FRA PROJECT BUSINESS REPORT

Problem Statement A:

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 interest 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.

Data Description

Sl. No	Column Name	Description
1	Co_Code	Company Code
2	Co_Name	Company Name
3	_Operating_Expense_Rate	Operating Expense Rate: Operating Expenses/Net Sales. The operating expense ratio (OER) is the cost to operate a piece of property compared to the income the property brings in.
4	_Research_and_development_expense_rate	Research and development expense rate: (Research and Development Expenses)/Net Sales. Research and development (R&D) expenses are direct expenditures relating to a company's efforts to develop, design, and enhance its products, services, technologies, or processes.
5	_Cash_flow_rate	Cash flow rate: Cash Flow from Operating/Current Liabilities. Cash flow is a measure of how

Sl. No	Column Name	Description
		much cash a business brought in or spent in total over a period of time.
6	_Interest_bearing_debt_interest_rate	Interest-bearing debt interest rate: Interest-bearing Debt/Equity
7	_Tax_rate_A	Tax rate (A): Effective Tax Rate. Effective tax rate represents the percentage of their taxable income that individuals pay in taxes. For corporations, the effective corporate tax rate is the rate they pay on their pre-tax profits.
8	_Cash_Flow_Per_Share	Cash Flow Per Share. It is the after-tax earnings plus depreciation on a per-share basis that functions as a measure of a firm's financial strength
9	*Per_Share_Net_profit_before_tax_Yuan*	Per Share Net profit before tax (Yuan ¥): Pretax Income Per Share. Pretax income, also known as earnings before tax or pretax earnings, is the net income earned by a business before taxes are subtracted/accounted for.
10	_Realized_Sales_Gross_Profit_Growth_Rate	Realized Sales Gross Profit Growth Rate.
11	_Operating_Profit_Growth_Rate	Operating Profit Growth Rate: Operating Income Growth. It is the rate of increase in operating income over the last year.
12	_Continuous_Net_Profit_Growth_Rate	Continuous Net Profit Growth Rate: Net Income-Excluding Disposal Gain or Loss Growth
13	_Total_Asset_Growth_Rate	Total Asset Growth Rate: Total Asset Growth. It is the rate at which how quickly the company has been growing its Assets

Sl. No	Column Name	Description
14	_Net_Value_Growth_Rate	Net Value Growth Rate: Total Equity Growth
15	_Total_Asset_Return_Growth_Rate_Ratio	Total Asset Return Growth Rate Ratio: Return on Total Asset Growth
16	_Cash_Reinvestment_perc	Cash Reinvestment %: Cash Reinvestment Ratio. It is the valuation ratio that is used to measure the percentage of annual cash flow that the company invests back into the business as a new investment.
17	_Current_Ratio	Current Ratio. The current ratio describes the relationship between a company's assets and liabilities
18	_Quick_Ratio	Quick Ratio: Acid Test. Acid-test ratio (also known as quick ratio) is a measure of a company's liquidity, which is its ability to pay its short-term obligations using only its most liquid assets.
19	_Interest_Expense_Ratio	Interest Expense Ratio: Interest Expenses/Total Revenue
20	_Total_debt_to_Total_net_worth	Total debt/Total net worth: Total Liability/Equity Ratio
21	_Long_term_fund_suitability_ratio_A	Long-term fund suitability ratio (A): (Long-term Liability+Equity)/Fixed Assets
22	_Net_profit_before_tax_to_Paid_in_capital	Net profit before tax/Paid-in capital: Pretax Income/Capital
23	_Total_Asset_Turnover	Total Asset Turnover. Net Sales/Average Total Assets
24	_Accounts_Receivable_Turnover	Accounts Receivable Turnover. The accounts receivable turnover ratio, or receivables turnover, is used in business accounting to quantify how well companies are managing the credit that they extend to their customers by evaluating how

Sl. No	Column Name	Description
		long it takes to collect the outstanding debt throughout the accounting period.
25	_Average_Collection_Days	Average Collection Days: Days Receivable Outstanding
26	_Inventory_Turnover_Rate_times	Inventory Turnover Rate (times). The inventory turnover ratio is the number of times a company has sold and replenished its inventory over a specific amount of time. The formula can also be used to calculate the number of days it will take to sell the inventory on hand.
27	_Fixed_Assets_Turnover_Frequency	Fixed Assets Turnover Frequency. Fixed Asset Turnover (FAT) is an efficiency ratio that indicates how well or efficiently a business uses fixed assets to generate sales. This ratio divides net sales by net fixed assets, calculated over an annual period.
28	_Net_Worth_Turnover_Rate_times	Net Worth Turnover Rate (times): Equity Turnover. Equity turnover is a ratio that measures the proportion of a company's sales to its stockholders' equity. The intent of the measurement is to determine the efficiency with which management is using equity to generate revenue.
29	_Operating_profit_per_person	Operating profit per person: Operation Income Per Employee
30	_Allocation_rate_per_person	Allocation rate per person: Fixed Assets Per Employee
31	_Quick_Assets_to_Total_Assets	Quick Assets/Total Assets
32	_Cash_to_Total_Assets	Cash/Total Assets
33	_Quick_Assets_to_Current_Liability	Quick Assets/Current Liability

Sl. No	Column Name	Description
34	_Cash_to_Current_Liability	Cash/Current Liability
35	_Operating_Funds_to_Liability	Operating Funds to Liability
36	_Inventory_to_Working_Capital	Inventory/Working Capital
37	_Inventory_to_Current_Liability	Inventory/Current Liability
38	_Long_term_Liability_to_Current_Assets	Long-term Liability to Current Assets
39	_Retained_Earnings_to_Total_Assets	Retained Earnings to Total Assets
40	_Total_income_to_Total_expense	Total income/Total expense
41	_Total_expense_to_Assets	Total expense/Assets
42	_Current_Asset_Turnover_Rate	Current Asset Turnover Rate: Current Assets to Sales. The current assets turnover ratio indicates how many times the current assets are turned over in the form of sales within a specific period of time. A higher asset turnover ratio means a better percentage of sales.
43	_Quick_Asset_Turnover_Rate	Quick Asset Turnover Rate: Quick Assets to Sales. The asset turnover ratio measures the efficiency of a company's assets in generating revenue or sales.
44	_Cash_Turnover_Rate	Cash Turnover Rate: Cash to Sales. The cash turnover ratio is an efficiency ratio that reveals the number of times that cash is turned over in an accounting period.
45	_Fixed_Assets_to_Assets	Fixed Assets to Assets. Fixed assets are also known as non-current assets—assets that can't be easily converted into cash.
46	_Cash_Flow_to_Total_Assets	Cash Flow to Total Assets. This ratio indicates the cash a company can generate in relation to its size.
47	_Cash_Flow_to_Liability	Cash Flow to Liability. The amount of money available to

Sl. No	Column Name	Description
		run business operations and complete transactions. This is calculated as current assets (cash or near-cash assets, like notes receivable) minus current liabilities (liabilities due during the upcoming accounting period)
48	_CFO_to_Assets	CFO to Assets. Cash flow on total assets is an efficiency ratio that rates cash flows to the company assets without being affected by income recognition or income measurements.
49	_Cash_Flow_to_Equity	Cash Flow to Equity. cash flow to equity is a measure of how much cash is available to the equity shareholders of a company after all expenses, reinvestment, and debt are paid.
50	_Current_Liability_to_Current_Assets	Current Liability to Current Assets. Current liabilities are a company's financial commitments that are due and payable within a year, Current assets are projected to be consumed, sold, or converted into cash within a year or within the operational cycle.
51	_Liability_Assets_Flag	Liability-Assets Flag: 1 if Total Liability exceeds Total Assets, 0 otherwise
52	_Total_assets_to_GNP_price	Total assets to GNP price. Gross National Product (GNP) is the total value of all finished goods and services produced by a country’s citizens in a given financial year, irrespective of their location.
53	_No_credit_Interval	No-credit Interval
54	_Degree_of_Financial_Leverage_DFL	Degree of Financial Leverage (DFL). The degree of financial

Sl. No	Column Name	Description
		leverage is a financial ratio that measures the sensitivity in fluctuations of a company's overall profitability to the volatility of its operating income caused by changes in its capital structure.
55	_Interest_Coverage_Ratio_Interest_expense_to_EBIT	Interest Coverage Ratio (Interest expense to EBIT). The interest coverage ratio is a debt and profitability ratio used to determine how easily a company can pay interest on its outstanding debt. The interest coverage ratio is calculated by dividing a company's earnings before interest and taxes (EBIT) by its interest expense during a given period.
56	_Net_Income_Flag	Net Income Flag: 1 if Net Income is Negative for the last two years, 0 otherwise
57	_Equity_to_Liability	Equity to Liability Ratio.
58	Default	Whether the Company has Default (Bankrupted) or not? 1 - Defaulted, 0 - Not Defaulted.

