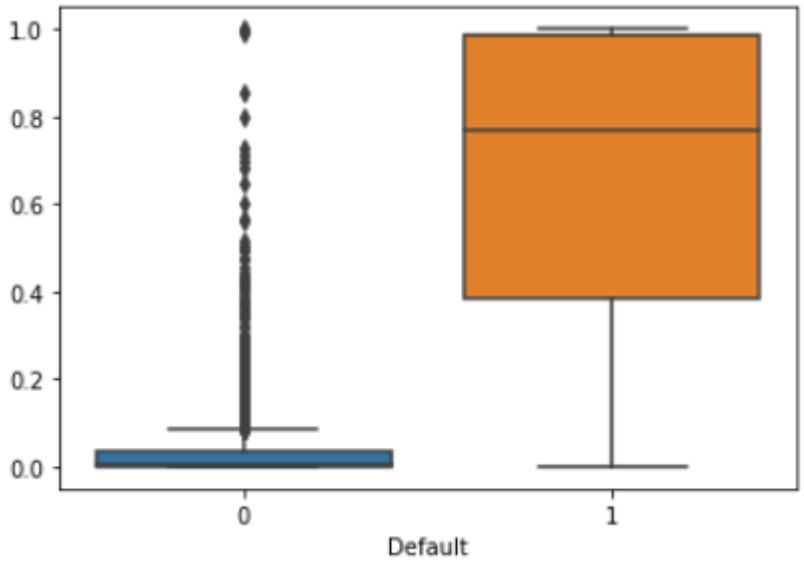

Figure 6 – Default Variable on Predicted Probability Values

We can see that the model is able to differentiate between the probability of defaulters and non – defaulters.

Building Logistic Regression Model (using statsmodel library) on most important variables on Train Dataset

Logistic Regression (LR) Model Building on Model 31

LR Model Building using 0.09 Threshold: Our Best Model from Milestone 1 Project

- **Confusion matrix for LR training set of 0.09 threshold**

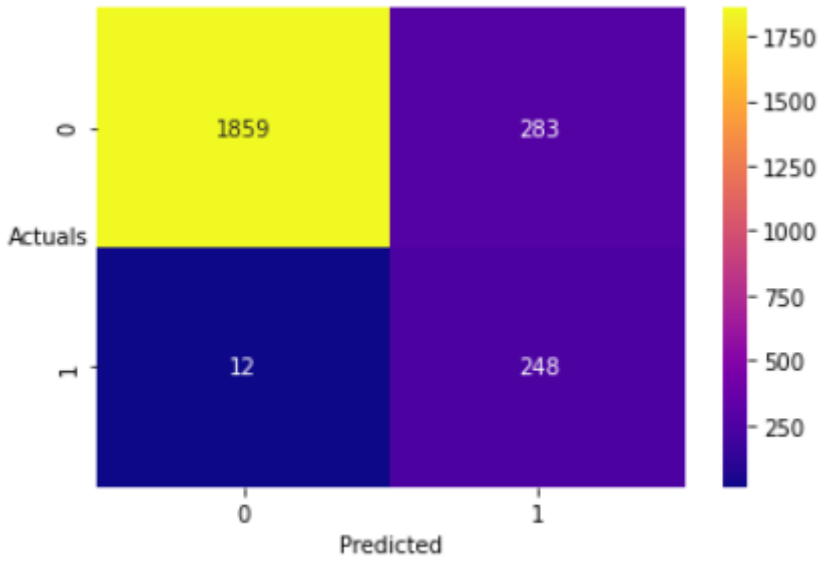

Figure 7 – Confusion matrix for LR training set of 0.09 threshold

- **Classification Report for LR training set of 0.09 threshold**

	precision	recall	f1-score	support
0.0	0.994	0.868	0.926	2142
1.0	0.467	0.954	0.627	260
accuracy			0.877	2402
macro avg	0.730	0.911	0.777	2402
weighted avg	0.937	0.877	0.894	2402

Figure 8 – Classification Report for LR training set of 0.09 threshold

As observed above, accuracy of the model i.e. percentage of overall correct predictions is 87.7% Sensitivity of the model is 95.4% i.e. 95.4% of those defaulted were correctly identified as defaulters by the model. Precision is seen at 46.7% for defaulters.