Basic Information about data:-

- The shape of. the dataset is (2058, 58), it means it has 2058 rows and 58 columns
- The size of the dataset is 119364
- The data has 298 null values
- "*Cash_Flow_Per_Share" has 167 null values, "*Total_debt_to_Total_net_worth" has 21 null values , "*Quick_Assets_to_Total_Assets" has 96 null values, "*Current_Liability_to_Current_Assets" has 14 null values
- 298/119364 = 0.00249656512851446, 2% of the data is missing in dataset
- There are no duplicate rows in data

```
 #   Column                                            Non-Null Count  Dtype
---  ------                                            --------------  -----
 0   Co_Code                                           2058 non-null   int64
 1   Co_Name                                           2058 non-null   
object
 2   _Operating_Expense_Rate                           2058 non-null   
float64
 3   _Research_and_development_expense_rate            2058 non-null   
float64
 4   _Cash_flow_rate                                   2058 non-null   
float64
 5   _Interest_bearing_debt_interest_rate              2058 non-null   
float64
 6   _Tax_rate_A                                       2058 non-null   
float64
 7   _Cash_Flow_Per_Share                              1891 non-null   
float64
 8   _Per_Share_Net_profit_before_tax_Yuan_            2058 non-null   
float64
 9   _Realized_Sales_Gross_Profit_Growth_Rate          2058 non-null   
float64
 10  _Operating_Profit_Growth_Rate                     2058 non-null   
float64
 11  _Continuous_Net_Profit_Growth_Rate                2058 non-null   
float64
 12  _Total_Asset_Growth_Rate                          2058 non-null   
float64
 13  _Net_Value_Growth_Rate                            2058 non-null   
float64
 14  _Total_Asset_Return_Growth_Rate_Ratio             2058 non-null   
float64
 15  _Cash_Reinvestment_perc                           2058 non-null   
float64
 16  _Current_Ratio                                    2058 non-null   
float64
 17  _Quick_Ratio                                      2058 non-null   
float64
 18  _Interest_Expense_Ratio                           2058 non-null   
float64
 19  _Total_debt_to_Total_net_worth                    2037 non-null   
float64
 20  _Long_term_fund_suitability_ratio_A               2058 non-null   
float64
 21  _Net_profit_before_tax_to_Paid_in_capital         2058 non-null   
float64
 22  _Total_Asset_Turnover                             2058 non-null   
float64
 23  _Accounts_Receivable_Turnover                     2058 non-null   
float64
 24  _Average_Collection_Days                          2058 non-null   
```

```
float64
 25  _Inventory_Turnover_Rate_times                     2058 non-null
float64
 26  _Fixed_Assets_Turnover_Frequency                   2058 non-null
float64
 27  _Net_Worth_Turnover_Rate_times                     2058 non-null
float64
 28  _Operating_profit_per_person                       2058 non-null
float64
 29  _Allocation_rate_per_person                        2058 non-null
float64
 30  _Quick_Assets_to_Total_Assets                      2058 non-null
float64
 31  _Cash_to_Total_Assets                              1962 non-null
float64
 32  _Quick_Assets_to_Current_Liability                 2058 non-null
float64
 33  _Cash_to_Current_Liability                         2058 non-null
float64
 34  _Operating_Funds_to_Liability                      2058 non-null
float64
 35  _Inventory_to_Working_Capital                      2058 non-null
float64
 36  _Inventory_to_Current_Liability                    2058 non-null
float64
 37  _Long_term_Liability_to_Current_Assets             2058 non-null
float64
 38  _Retained_Earnings_to_Total_Assets                 2058 non-null
float64
 39  _Total_income_to_Total_expense                     2058 non-null
float64
 40  _Total_expense_to_Assets                           2058 non-null
float64
 41  _Current_Asset_Turnover_Rate                       2058 non-null
float64
 42  _Quick_Asset_Turnover_Rate                         2058 non-null
float64
 43  _Cash_Turnover_Rate                                2058 non-null
float64
 44  _Fixed_Assets_to_Assets                            2058 non-null
float64
 45  _Cash_Flow_to_Total_Assets                         2058 non-null
float64
 46  _Cash_Flow_to_Liability                            2058 non-null
float64
 47  _CFO_to_Assets                                     2058 non-null
float64
 48  _Cash_Flow_to_Equity                               2058 non-null
float64
 49  _Current_Liability_to_Current_Assets               2044 non-null
```

```
float64
 50  _Liability_Assets_Flag                            2058 non-null   int64
 51  _Total_assets_to_GNP_price                        2058 non-null   
float64
 52  _No_credit_Interval                               2058 non-null   
float64
 53  _Degree_of_Financial_Leverage_DFL                 2058 non-null   
float64
 54  _Interest_Coverage_Ratio_Interest_expense_to_EBIT 2058 non-null   
float64
 55  _Net_Income_Flag                                  2058 non-null   int64
 56  _Equity_to_Liability                              2058 non-null   
float64
 57  Default                                           2058 non-null   int64
dtypes: float64(53), int64(4), object(1)
memory usage: 932.7+ KB
```

- The dataset has 53 float, 4 int nad 1 onject datatype variables

count	mean	std	min	25%	50%	75%	max
Co_Code	2058.0	1.757211e+04	2.189289e+04	4.000000	3.674000e+03	6.240000e+03	2.428075e+04
_Operating_Expense_Rate	2058.0	2.052389e+09	3.252624e+09	0.000100	1.578727e-04	3.330330e-04	4.110000e+09
_Research_and_development_expense_rate	2058.0	1.208634e+09	2.144568e+09	0.000000	0.000000e+00	1.994130e-04	1.550000e+09
_Cash_flow_rate	2058.0	4.652426e-01	2.266269e-02	0.000000	4.600991e-01	4.634450e-01	4.680691e-01
_Interest_bearing_debt_interest_rate	2058.0	1.113022e+07	9.042595e+07	0.000000	2.760280e-04	4.540450e-04	6.630660e-04
_Tax_rate_A	2058.0	1.147770e-01	1.524457e-01	0.000000	0.000000e+00	3.709890e-02	2.161909e-01
_Cash_Flow_Per_Share	1891.0	3.199856e-01	1.529979e-02	0.169449	3.149890e-01	3.206479e-01	3.259178e-01
Per_Share_Net_profit_before_tax_Yuan	2058.0	1.769673e-01	3.015730e-02	0.000000	1.666039e-01	1.756421e-01	1.858854e-01

count	mean	std	min	25%	50%	75%	max
_Realized_Sales_Gross_Profit_Growth_Rate	2058.0	2.276117e-02	2.170104e-02	0.004282	2.205831e-02	2.210001e-02	2.215200e-02
_Operating_Profit_Growth_Rate	2058.0	8.481083e-01	4.589093e-03	0.736430	8.479740e-01	8.480386e-01	8.481147e-01
_Continuous_Net_Profit_Growth_Rate	2058.0	2.173915e-01	5.678779e-03	0.000000	2.175741e-01	2.175961e-01	2.176198e-01
_Total_Asset_Growth_Rate	2058.0	5.287663e+09	2.912615e+09	0.000000	4.315000e+09	6.225000e+09	7.220000e+09
_Net_Value_Growth_Rate	2058.0	5.189504e+06	2.077918e+08	0.000000	4.362833e-04	4.554170e-04	4.883758e-04
_Total_Asset_Return_Growth_Rate_Ratio	2058.0	2.641004e-01	2.415661e-03	0.251620	2.637383e-01	2.640161e-01	2.643097e-01
_Cash_Reinvestment_perc	2058.0	3.771970e-01	2.737311e-02	0.025828	3.707295e-01	3.789678e-01	3.855575e-01
_Current_Ratio	2058.0	1.336249e+06	6.061917e+07	0.000000	6.567062e-03	8.945370e-03	1.350542e-02
_Quick_Ratio	2058.0	2.775510e+07	4.448654e+08	0.000000	2.946399e-03	5.284241e-03	8.902983e-03
_Interest_Expense_Ratio	2058.0	6.312913e-01	6.785512e-03	0.525126	6.306116e-01	6.307999e-01	6.317437e-01
_Total_debt_to_Total_net_worth	2037.0	1.071429e+07	2.696960e+08	0.000000	3.924894e-03	7.270721e-03	1.306869e-02
_Long_term_fund_suitability_ratio_A	2058.0	8.973310e-03	3.485186e-02	0.004129	5.162031e-03	5.517000e-03	6.415301e-03
_Net_profit_before_tax_to_Paid_in_capital	2058.0	1.753994e-01	2.622348e-02	0.000000	1.658623e-01	1.745683e-01	1.844450e-01
_Total_Asset_Turnover	20	1.2864	1.0062	0.00	6.1469	1.0344	1.6791

count	mean	std	min	25%	50%	75%	max
	58.0	05e-01	16e-01	0000	27e-02	83e-01	60e-01
_Accounts_Receivable_Turnover	2058.0	4.159864e+07	5.047673e+08	0.000000	7.446260e-04	1.081432e-03	1.854463e-03
_Average_Collection_Days	2058.0	2.629786e+07	4.109967e+08	0.000000	3.576384e-03	6.001272e-03	8.638997e-03
_Inventory_Turnover_Rate_times	2058.0	2.030227e+09	3.077250e+09	0.000000	1.909297e-04	1.910000e+07	3.815000e+09
_Fixed_Assets_Turnover_Frequency	2058.0	1.230898e+09	2.649289e+09	0.000000	2.278950e-04	5.995245e-04	8.423224e-03
_Net_Worth_Turnover_Rate_times	2058.0	3.957710e-02	4.239591e-02	0.008871	2.048387e-02	2.870968e-02	4.435484e-02
_Operating_profit_per_person	2058.0	4.036693e-01	5.358970e-02	0.000000	3.913864e-01	3.950781e-01	4.008927e-01
_Allocation_rate_per_person	2058.0	5.725559e+06	1.979500e+08	0.000000	4.671612e-03	1.062969e-02	2.457491e-02
_Quick_Assets_to_Total_Assets	2058.0	3.421979e-01	2.103925e-01	0.000000	1.734827e-01	3.061276e-01	4.845435e-01
_Cash_to_Total_Assets	1962.0	7.993675e-02	9.862260e-02	0.000184	2.061909e-02	4.563187e-02	9.771301e-02
_Quick_Assets_to_Current_Liability	2058.0	1.190476e+07	3.122923e+08	0.000000	3.616304e-03	5.972976e-03	9.608533e-03
_Cash_to_Current_Liability	2058.0	9.282507e+07	7.851899e+08	0.000101	1.085476e-03	2.684338e-03	7.540535e-03
_Operating_Funds_to_Liability	2058.0	3.482338e-01	3.840302e-02	0.026274	3.377032e-01	3.450257e-01	3.541402e-01
_Inventory_to_Working_Capital	2058.	2.777491e-01	1.844394e-02	0.000000	2.770093e-01	2.772511e-01	2.777111e-01

count	mean	std	min	25%	50%	75%	max
	0						
_Inventory_to_Current_Liability	2058.0	5.786346e+07	6.278795e+08	0.000000	2.890842e-03	6.781166e-03	1.275116e-02
_Long_term_Liability_to_Current_Assets	2058.0	7.340107e+07	6.693526e+08	0.000000	0.000000e+00	2.587130e-03	1.049684e-02
_Retained_Earnings_to_Total_Assets	2058.0	9.303546e-01	2.976067e-02	0.000000	9.278868e-01	9.350756e-01	9.409371e-01
_Total_income_to_Total_expense	2058.0	2.357977e-03	4.644258e-04	0.000000	2.186964e-03	2.297452e-03	2.433146e-03
_Total_expense_to_Assets	2058.0	3.109208e-02	3.870042e-02	0.000853	1.270426e-02	2.086322e-02	3.530120e-02
_Current_Asset_Turnover_Rate	2058.0	1.273303e+09	2.839741e+09	0.000000	1.504698e-04	2.461660e-04	1.264005e-03
_Quick_Asset_Turnover_Rate	2058.0	2.571768e+09	3.453544e+09	0.000000	1.511758e-04	3.794085e-04	5.790000e+09
_Cash_Turnover_Rate	2058.0	2.653696e+09	2.821245e+09	0.000100	1.737418e-03	1.730000e+09	4.550000e+09
_Fixed_Assets_to_Assets	2058.0	4.042760e+06	1.834006e+08	0.000000	9.650577e-02	2.138107e-01	4.150287e-01
_Cash_Flow_to_Total_Assets	2058.0	6.442325e-01	4.505929e-02	0.000000	6.333645e-01	6.432462e-01	6.541577e-01
_Cash_Flow_to_Liability	2058.0	4.599747e-01	3.288112e-02	0.032583	4.574802e-01	4.593408e-01	4.617433e-01
_CFO_to_Assets	2058.0	5.797344e-01	6.375060e-02	0.000000	5.503790e-01	5.825431e-01	6.123215e-01
_Cash_Flow_to_Equity	2058.0	3.146292e-01	1.277967e-02	0.000000	3.127830e-01	3.146423e-01	3.165460e-01

count	mean	std	min	25%	50%	75%	max
_Current_Liability_to_Current_Assets	2044.0	3.935178e-02	4.797815e-02	0.000000	2.177539e-02	3.265229e-02	4.394684e-02
_Liability_Assets_Flag	2058.0	3.401361e-03	5.823606e-02	0.000000	0.000000e+00	0.000000e+00	0.000000e+00
_Total_assets_to_GNP_price	2058.0	2.779397e+07	4.717714e+08	0.000000	9.124052e-04	2.479550e-03	7.004449e-03
_No_credit_Interval	2058.0	6.236856e-01	1.163052e-02	0.408682	6.233274e-01	6.237496e-01	6.240452e-01
_Degree_of_Financial_Leverage_DFL	2058.0	2.785248e-02	1.383854e-02	0.012845	2.677558e-02	2.681466e-02	2.702943e-02
_Interest_Coverage_Ratio_Interest_expense_to_EBIT	2058.0	5.654355e-01	1.153538e-02	0.172065	5.651580e-01	5.653149e-01	5.662324e-01
_Net_Income_Flag	2058.0	1.000000e+00	0.000000e+00	1.000000	1.000000e+00	1.000000e+00	1.000000e+00
_Equity_to_Liability	2058.0	4.252852e-02	5.952518e-02	0.003946	2.040787e-02	2.846004e-02	4.343255e-02
Default	2058.0	1.068999e-01	3.090610e-01	0.000000	0.000000e+00	0.000000e+00	0.000000e+00

1. **Count**: Indicates the number of observations for each variable. All variables have 2058 observations except for "*Cash_Flow_Per_Share" which has 1891 observations and* "Total_debt_to_Total_net_worth" which has 2037 observations.

2. **Mean**: Represents the average value of each variable across all observations.

3. **Standard Deviation (std)**: Gives a measure of the dispersion or spread of the data from the mean. Higher standard deviations indicate greater variability in the data.

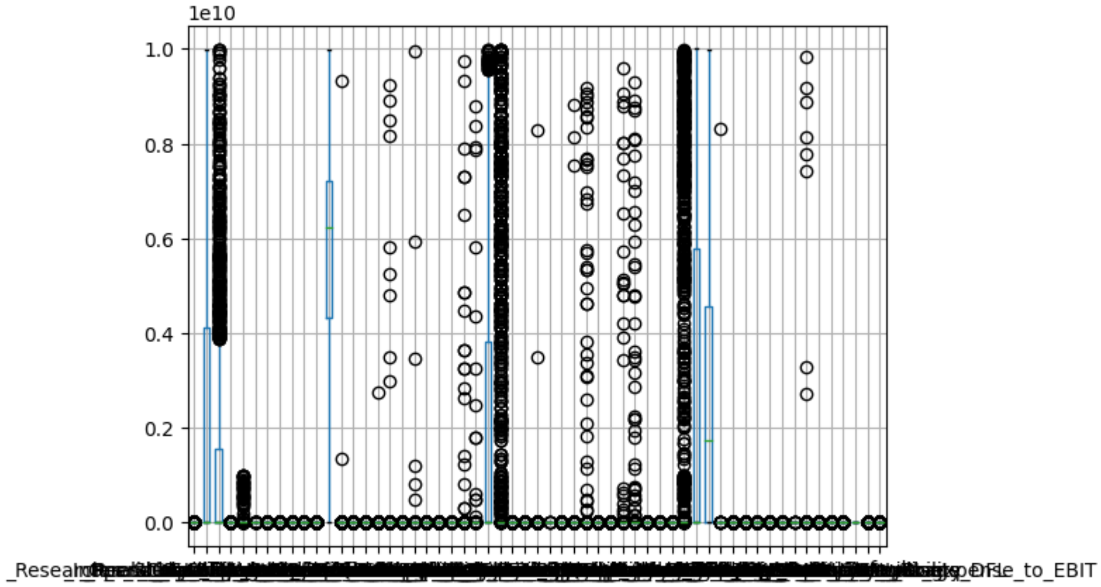
4. **Min and Max**: Show the minimum and maximum values observed for each variable, indicating the range of values present in the dataset.

5. **Percentiles (25%, 50%, 75%)**: Represent values below which a given percentage of observations fall. The 50th percentile (median) is the value below which 50% of the observations fall.

From these statistics, you can make various assessments about the dataset:

- For instance, "_Operating_Expense_Rate" has a very high mean compared to other variables, indicating that, on average, operating expenses are substantial relative to other financial metrics.
- "_Tax_rate_A" has a mean of 0.114, suggesting that the average tax rate for the companies in the dataset is around 11.4%.
- "*Current_Ratio" and "*Quick_Ratio" have significantly high standard deviations compared to their means, indicating a wide variation in these ratios across the dataset.
- "_Net_Income_Flag" has a mean of 1.0 and a standard deviation of 0.0, suggesting it may be a binary indicator variable with all observations being 1.

PART A: Outlier Treatment



- The graph shows too many outliers , the treatment of outliers depends on the origianizational goals and what they are looking for
- Outliers are data points that significantly differ from other observations in a dataset. They can arise due to various reasons such as measurement errors, data entry mistakes, natural variations in the data, or rare events. Outliers can distort statistical analyses and machine learning models, leading to inaccurate results and conclusions. Therefore, it's essential to identify and appropriately handle outliers.

  Here are common methods to treat outliers:

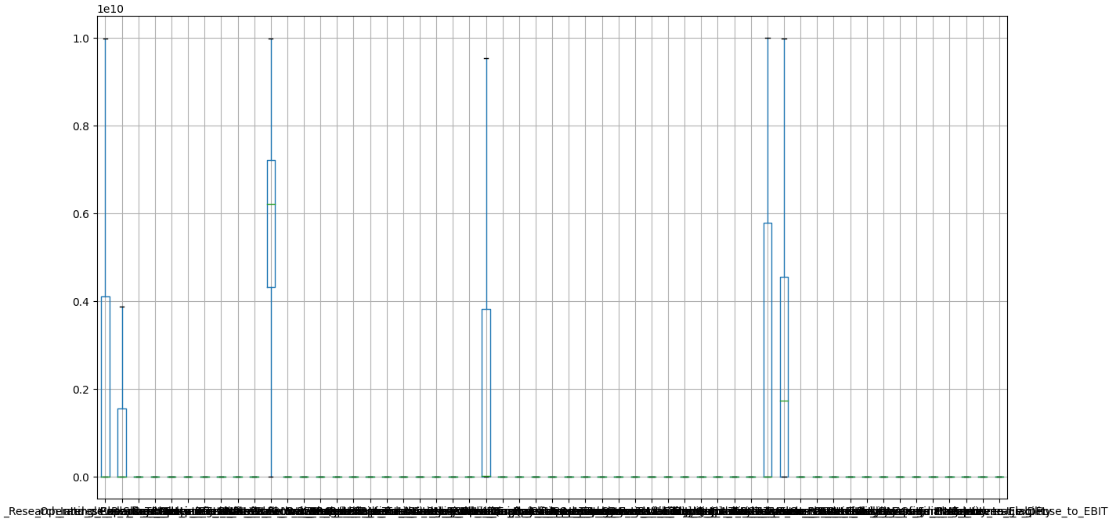
1. **Data Visualization**: Plot the data using histograms, box plots, scatter plots, or QQ plots to visually identify outliers. This initial step helps to understand the distribution and detect any extreme values.

2. **Statistical Methods**:

   - **Z-Score**: Calculate the Z-score for each data point, which measures the number of standard deviations away from the mean. Remove data points with Z-scores exceeding a certain threshold (commonly |Z-score| > 3).

   - **Interquartile Range (IQR)**: Calculate the IQR, which is the range between the first (Q1) and third (Q3) quartiles. Remove data points that fall below Q1 - 1.5 * IQR or above Q3 + 1.5 * IQR.