- **AUC and ROC for LR training set of 0.09 threshold**

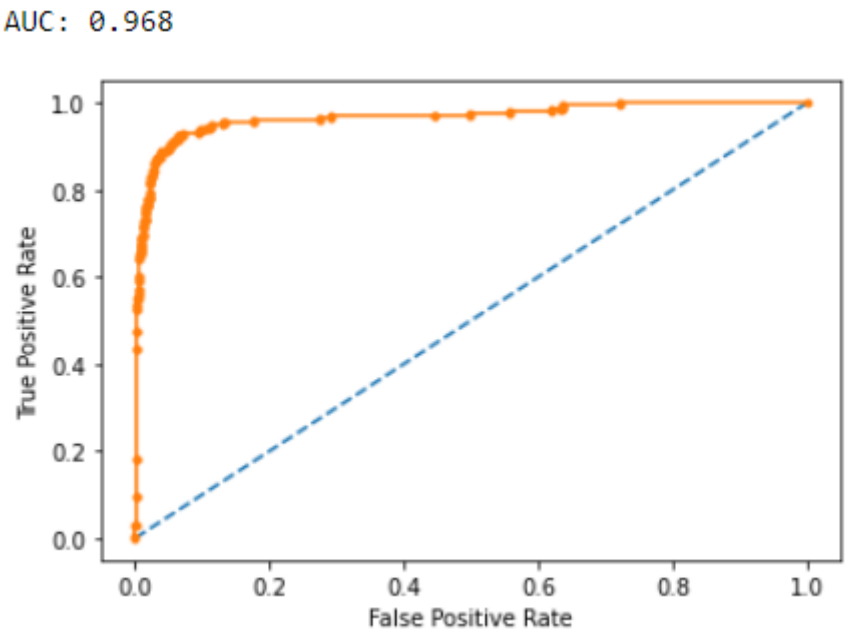

Figure 9 – AUC and ROC for LR training set of 0.09 threshold

- **Confusion matrix for LR test set of 0.09 threshold**

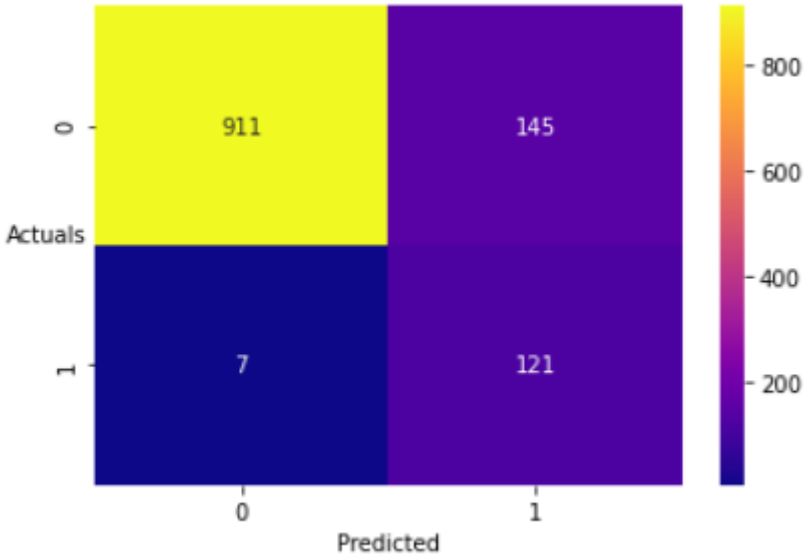

Figure 10 – Confusion matrix for LR test set of 0.09 threshold

- **Classification Report for LR test set of 0.09 threshold**

	precision	recall	f1-score	support
0.0	0.992	0.863	0.923	1056
1.0	0.455	0.945	0.614	128
accuracy			0.872	1184
macro avg	0.724	0.904	0.769	1184
weighted avg	0.934	0.872	0.890	1184

Figure 11 – Classification Report for LR test set of 0.09 threshold

As observed above, accuracy of the model i.e. percentage of overall correct predictions is 87.2% Sensitivity of the model is 94.5% i.e. 94.5% of those defaulted were correctly identified as defaulters by the model and the result is similar to the train data as well. Precision is 45.5 % for the defaulters.

- **AUC and ROC for LR test set of 0.09 threshold**

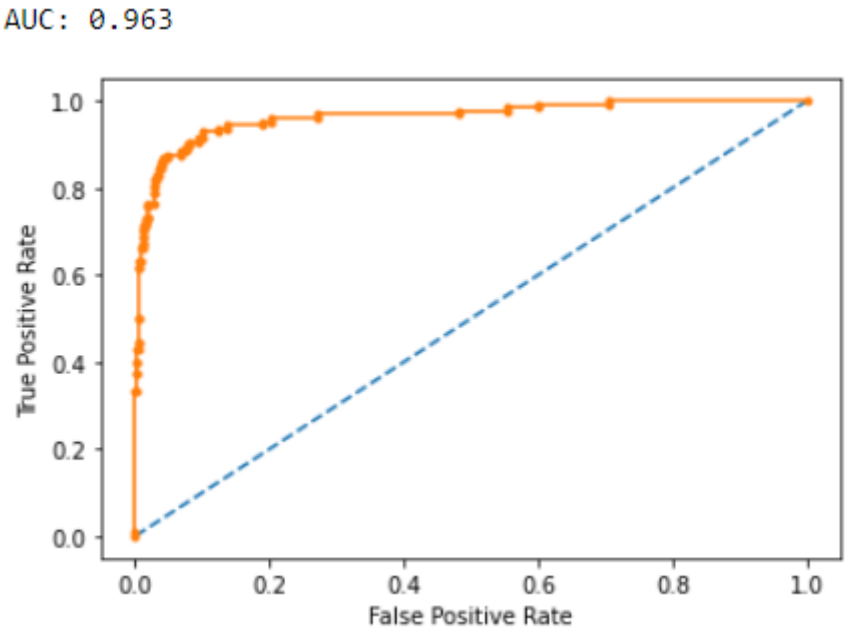

Figure 12 – AUC and ROC for LR test set of 0.09 threshold

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

Random Forest (RF) Model:

Considered Parameters for hyper tuning in the model building approach:

'max_depth': [4, 7, 15],
'min_samples_leaf': [10, 25, 50],
'min_samples_split': [30, 75, 150],

'n_estimators': [75, 150, 225]

Model building approach:
We had 2402 observations and 64 variables in our train set in total.

To keep inclusion of correlated variables to a minimum, we provided smaller values for 'max depth' parameter of 4, 7 and 15.

'Min_sample_leaf' was also kept proportional to < 1% (10), 1% (25) and 2% (50) of the total observations in our train set and followed it by considering 3 times the figures of 'Min_sample_leaf' to determine the 'min_samples_split'.

The 'n_estimators' were kept in the range of 3% to 10% of the total observations in our train set

The Random Forest model was build based on the below Best Parameters suggested by GridSearchCV from the above subsets.

Best Parameters suggested by GridSearchCV:

```
{'max_depth': 7,
 'min_samples_leaf': 10,
 'min_samples_split': 75,
 'n_estimators': 75}
```

- **Confusion matrix for RF training set**

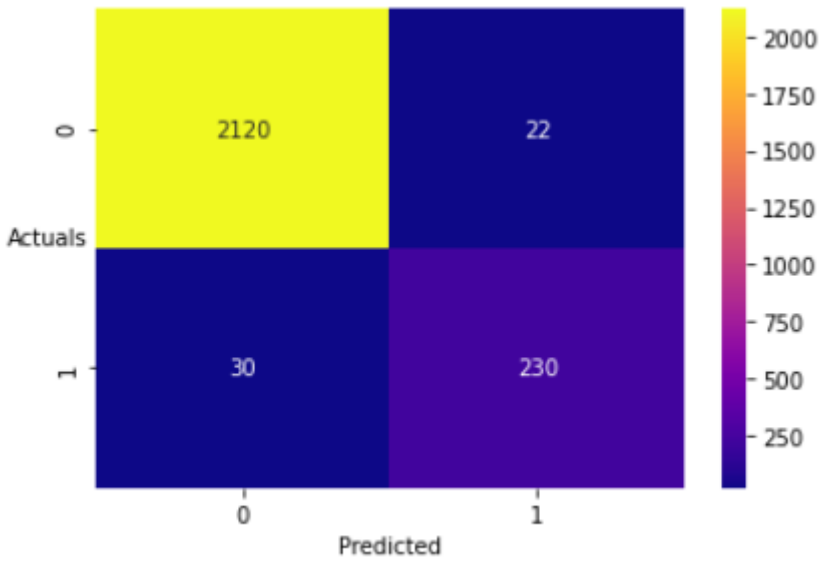

Figure 13 – Confusion matrix for RF training set

- **Classification Report for RF training set**

	precision	recall	f1-score	support
0.0	0.99	0.99	0.99	2142
1.0	0.91	0.88	0.90	260
accuracy			0.98	2402
macro avg	0.95	0.94	0.94	2402
weighted avg	0.98	0.98	0.98	2402

Figure 14 – Classification Report for RF training set

As observed above, accuracy of the model i.e. percentage of overall correct predictions is 98.% Sensitivity of the model is 88% i.e. 88% of those defaulted were correctly identified as defaulters by the model.

Whereas, the sensitivity is significantly lower for defaulter compared to the previous LR model with threshold of 0.09, the accuracy and the precision has also improved significantly.