- After treating the outliers with IQR method we can see that the box plot doesn't show any outliers



PART A: Missing Value Treatment

To treat missing values in Python:

1. Use the ".fillna()" method in pandas to fill missing values with a specified value, such as the mean or median of the column.

2. Alternatively, drop rows or columns containing missing values using ".dropna()" method in pandas.

3. For more complex cases, use imputation techniques such as mean, median, or mode imputation for numerical data or using predictive models like KNN imputation.

4. Utilize libraries like scikit-learn's "simpleImputer" for a more systematic approach to impute missing values based on different strategies.

```
_Operating_Expense_Rate                                0
_Research_and_development_expense_rate                 0
_Cash_flow_rate                                        0
_Interest_bearing_debt_interest_rate                   0
_Tax_rate_A                                            0
_Cash_Flow_Per_Share                                 167
_Per_Share_Net_profit_before_tax_Yuan_                 0
_Realized_Sales_Gross_Profit_Growth_Rate               0
_Operating_Profit_Growth_Rate                          0
_Continuous_Net_Profit_Growth_Rate                     0
_Total_Asset_Growth_Rate                               0
_Net_Value_Growth_Rate                                 0
_Total_Asset_Return_Growth_Rate_Ratio                  0
_Cash_Reinvestment_perc                                0
_Current_Ratio                                         0
_Quick_Ratio                                           0
_Interest_Expense_Ratio                                0
_Total_debt_to_Total_net_worth                        21
_Long_term_fund_suitability_ratio_A                    0
_Net_profit_before_tax_to_Paid_in_capital              0
_Total_Asset_Turnover                                  0
_Accounts_Receivable_Turnover                          0
_Average_Collection_Days                               0
_Inventory_Turnover_Rate_times                         0
_Fixed_Assets_Turnover_Frequency                       0
_Net_Worth_Turnover_Rate_times                         0
_Operating_profit_per_person                           0
_Allocation_rate_per_person                            0
_Quick_Assets_to_Total_Assets                          0
_Cash_to_Total_Assets                                 96
_Quick_Assets_to_Current_Liability                     0
_Cash_to_Current_Liability                             0
_Operating_Funds_to_Liability                          0
_Inventory_to_Working_Capital                          0
_Inventory_to_Current_Liability                        0
_Long_term_Liability_to_Current_Assets                 0
_Retained_Earnings_to_Total_Assets                     0
_Total_income_to_Total_expense                         0
_Total_expense_to_Assets                               0
_Current_Asset_Turnover_Rate                           0
_Quick_Asset_Turnover_Rate                             0
_Cash_Turnover_Rate                                    0
_Fixed_Assets_to_Assets                                0
_Cash_Flow_to_Total_Assets                             0
_Cash_Flow_to_Liability                                0
_CFO_to_Assets                                         0
_Cash_Flow_to_Equity                                   0
```

```
_Current_Liability_to_Current_Assets                  14
_Liability_Assets_Flag                                 0
_Total_assets_to_GNP_price                             0
_No_credit_Interval                                    0
_Degree_of_Financial_Leverage_DFL                      0
_Interest_Coverage_Ratio_Interest_expense_to_EBIT      0
_Net_Income_Flag                                       0
_Equity_to_Liability                                   0
```

- The columns Cash_Flow_Per_Share", "*Total_debt_to_Total_net_worth",* "Cash_to_Total_Assets", "_Current_Liability_to_Current_Assets" have missing values

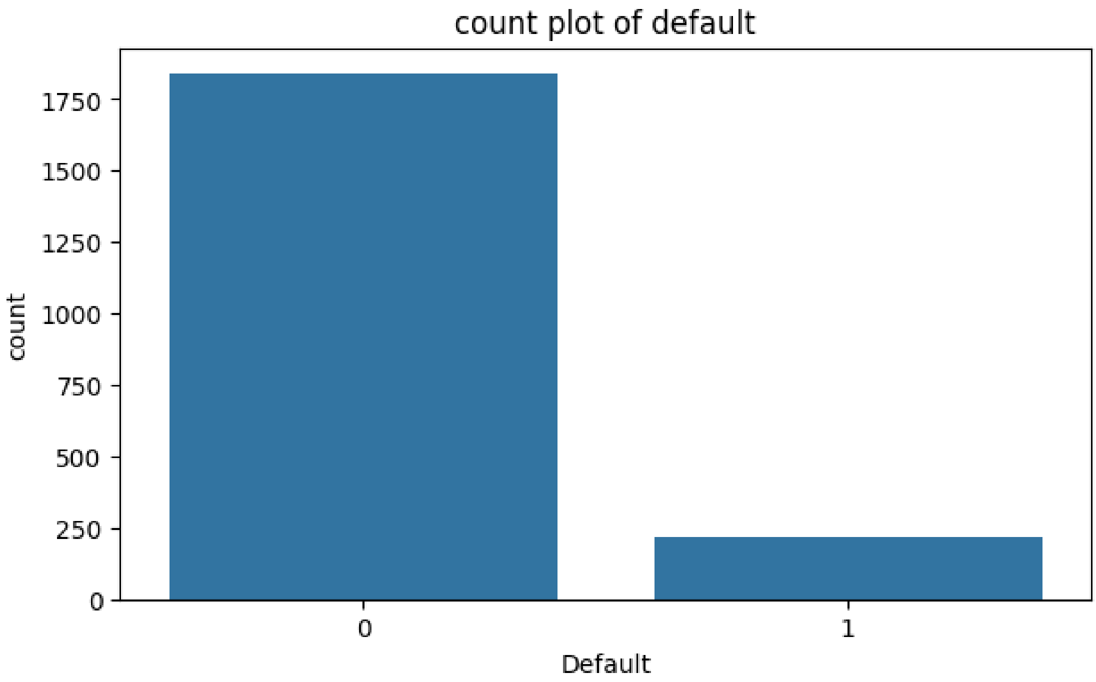
- after treating missings values

```
_Operating_Expense_Rate                              0
_Research_and_development_expense_rate               0
_Cash_flow_rate                                      0
_Interest_bearing_debt_interest_rate                 0
_Tax_rate_A                                          0
_Cash_Flow_Per_Share                                 0
_Per_Share_Net_profit_before_tax_Yuan_               0
_Realized_Sales_Gross_Profit_Growth_Rate             0
_Operating_Profit_Growth_Rate                        0
_Continuous_Net_Profit_Growth_Rate                   0
_Total_Asset_Growth_Rate                             0
_Net_Value_Growth_Rate                               0
_Total_Asset_Return_Growth_Rate_Ratio                0
_Cash_Reinvestment_perc                              0
_Current_Ratio                                       0
_Quick_Ratio                                         0
_Interest_Expense_Ratio                              0
_Total_debt_to_Total_net_worth                       0
_Long_term_fund_suitability_ratio_A                  0
_Net_profit_before_tax_to_Paid_in_capital            0
_Total_Asset_Turnover                                0
_Accounts_Receivable_Turnover                        0
_Average_Collection_Days                             0
_Inventory_Turnover_Rate_times                       0
_Fixed_Assets_Turnover_Frequency                     0
_Net_Worth_Turnover_Rate_times                       0
_Operating_profit_per_person                         0
_Allocation_rate_per_person                          0
_Quick_Assets_to_Total_Assets                        0
_Cash_to_Total_Assets                                0
_Quick_Assets_to_Current_Liability                   0
_Cash_to_Current_Liability                           0
_Operating_Funds_to_Liability                        0
_Inventory_to_Working_Capital                        0
_Inventory_to_Current_Liability                      0
_Long_term_Liability_to_Current_Assets               0
```

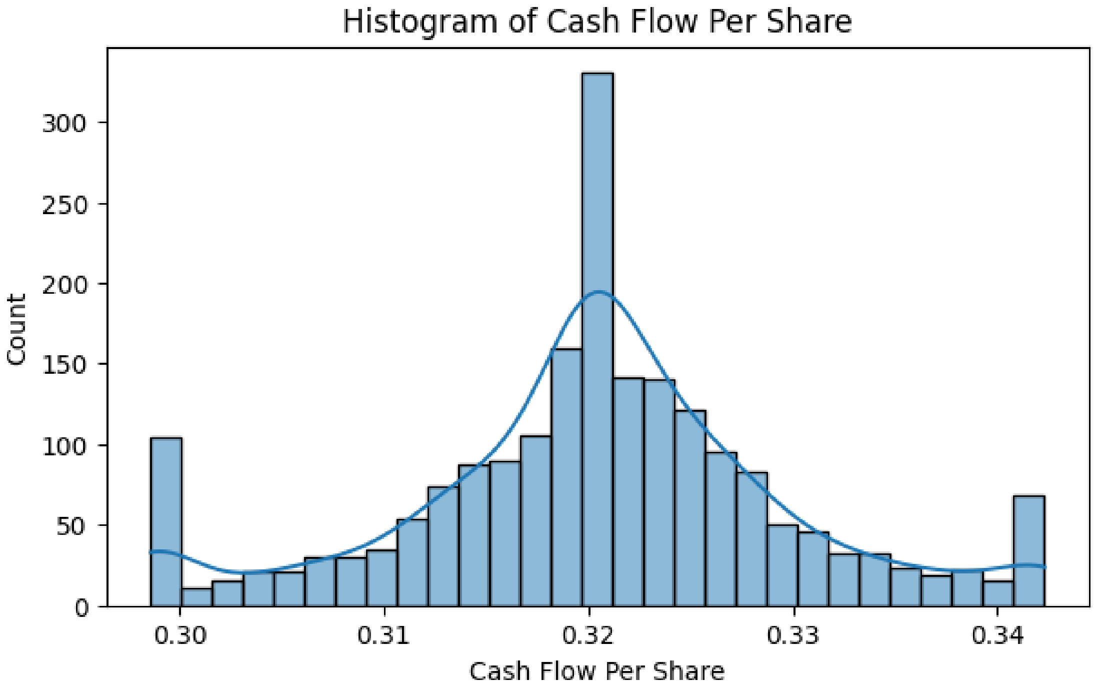
```
_Retained_Earnings_to_Total_Assets                    0
_Total_income_to_Total_expense                        0
_Total_expense_to_Assets                              0
_Current_Asset_Turnover_Rate                          0
_Quick_Asset_Turnover_Rate                            0
_Cash_Turnover_Rate                                   0
_Fixed_Assets_to_Assets                               0
_Cash_Flow_to_Total_Assets                            0
_Cash_Flow_to_Liability                               0
_CFO_to_Assets                                        0
_Cash_Flow_to_Equity                                  0
_Current_Liability_to_Current_Assets                  0
_Liability_Assets_Flag                                0
_Total_assets_to_GNP_price                            0
_No_credit_Interval                                   0
_Degree_of_Financial_Leverage_DFL                     0
_Interest_Coverage_Ratio_Interest_expense_to_EBIT     0
_Net_Income_Flag                                      0
_Equity_to_Liability                                  0
```