AUC and ROC for RF train set

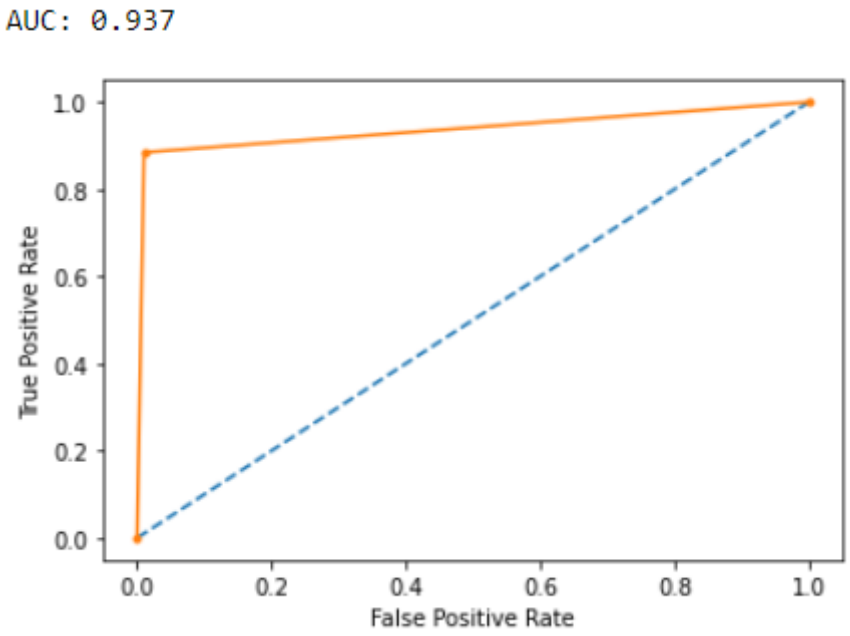

Figure 15 – AUC and ROC for RF train set

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

- **Confusion matrix for RF test set**

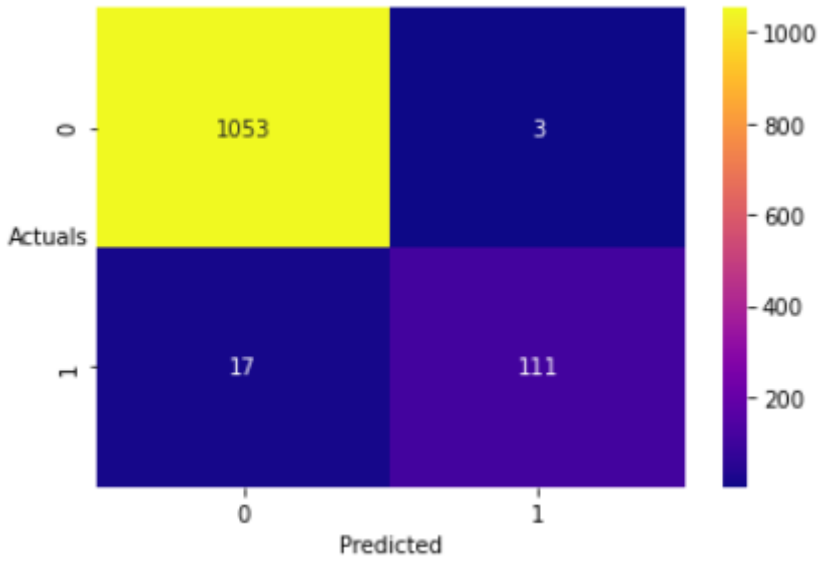

Figure 16 – Confusion matrix for RF test set

- **Classification Report for RF test set**

	precision	recall	f1-score	support
0.0	0.98	1.00	0.99	1056
1.0	0.97	0.87	0.92	128
accuracy			0.98	1184
macro avg	0.98	0.93	0.95	1184
weighted avg	0.98	0.98	0.98	1184

Figure 17 – Classification Report for RF test set

As observed above, accuracy of the model i.e. percentage overall correct predictions is 98% which is the best so far. Sensitivity of the model is 87% i.e. 87% of those defaulted were correctly identified as defaulters by the model. Moreover, the precision has increased significantly to 97% in test set whereas it was 92% for the train set.

Whereas, the sensitivity is significantly lower for defaulters compared to the previous LR model with threshold of 0.09 but the accuracy and the precision has increased tremendously.

AUC and ROC for RF test set

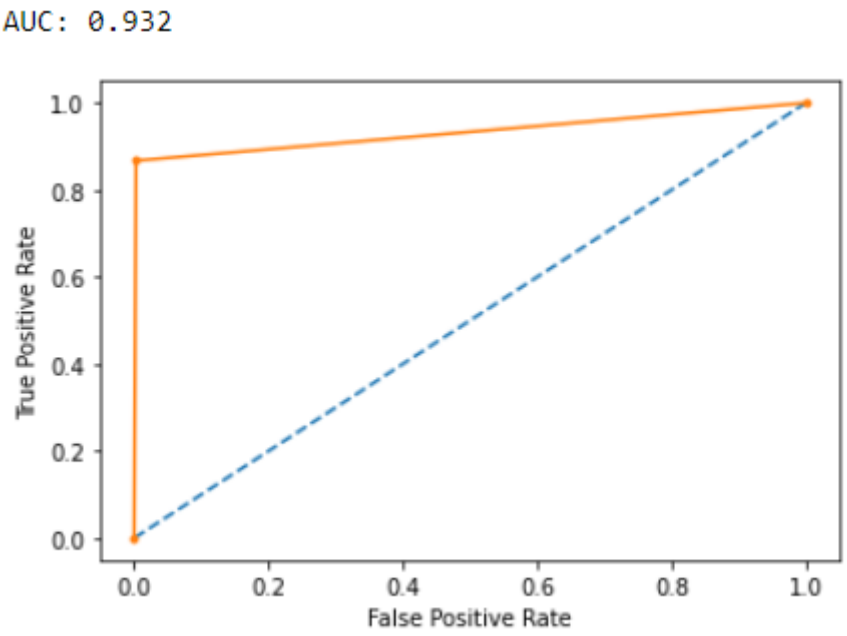

Figure 18: AUC and ROC for RF test set

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

Linear Discriminant Analysis (LDA) Model Building:

- **Confusion matrix for LDA training set**

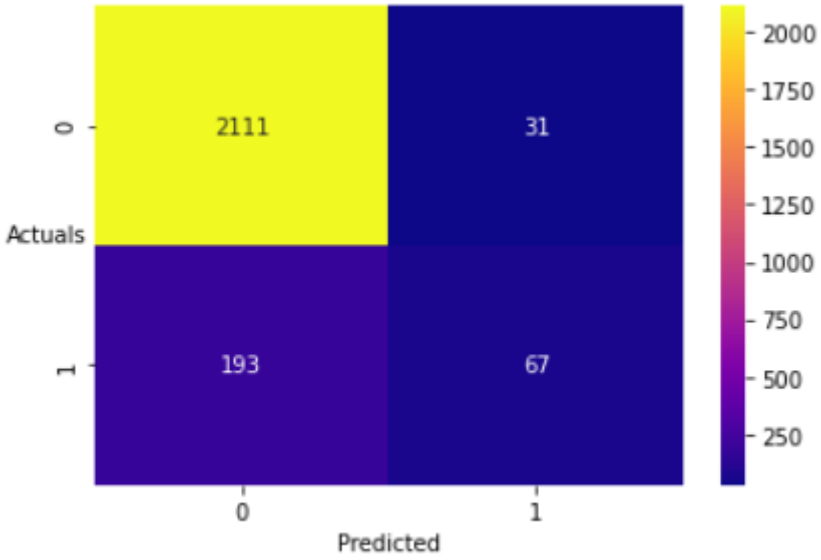

Figure 19 – Confusion matrix for LDA training set

- **Classification Report for LDA training set**

	precision	recall	f1-score	support
0.0	0.92	0.99	0.95	2142
1.0	0.68	0.26	0.37	260
accuracy			0.91	2402
macro avg	0.80	0.62	0.66	2402
weighted avg	0.89	0.91	0.89	2402

Figure 20 – Classification Report for LDA training set

As observed above, accuracy of the model i.e. percentage of overall correct predictions is 91%. Sensitivity of the model is extremely poor at only 26%. Also, the precision is low a 68% for this model.

AUC and ROC for LDA train set

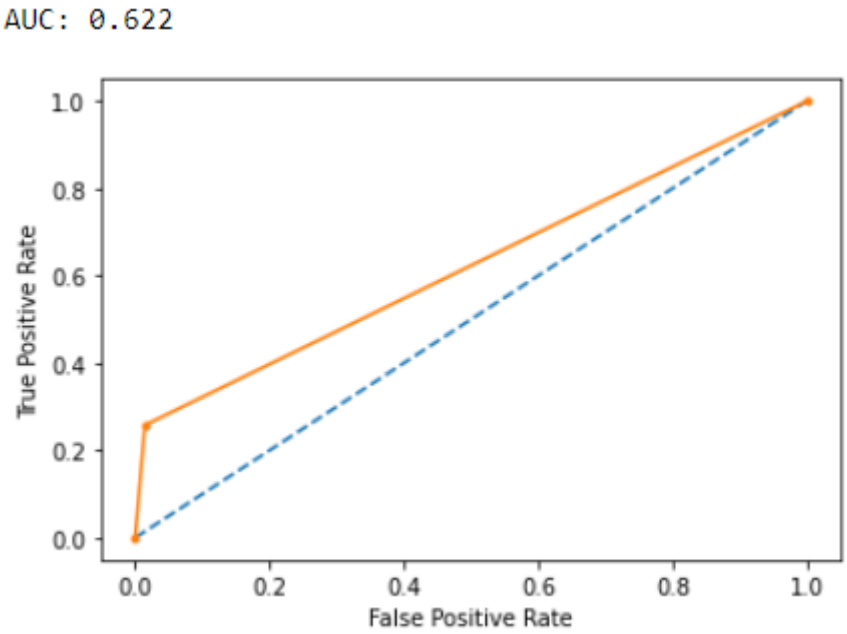

Figure 21: AUC and ROC for LDA train set

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

- **Confusion matrix for LDA test set**

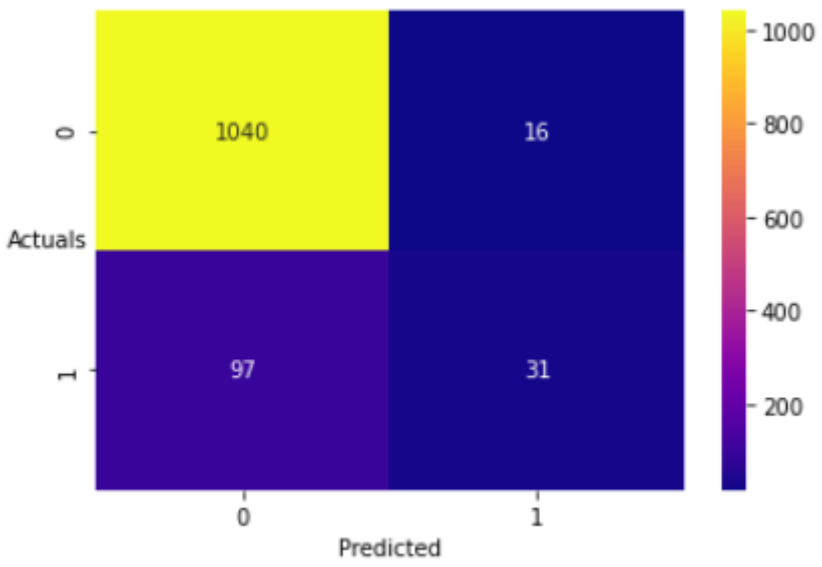

Figure 22 – Confusion matrix for LDA test set

- **Classification Report for LDA test set**

	precision	recall	f1-score	support
0.0	0.91	0.98	0.95	1056
1.0	0.66	0.24	0.35	128
accuracy			0.90	1184
macro avg	0.79	0.61	0.65	1184
weighted avg	0.89	0.90	0.88	1184

Figure 23 – Classification Report for LDA test set

As observed above, accuracy of the model i.e. percentage of overall correct predictions is 90%. Sensitivity of the model is extremely poor at only 24%. Also, the precision is low a 66% for this model.

AUC and ROC for LDA test set

AUC: 0.614

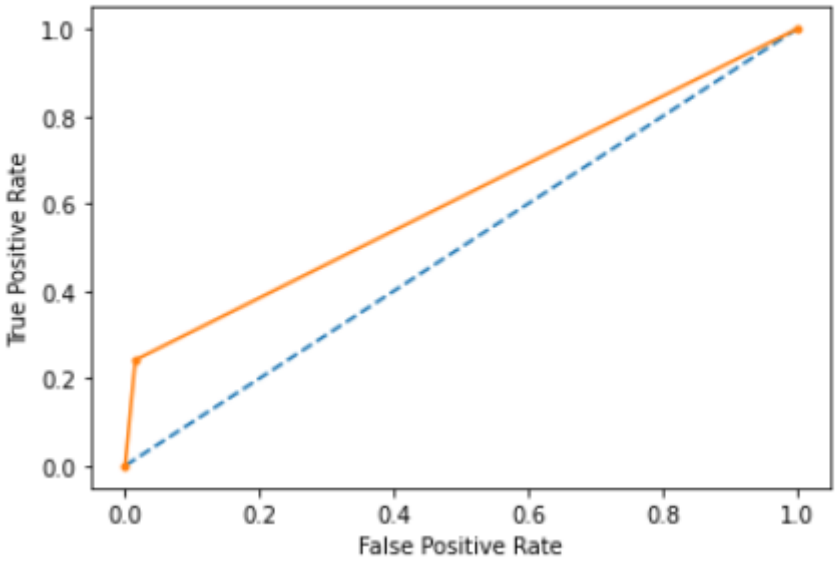

Figure 24 – AUC and ROC for LDA test set

LDA Model building approach:

We fit the model on the train and tried to check the accuracy on the confusion matrix with the predicted values which did not generate a good recall as well as precision value. Hence, we tried to further to find the optimum threshold values, this time on the probability of predicted values.