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

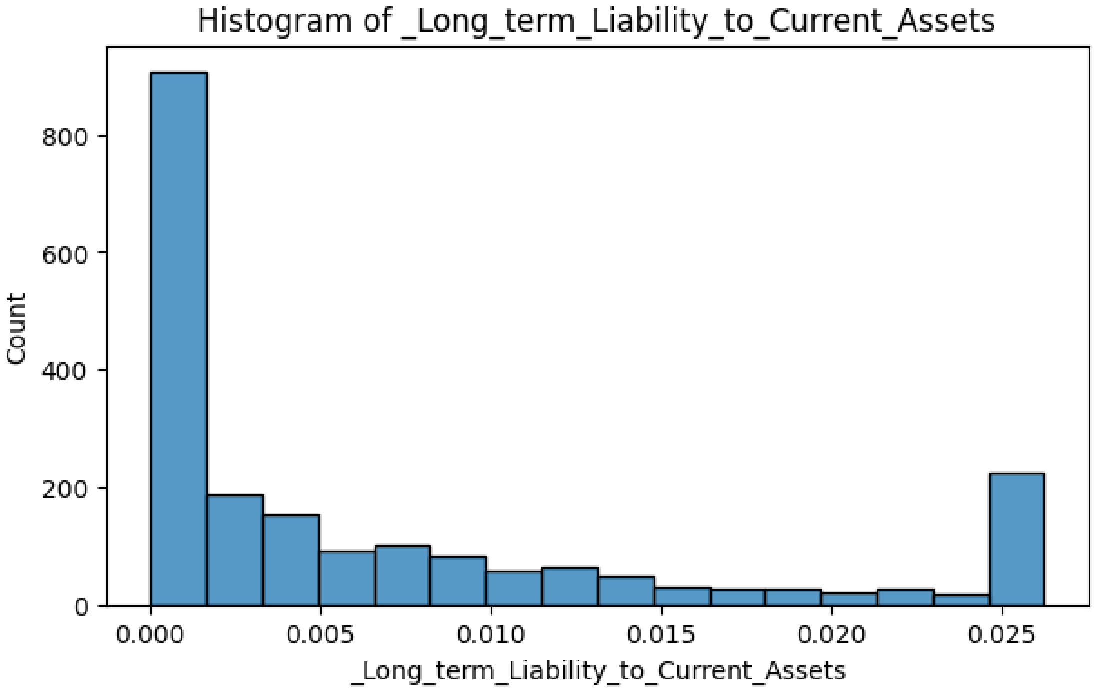
Univariate analysis:-



- The count plot shows the number of defaults that have occurred in two categories. The most frequent category, with a count of 1750, is labeled "0". The least frequent category, with a count of 250, is labeled "1".

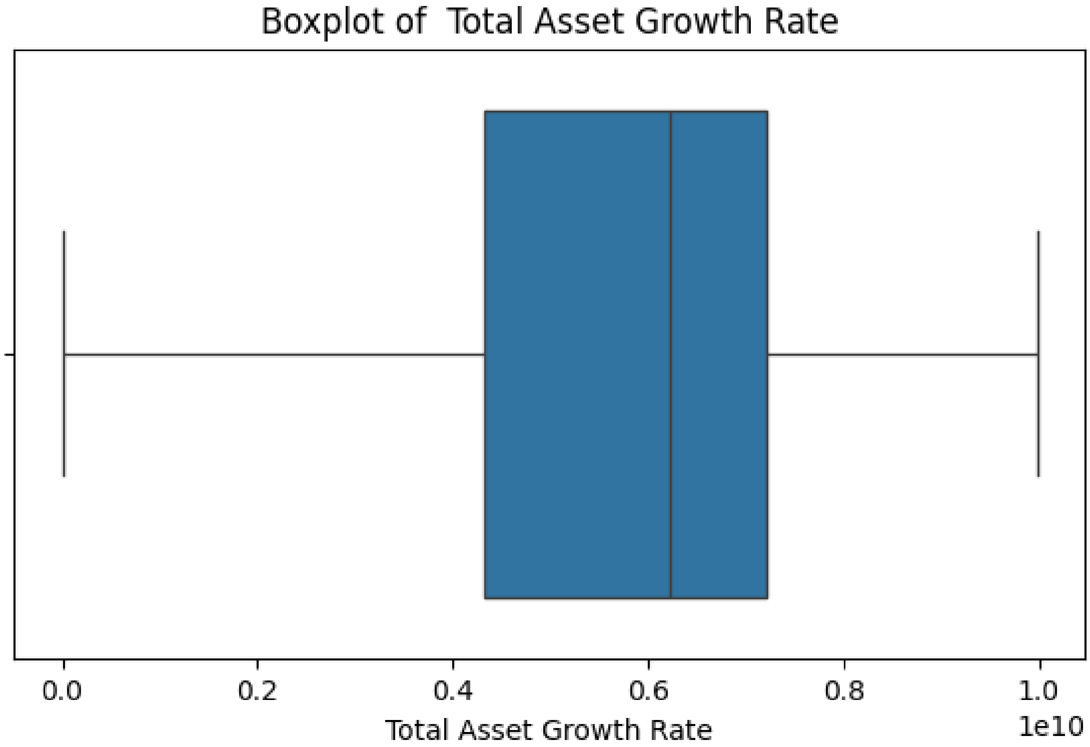


- A histogram of cash flow per share. The x-axis of the histogram shows the cash flow per share, while the y-axis shows the number of companies that have had that cash flow per share. The histogram shows that the most common cash flow per share is around $0.32. There are fewer companies with a cash flow per share that is much lower or much higher than $0.32.



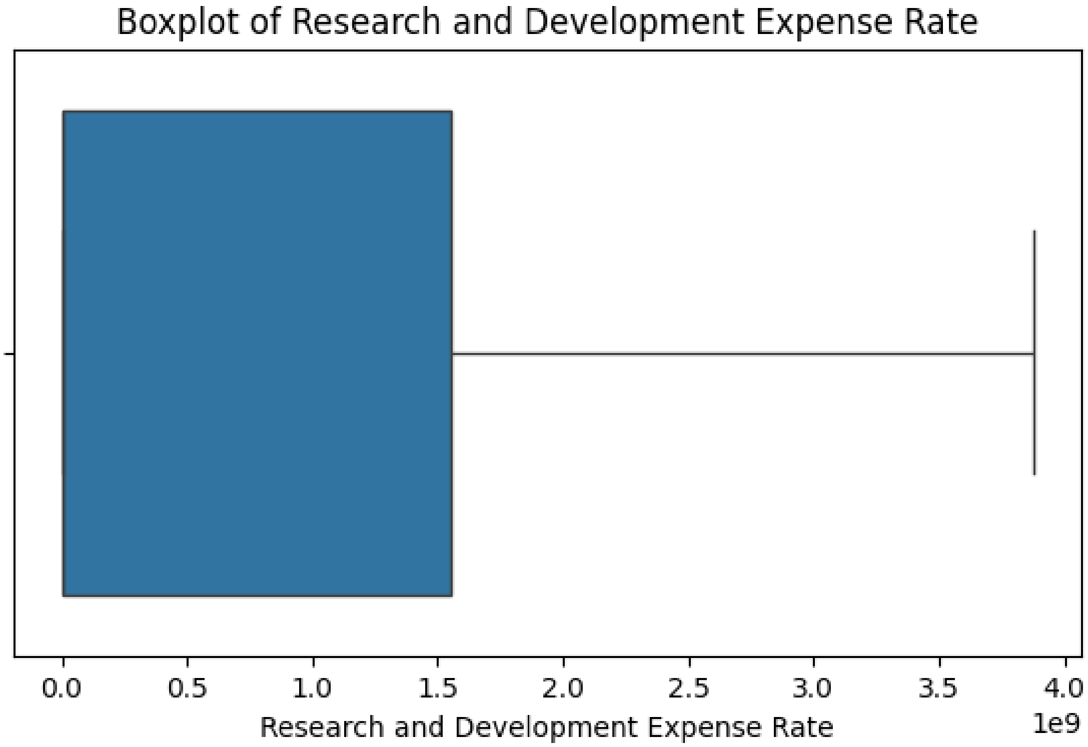
- A histogram of long-term liability to current assets ratio. The x-axis of the histogram shows the ratio, while the y-axis shows the number of businesses that have that ratio.

The histogram shows that the most common long-term liability to current asset ratio falls between 0.005 and 0.010. There are fewer businesses with a ratio that is much lower or much higher than this range.