The result saw that 0.4 and 0.3 gave better accuracy than the rest of the custom cut-off values. But 0.2 cut-off gave us the best 'f1-score'. So, we built a model with cut-off as 0.2.

The model gave an accuracy of 88.9% on test but poor recall of 66.4% and precision on 48.9%. Below is the classification report for the same. We also tried to build another LDA model with scaled predicted values but it did not provide good results as well (Details can be found in the Python File)

	precision	recall	f1-score	support
0.0	0.957	0.916	0.936	1056
1.0	0.489	0.664	0.563	128
accuracy			0.889	1184
macro avg	0.723	0.790	0.750	1184
weighted avg	0.907	0.889	0.896	1184

Figure 25 – Classification Report for LDA test set with 0.02 threshold

AUC and ROC for LDA test set with 0.02 threshold

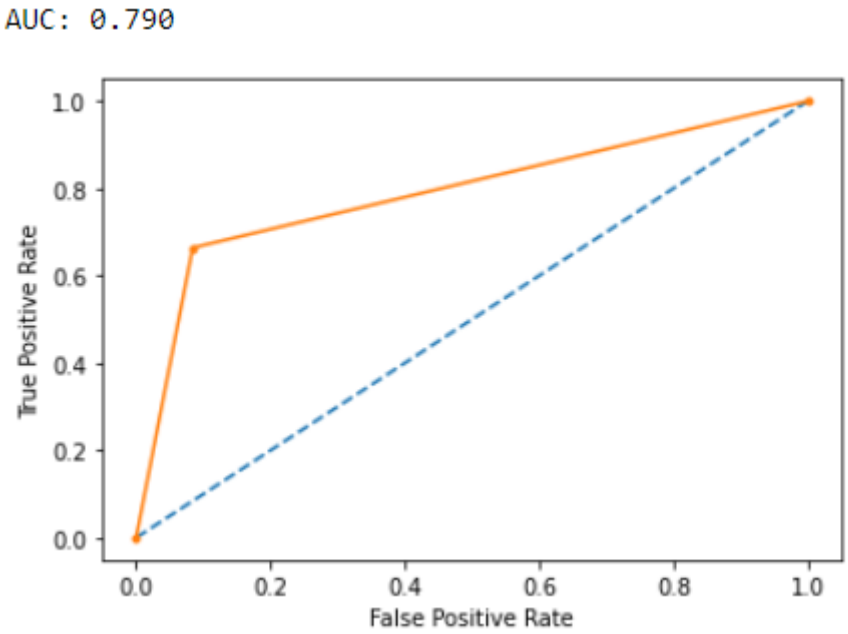

Figure 26 – AUC and ROC for LDA test set with 0.02 threshold

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

Our objective of the analysis is to correctly predict the defaulters which two of the four models built were able to achieve. Of these two, Logistic Regression model with 0.09 threshold achieved a recall of 94.5% and Random Forest had a recall of 87% on their respective test sets. They equally performed better on their train sets as well where Logistic Regression model with 0.09 threshold's recall was 95.4% and Random Forest recall was of 88%.

W.r.t the accuracy, precision and f scores also both these models performed well on the train as well as test sets. Whereas, both the Linear Discriminant Analysis models did not perform better than the Logistic Regression model and Random Forest model even after building a couple of more LDA models by scaling the predictors and also by using the optimum threshold cut off value. Hence, LDA can't be a usable approach for this dataset.

On basis of recall alone for Class 1 (defaulters), Logistic Regression model beats the performance of Random Forest by about 6.5% which is a huge margin when it comes to credit risk as otherwise an incorrect decision to identify a defaulter as non-defaulter could prove in heavy losses for a creditor, while identifying a non-defaulter as defaulter may not be of a great concern.

However, then a creditor may also miss on opportunity to grow one's business and we can see with Random Forest model, it also managed to identify non-defaulters as non -defaulters precisely (recall for test is 100% on test and 99% on train). So, Random Forest model brings double benefits to creditors.

Hence, even if Logistic Regression model has the best recall, it can be argued that overall Random Forest is the better model as it has a way better precision, better accuracy, better f score and equally good recall as well for both defaulters and non-defaulters. Therefore, we determine that Random Forest is the best model for this dataset.

	ACCURACY (%)	PRECISION (%)	RECALL (%)	F SCORE (%)
LR Thres 0.09 TRAIN	87.7	46.7	95.4	62.7
LR Thres 0.09 TEST	87.2	45.5	94.5	61.4
RF TRAIN	98	91	88	90
RF TEST	98	97	87	92
LDA TRAIN	91	68	26	37
LDA TEST	90	66	24	35
LDA Thres 0.02 TEST	88.9	48.9	66.4	56.3

Table 1 - Comparison Chart for all models

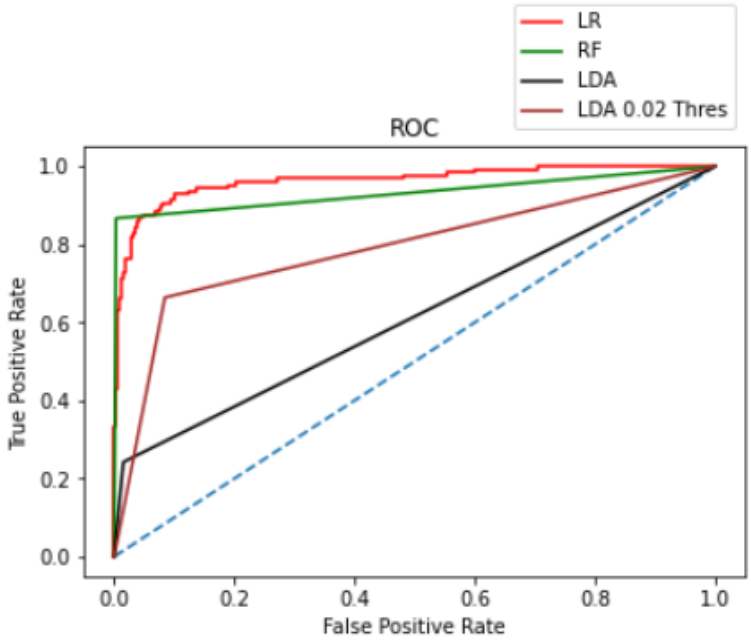

Figure 27 – AUC and ROC for all models

1.13 State Recommendations from the above models

Recommendations:

- LDA Models usually assume that the data is normally distributed, variance is the same across all predictors and multicollinearity decreases the predictive power of the model. For this data set all these assumptions are untrue as data is not normally distributed for most variables, variance is also difference and there is a lot of multicollinearity between the predictors. Hence, this model did not perform for this dataset. Therefore, LDA would not be recommended for such datasets.

- LR models usually perform better than LDA models for binary classification based problems and our dataset was a binary classification based problems and hence, we could see that LR

model performed much better than the LDA models despite trying various methods to improve their performance. Hence, it is recommended to try LDA tried on multi-classification dataset with low no of variables.

- LR models performed well for another reason as the dataset had imbalanced classes. We could have used smote to reduce this imbalance and then check the performance. However, it would be a last resort to smote unless we found a better workable model, which we found in Random Forest. Hence, it is advisable to use LR based models wherever there is a dataset with class imbalance and we are not suppose to smote the data as it will provide a realistic solution as per the actual data in hand.

- LR did perform better in predicting the defaulters. However, the precision was not as good as expected. This could be majorly because we were still left with multicollinearity in the predictors and LR models work better with independent variables. Hence, these models should not be given priority especially to a dataset as ours where most of the financial ratios happen to be interconnected or related to each other.

- Random Forest performed well as it is follows a wisdom or the crowd approach where it combines several based models and then provides the most suitable prediction. Also, the collinearity factor is also taken care of as the decision tree built are not correlated to each other. Hence, it is recommended to approach this model building method for critical and/or sensitive predictions to try and get the best results. One can easily extract/distinguish between the significant and non-significant predictors using this technique and it is easy to execute as well. Below we have provided a list of all the variables which were identified as significant by our Random Forest model to conclude our analysis. Businesses can take note and structure their credit provisions accordingly.

Significant Predictors basis RF Model (Descending Order):

	Imp
Networth	0.24
Book_Value_Unit_Curr	0.21
Book_Value_Adj_Unit_Curr	0.19
Capital_Employed	0.06
Curr_Ratio_Latest	0.04
PBDT	0.03
CEPS_annualised_Unit_Curr	0.03
CP	0.03
Net_Working_Capital	0.02
PAT	0.02
PBIT	0.01
PBIDT	0.01
ROG_Capital_Employed_perc	0.01
PBT	0.01
Adjusted_PAT	0.01
ROG_Net_Worth_perc	0.01
PBDTM_perc_Latest	0.01
Total_Debt	0.01
PBITM_perc_Latest	0.01
Total_Asset_Turnover_Ratio_Latest	0.01
APATM_perc_Latest	0.01
Interest_Cover_Ratio_Latest	0.01

Figure 28 : Significant Predictors basis RF Model

PROBLEM 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. You are expected to do the Market Risk Analysis.

Problem Statement 2:

Perform Market Risk Analysis by comparing the mean returns and standard deviation for the stocks provided in the dataset and provide insights, conclusion and recommendations based on the analysis.

Dataset Snippet:

	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
...	...	...	...	...	...	...	...	...	...	...	...
309	02-03-2020	729	120	469	658	33	23110	401	146	3	22
310	09-03-2020	634	114	427	569	30	21308	384	121	6	18
311	16-03-2020	577	90	321	428	27	18904	365	105	3	16
312	23-03-2020	644	75	293	360	21	17666	338	89	3	14
313	30-03-2020	633	75	284	379	23	17546	352	82	3	14

There are a total of 314 rows and 11 columns in the dataset. The dataset contains weekly stock prices for 10 companies as per mentioned in the column names and 1 date row. The dates range from 31st March 2014 till 30th March 2020.

Dataset Shape:

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

Dataset Info:

```
<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
dtypes: int64(10), object(1)
```