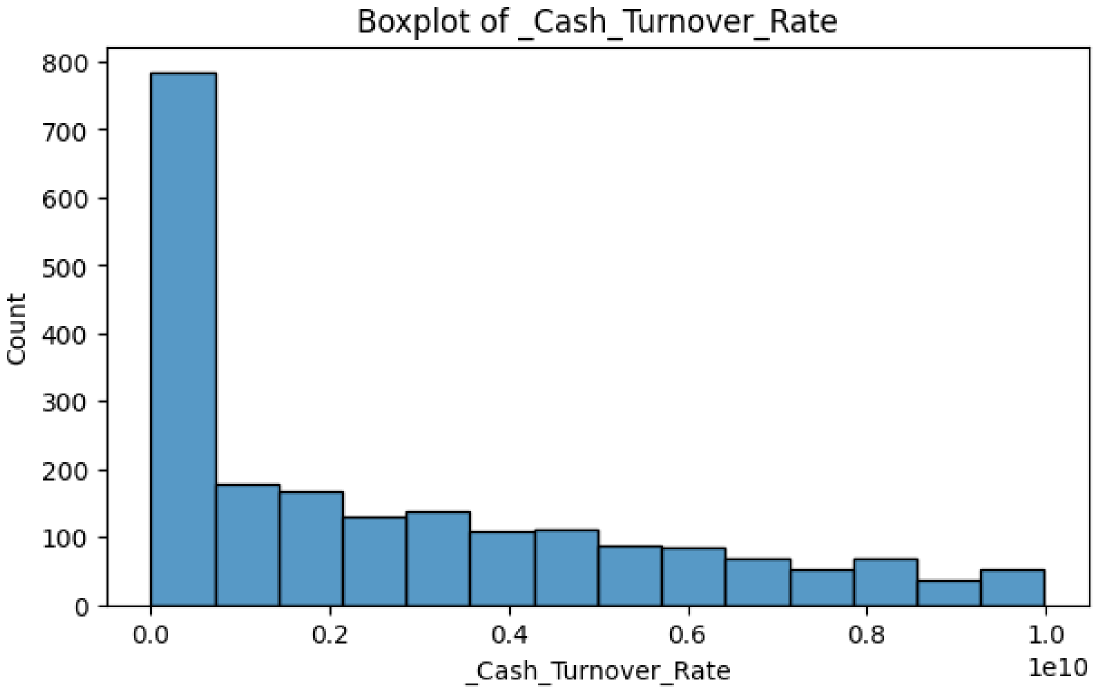
- The box plot shows the distribution of the total asset growth rate for a company over a certain period of time. The box in the center of the plot represents the middle 50% of the data. The line in the middle of the box is the median, which is the 50th percentile. The whiskers extend from the top and bottom of the box to the most extreme data points that are not considered outliers. Outliers are represented by individual points beyond the whiskers.

  In this specific box plot, the median total asset growth rate is between 0% and 10%. The whiskers extend to -10% and 50%. This means that the middle 50% of the companies in the data set experienced a total asset growth rate between 0% and 10% over the specified time period. There are also some companies that experienced a total asset growth rate that falls outside the range of -10% to 50%.



- boxplot of research and development expense rate. The y-axis of the plot shows the research and development expense rate, while the x-axis is not labeled.

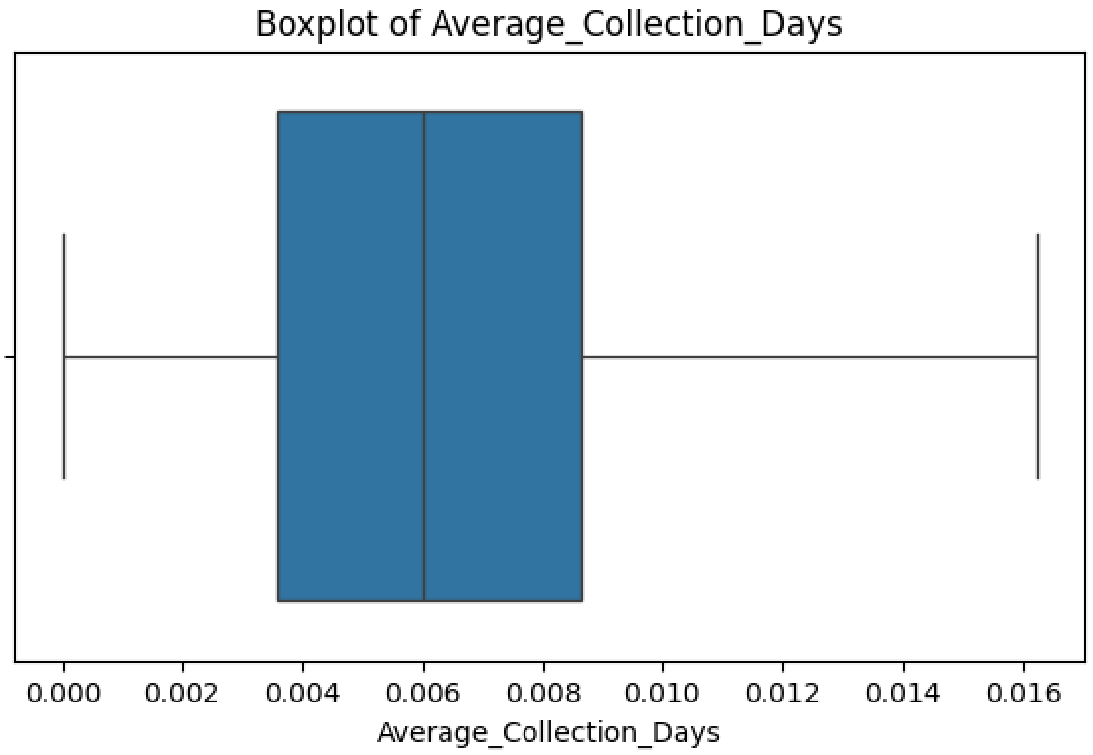
The box in the center of the plot represents the middle 50% of the data. The line in the middle of the box is the median, which is the 50th percentile. The whiskers extend from the top and bottom of the box to the most extreme data points that are not considered outliers. Outliers are represented by individual points beyond the whiskers.



- The cash turnover rate is a measure of how efficiently a company uses its cash. A higher cash turnover rate indicates that a company is collecting its receivables and paying its bills quickly.

  The box in the center of the plot represents the middle 50% of the data. The line in the middle of the box is the median, which is the 50th percentile. The whiskers extend from the top and bottom of the box to the most extreme data points that are not considered outliers. Outliers are represented by individual points beyond the whiskers.

  In this specific box plot, the median cash turnover rate is between 0.4 and 0.6. The whiskers extend to 0.2 and 0.8. This means that the middle 50% of the companies in the data set experienced a cash turnover rate between 0.4 and 0.6. There are also some companies that experienced a cash turnover rate that falls outside the range of 0.2 to 0.8.

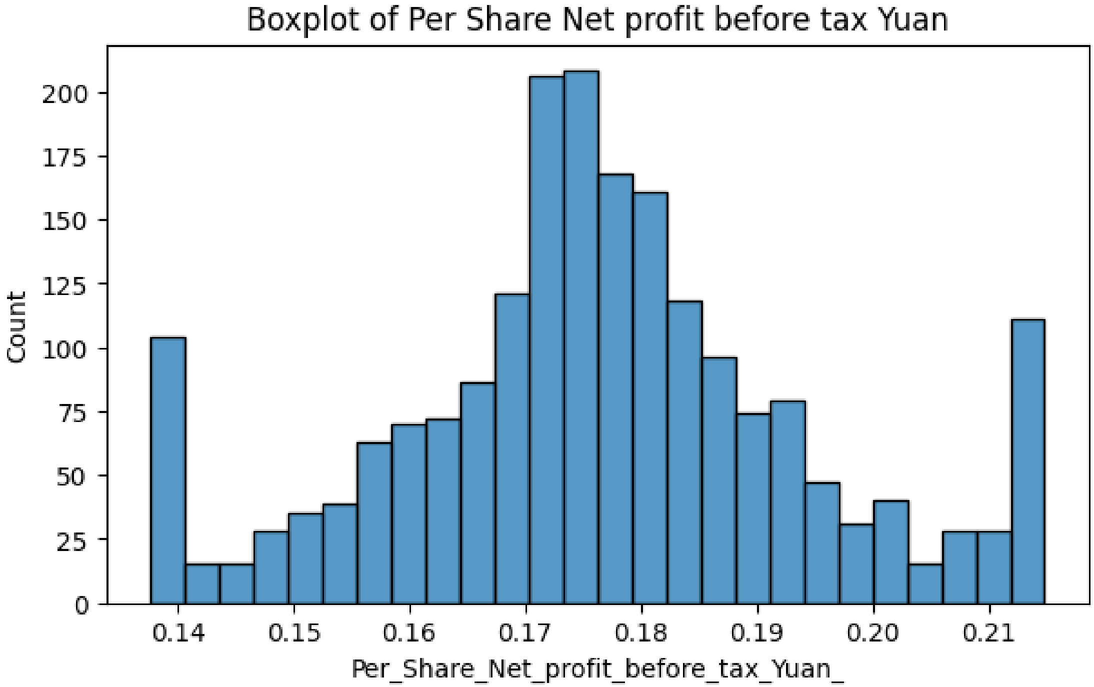


- The y-axis of the plot shows the number of days, while the x-axis is not labeled. The box in the center of the plot represents the middle 50% of the data. The line in the middle of the box is the median, which is the 50th percentile. The whiskers extend from the top and bottom of the box to the most extreme data points that are not considered outliers. Outliers are represented by individual points beyond the whiskers.