There are no null values and duplicate rows in the dataset. It contains 10 integer type and 1 object type variables. However, 'Date' variable is incorrectly captured as object data type which will not be suitable to run analysis and hence, we would be adding another variable for this variable which will be of the correct format ie; of date and time format and use that instead. Below image includes the corrected variable named 'date' for your reference.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 314 entries, 0 to 313
Data columns (total 12 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
 11  dates                 314 non-null    datetime64[ns]
dtypes: datetime64[ns](1), int64(10), object(1)
```

Dataset Description (Continuous Variables):

	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

Mean and Median values are almost similar for all the variables. Shree Cement seems to have the highest stock price among the variables whereas, Idea Vodafone had the lowest historical low stock price of 3 while SAIL had the lowest historical high price of 104.

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

A) Infosys Stock Price Graph

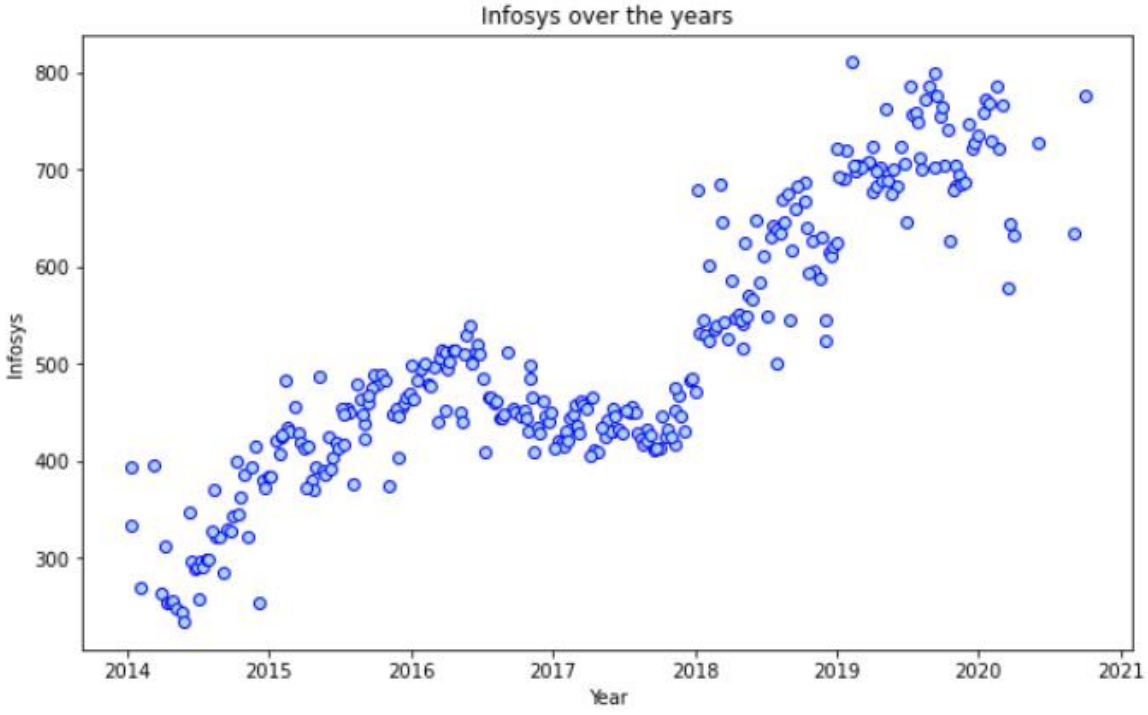

Figure 29 - Infosys Stock Price Graph

Infosys has seen an uptrend for most of the period for these last 6 years rising from a level of approx 250 to 800. It had slightly dipped in the year 2016 to 2017 and performed flattish from 2017 to 2018 before again gaining upward momentum. We can also notice slight fluctuations in the stock prices between 2020 to 2021 which majorly could be due to the pandemic.

B) Mahindra & Mahindra Stock Price Graph

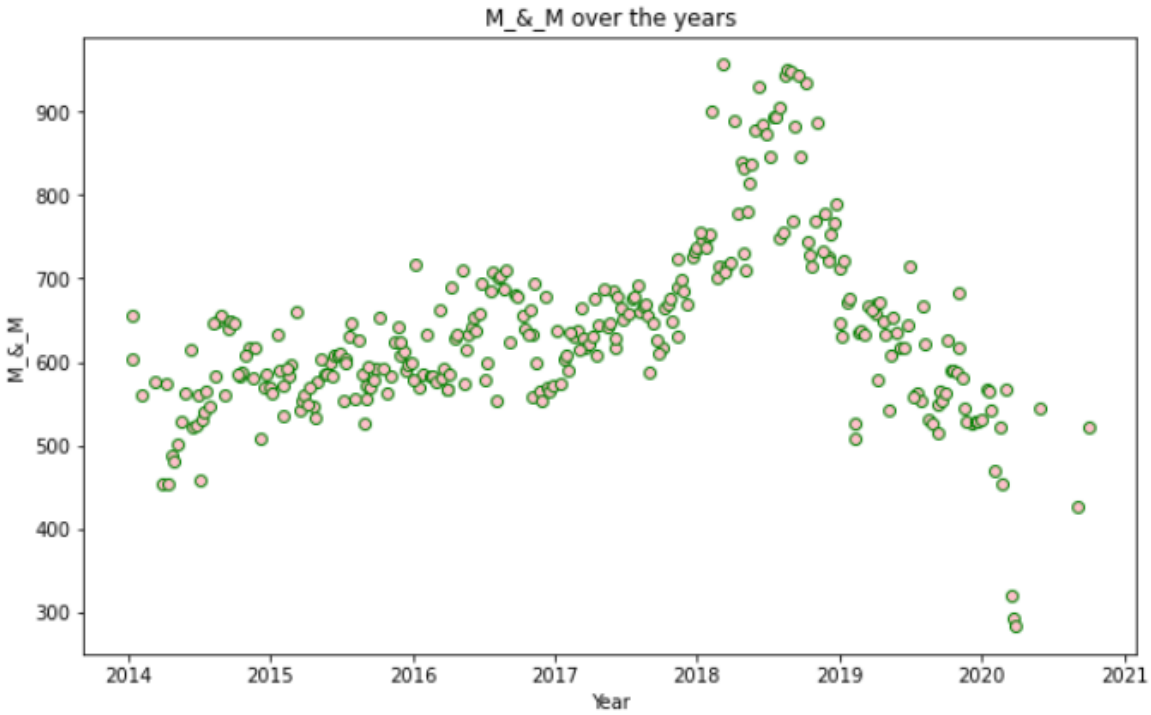

Figure 30 - Mahindra & Mahindra Stock Price Graph

Mahindra and Mahindra has seen both upward and downward trends in these last 6 years. The stock price rose steadily between 2014 to 2018 rising from a level of approx 450 to 700 and then suddenly gained a lot of momentum in 1st half of 2018 which helped it climb to touch a high of approx 950 before stumbling down back sharply to levels of as low as 300 in these 2 years from mid-2018 till 2020. The sharp fall in the stock prices between 2020 to 2021 could also have been largely due to the pandemic.

2.2 Calculate Returns for all stocks with inference

Steps for calculating returns from prices on a weekly basis:
- Take logarithms
- Take differences

Stock Returns (Head):

	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

Stock Returns (Tail):

	Infosys	Indian_Hotel	Mahindra_&_Mahindra	Axis_Bank	SAIL	Shree_Cement	Sun_Pharma	Jindal_Steel	Idea_Vodafone	Jet_Airways
309	0.009649	-0.110348	0.030305	-0.057580	-0.087011	0.023688	0.072383	-0.053346	-0.287682	-0.127833
310	-0.139625	-0.051293	-0.093819	-0.145324	-0.095310	-0.081183	-0.043319	-0.187816	0.693147	-0.200671
311	-0.094207	-0.236389	-0.285343	-0.284757	-0.105361	-0.119709	-0.050745	-0.141830	-0.693147	-0.117783
312	0.109856	-0.182322	-0.091269	-0.173019	-0.251314	-0.067732	-0.076851	-0.165324	0.000000	-0.133531
313	-0.017228	0.000000	-0.031198	0.051432	0.090972	-0.006816	0.040585	-0.081917	0.000000	0.000000

The above details are the difference in the stock prices on a weekly basis in top to bottom approach. Hence, the 1st row has NaN value in it as there is no stock price for the previous week to generate the difference.

The log values represent the drop or rise in the stock prices on a weekly basis.

Wherever, the subsequent row has a negative log value, it suggests that the stock price had fallen from its previous week as shown in the below image.

Infosys
0.009649
-0.139625

Wherever, the subsequent row has a positive log value, it suggests that the stock price had risen from its previous week as shown in the below image.

Idea_Vodafone
-0.287682
0.693147

Wherever, the subsequent row has a 0 log value, it suggests that there was no change in the stock price from its previous week as shown in the below image.

Indian_Hotel
NaN
-0.014599
0.000000

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

We now look at Means & Standard Deviations of these returns

- Stock Means: Average returns that the stock is making on a week to week basis
- Stock Standard Deviation : It is a measure of volatility meaning the more a stock's returns vary from the stock's average return, the more volatile the stock

Calculating stock means (Descending Order):

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

Shree Cement has given the highest average returns when compared on a week on week basis during these 6 years followed by Infosys in 2nd & Axis Bank in 3rd Spot and Indian Hotel in 4th. These 4 stocks have given a positive return on an average.