  Based on the boxplot, we can see the following:

  - The median number of average collection days is around 0.008.
  - The middle 50% of the data falls between approximately 0.006 and 0.010 days.
  - There are some outliers that fall outside the range of 0.002 and 0.014 days.

    pen_spark

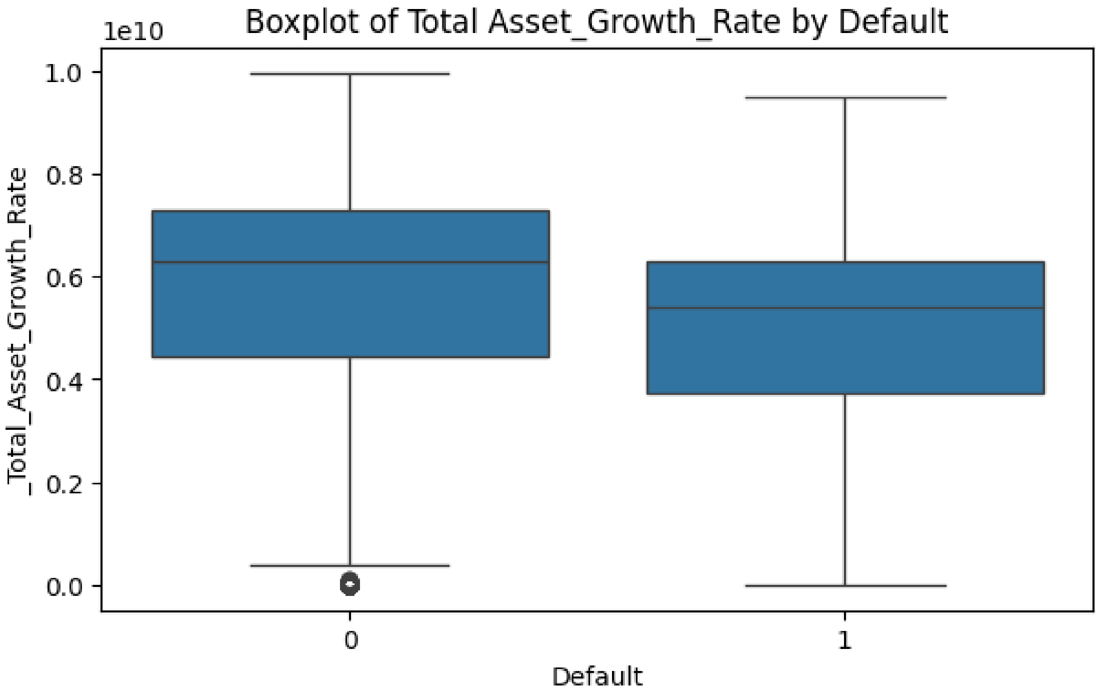


- The box in the center of the plot represents the middle 50% of the data. The line in the middle of the box is the median, which is the 50th percentile. The whiskers extend from the top and bottom of the box to the most extreme data points that are not considered outliers. Outliers are represented by individual points beyond the whiskers.

  Based on the boxplot, we can see the following:

  - The median net profit before tax yuan is between 0.15 and 0.16.
  - The middle 50% of the data falls between approximately 0.14 and 0.17 yuan.
  - There are some outliers that fall outside the range of 0.10 and 0.21 yuan.

Bi-varient analysi

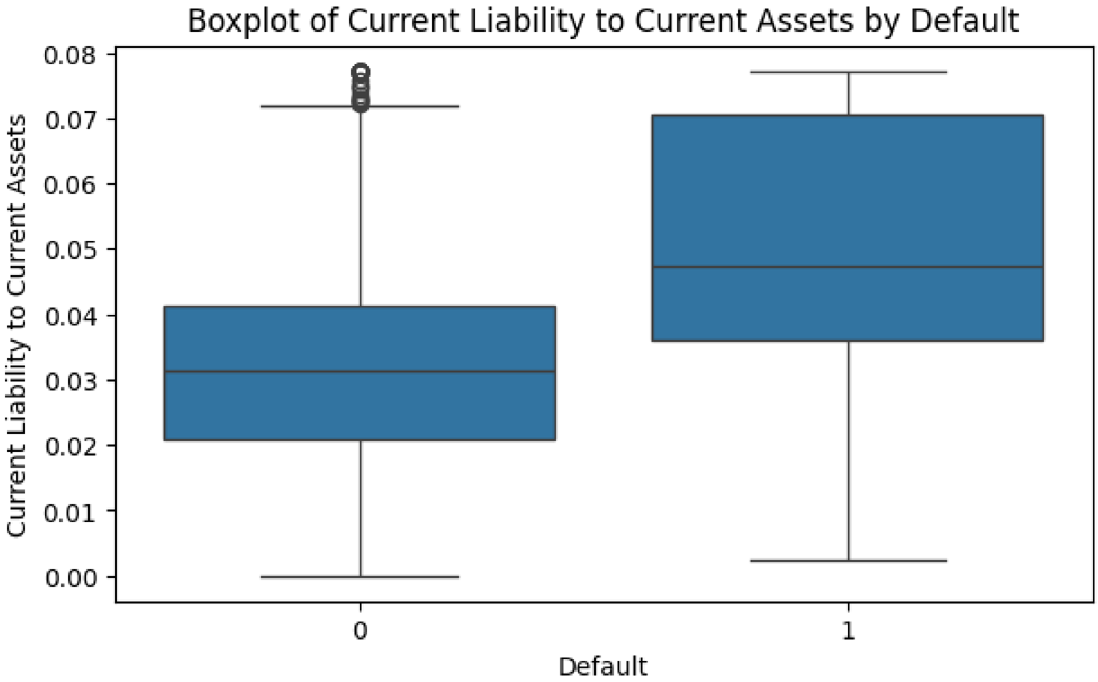


- The default setting on the boxplot means that the data is split into two groups (0 and 1) based on an unspecified variable. In this case, the boxplot shows that the total asset growth rate for companies in the "default" group is significantly lower than the total asset growth rate for companies in the "non-default" group.

The median total asset growth rate for companies in the "default" group is around -10%. The whiskers extend down to -50% and up to 0%. This means that the middle 50% of the companies in the "default" group experienced a total asset growth rate between -10% and 0% over the specified time period. There are also some companies in the "default" group that experienced a total asset growth rate that falls outside the range of -50% to 0%.

The median total asset growth rate for companies in the "non-default" group is between 20% and 30%. The whiskers extend to 10% and 50%. This means that the middle 50% of the companies in the "non-default" group experienced a total asset growth rate between 20% and 30% over the specified time period. There are also some companies in the "non-default" group that experienced a total asset growth rate that falls outside the range of 10% to 50%.

pen_spark

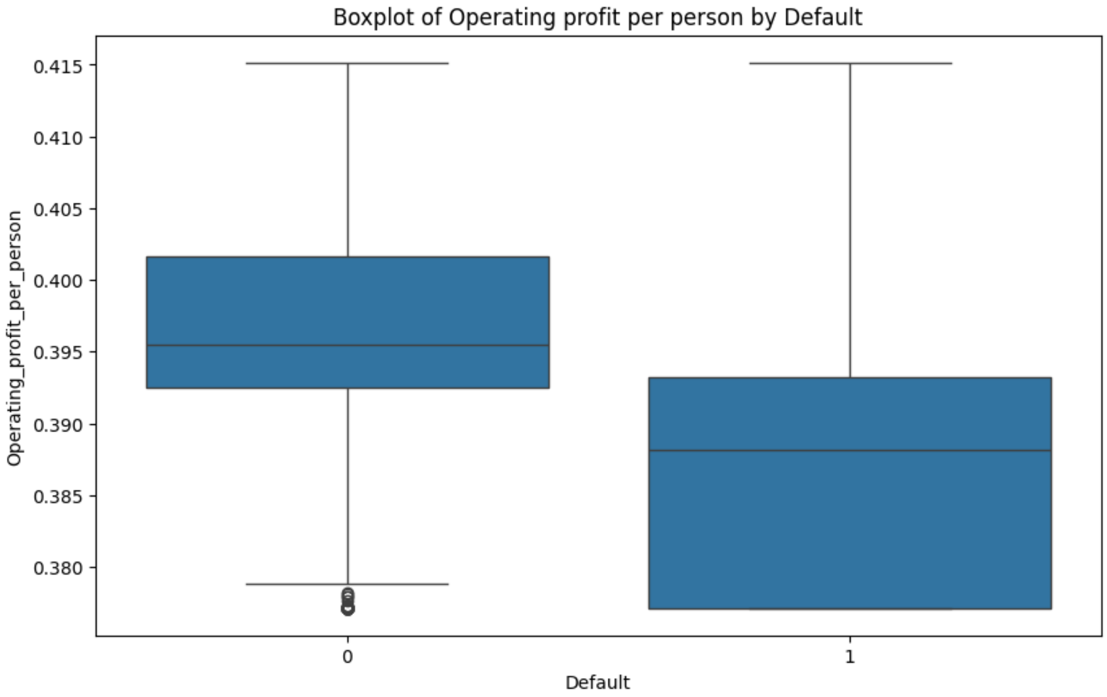


- The box in the center of the plot represents the middle 50% of the data for each group. The line in the middle of the box is the median, which is the 50th percentile. The whiskers extend from the top and bottom of the box to the most extreme data points that are not considered outliers. Outliers are represented by individual points beyond the whiskers.

Based on the boxplot, we can see the following:

- The median current liability to current asset ratio is higher for companies that defaulted on their loans (around 0.6) than for companies that did not default (around 0.2).
- The data is more spread out for companies that defaulted on their loans than for companies that did not default. This means that there is a larger variation in the current liability to current asset ratio among companies that defaulted on their loans.
- There are some outliers for both groups.

In conclusion, the data suggests that there is a positive correlation between current liability to current asset ratio and loan default. This means that companies that have a higher current liability to current asset ratio are more likely to default on their loans.



- The x-axis of the plot shows default (0) or non-default (1), and the y-axis shows operating profit per person. The box in the center of the plot represents the middle 50% of the data for each group. The line in the middle of the box is the median, which is the 50th percentile. The whiskers extend from the top and bottom of the box to the most extreme data points that are not considered outliers. Outliers are represented by individual points beyond the whiskers.