Sun Pharma, Mahindra & Mahindra, SAIL, Jindal Steel, Jet Airways & Idea Vodafone all have given negative average weekly returns during these 6 years with Idea Vodafone giving the lowest returns to their shareholders followed by Jet Airways in 2nd and Jindal Steel in 3rd place.

Calculating stock standard deviation (Descending Order):

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

Idea Vodafone is the most volatile stock w.r.t weekly returns in these 6 years followed by Jet Airways in 2nd and Jindal Steel in 3rd place. Whereas, Infosys is the least volatile stock followed by Shree Cement in 2nd and Mahindra & Mahindra in 3rd place.

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

We will first add the Stock Return Means and Stock Standard Deviation in a dataframe and rename Stock Return Means to ‘Average’ and Standard Deviation to ‘Volatility’. Below is the image of the dataframe created.

	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

Plotting Stock Means and Standard Deviation:

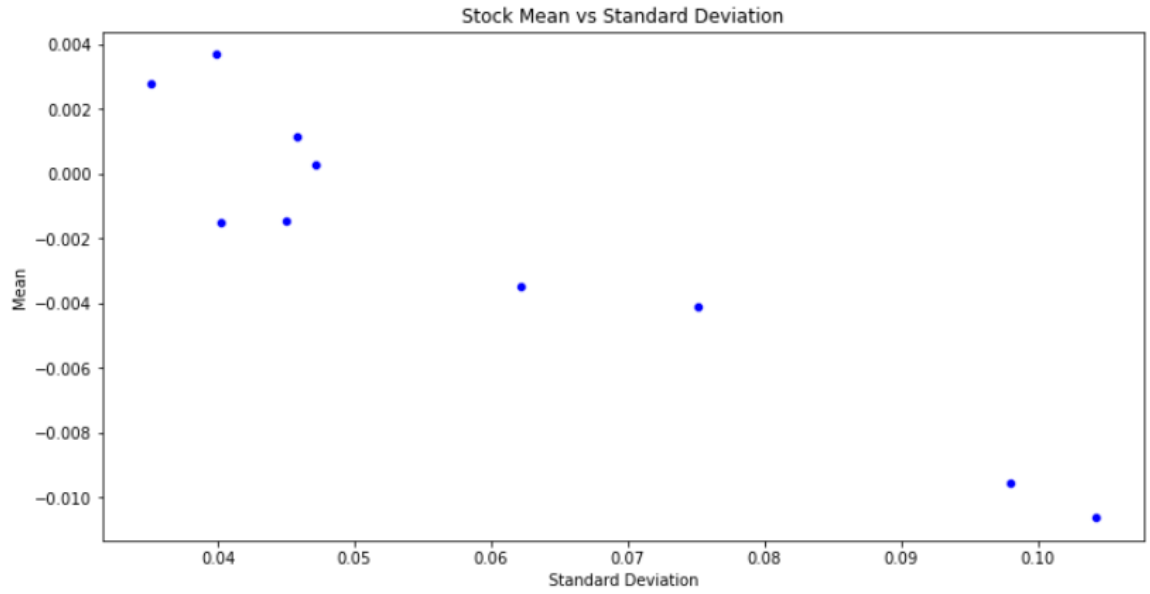

Figure 31 - Stock Mean Vs Stock Standard Deviation

The above plot helps to give a glimpse of the risk to reward characteristics of the stocks with the returns means on the Y axis and Standard Deviation on the X axis. Risk here represents Standard Deviation and Reward represents the Stock Return Mean.

There are a few stocks such as Infosys, Shree Cement, Indian Hotel which have provided better returns comparatively at lower standard deviations.

Also, there are some stocks such as Mahindra & Mahindra, Sun Pharma and Axis Bank which are less risky but have performed poorly when it comes to generating returns.

Then, there are stocks such as Jindal Steel and SAIL which are moderately risky but do not generate enough returns as well.

Whereas, there a stocks such as Idea Vodafone and Jet Airways which are highly risky but which also end up generating the least returns.

We now try and compare the stocks return means and std deviations with all the 10 stocks average total return means and all the 10 stocks average total std deviations

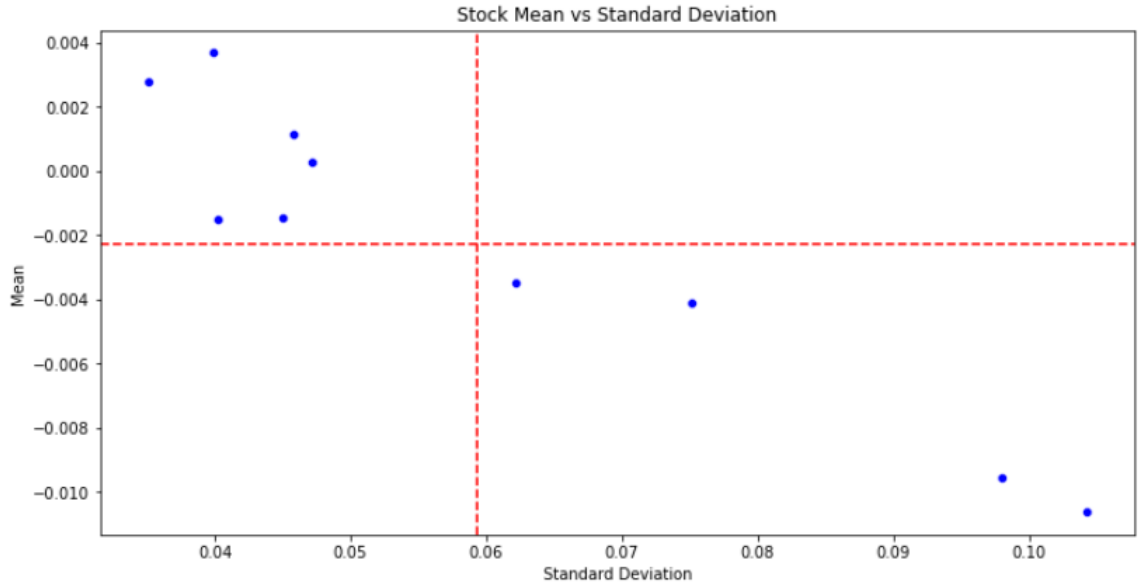

Figure 32 - Stock Mean Vs Stock Standard Deviation II

The red dotted lines on Y axis is the average total return mean for all the 10 stocks and red dotted lines on X axis average total std deviation for all the 10 stocks.

As we can see from above, the stocks on the left side of the red dotted line on std deviation axis and above the red dotted line on the mean axis are the stocks that can be considered for investment from these 10 stocks to gain returns more than the average for this portfolio alone and these stocks can be further narrowed down basis the one's that provide higher returns or lower standard deviations depending on the investors stock investing preference and risk appetite.

2.5 Conclusion and Recommendations

- Probability of Return and Risk to achieve this return is something that every investor should calculate before making an important investment decision. The above steps work as a guideline to make those calculations given that the data is available and accurate.

- An apt portfolio for any investor should be able to generate high returns but at a lower volatility to increase the probability of not making losses/secure capital in the long run.

- Average Returns and Standard Deviation are very important characteristics of any stock's health and should be given a priority while making stock selections or even diversifying one's portfolio.

- Investors should refrain from stocks such as or similar to Idea Vodafone & Jet Airways (high volatile - low return stocks) or sell them if they are part of the portfolio and instead add stocks such as or similar to Infosys, Shree Cement, Indian Hotel (low volatile - high return stocks) instead.

- If you have high return high volatile stocks already in your portfolio then you can consider adding Axis Bank for low risk low return stock in the portfolio to reduce/balance the risk of your portfolio without expecting too much returns, simply for diversification purpose.

- Mahindra & Mahindra and Sun Pharma can be tracked for further few years w.r.t returns and volatility and in case of improvements in returns, they can also be good buying opportunities to reduce/balance the risk of an investor's portfolio.

- Jindal Steel and SAIL also must be avoided unless they appear in the top left of the figure .... after which they can be a part of an investors watchlist who is looking at opportunities in the metals sector especially in steel industry.

THE END