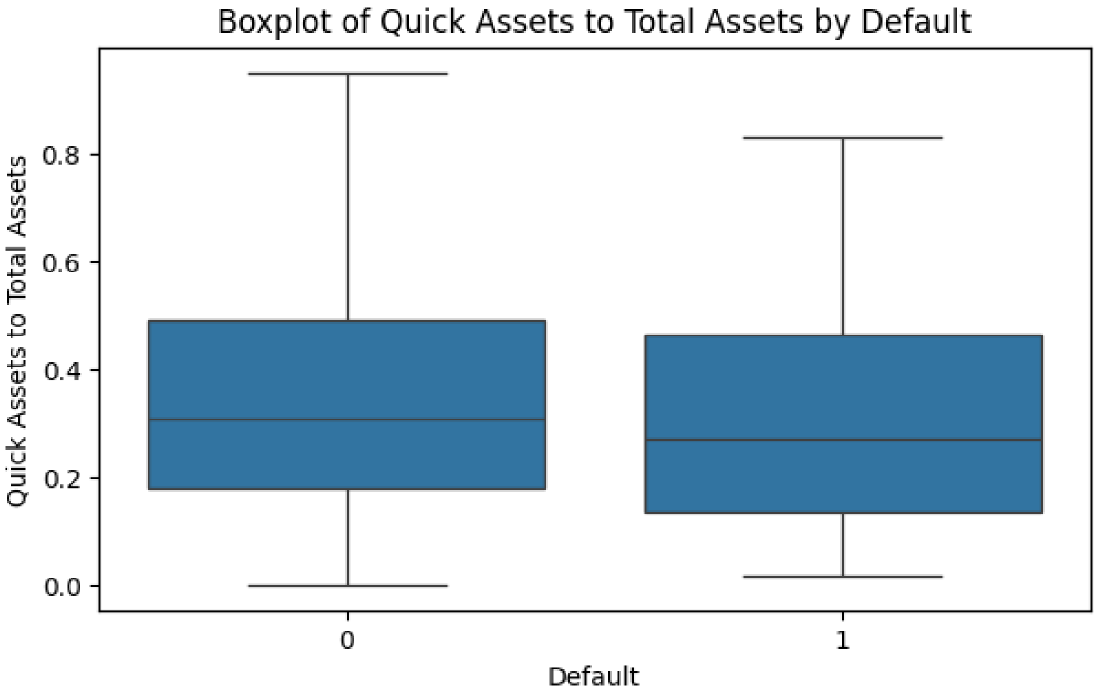
  Based on the boxplot, we can see the following:

  - The median operating profit per person is lower for companies that defaulted on their loans (around 0.38) than for companies that did not default (around 0.40).

  - The data is more spread out for companies that defaulted on their loans than for companies that did not default. This means that there is a larger variation in the operating profit per person among companies that defaulted on their loans.

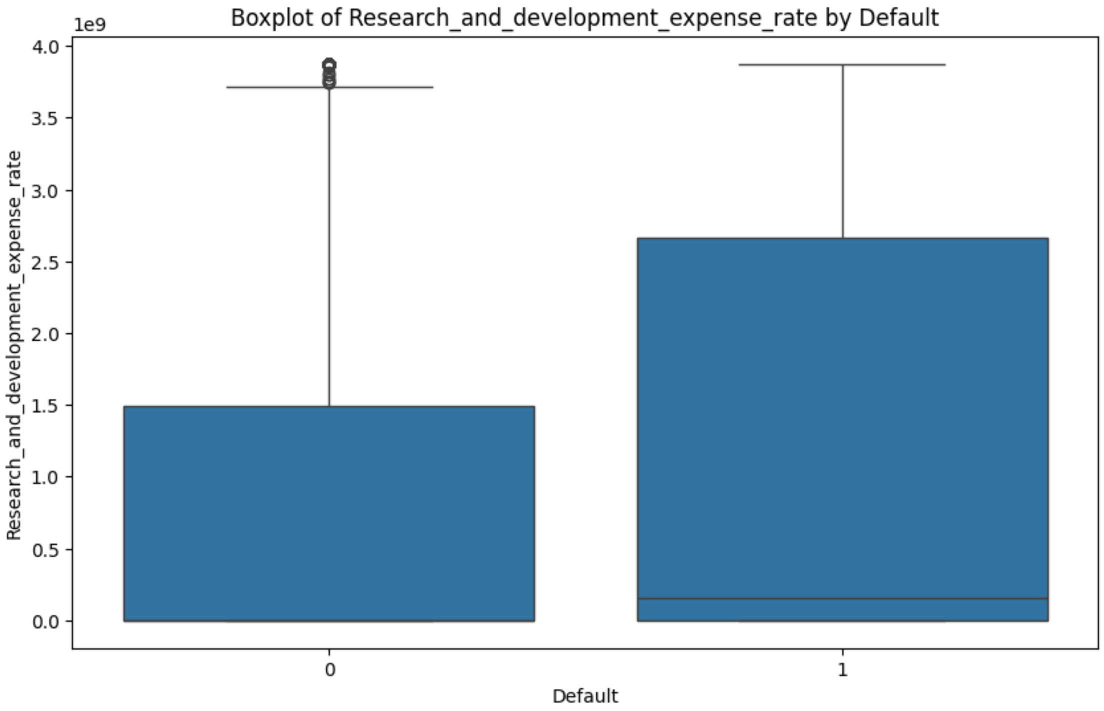
  - There are some outliers for both groups.

  In conclusion, the data suggests that there is a negative correlation between operating profit per person and loan default. This means that companies that have a lower operating profit per person are more likely to default on their loans.



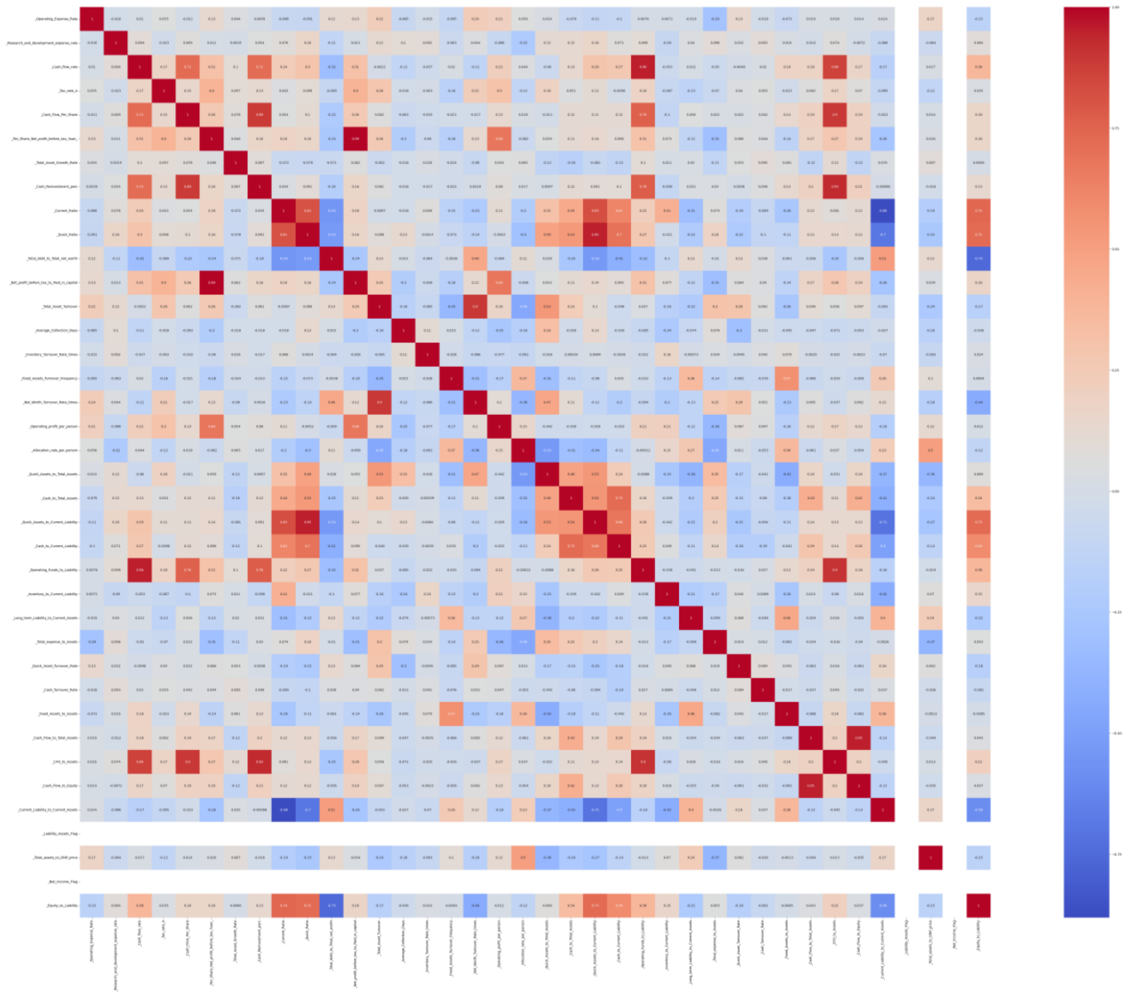
- Based on the boxplot, we can see the following:
    - The median quick assets to total assets ratio is lower for companies that defaulted on their loans (around 0.2) than for companies that did not default (around 0.4). This suggests that companies that defaulted on their loans had a lower proportion of liquid assets to total assets than companies that did not default.
    - The data is more spread out for companies that defaulted on their loans than for companies that did not default. This means that there is a larger variation in the quick assets to total assets ratio among companies that defaulted on their loans.
    - There are some outliers for both groups.

In conclusion, the data suggests that there is a negative correlation between quick assets to total assets ratio and loan default. This means that companies that have a lower quick assets to total assets ratio are more likely to default on their loans.



- The box in the center of each region on the plot represents the middle 50% of the data for that region. The line in the middle of the box is the median, which is the 50th percentile. The whiskers extend from the top and bottom of the box to the most extreme data points that are not considered outliers. Outliers are represented by individual points beyond the whiskers.

Unfortunately, since the scale of the y-axis is not labeled, it is difficult to say what the specific values on the y-axis represent. For example, it is not possible to say whether a research and development expense rate of 0.2 is high or low.

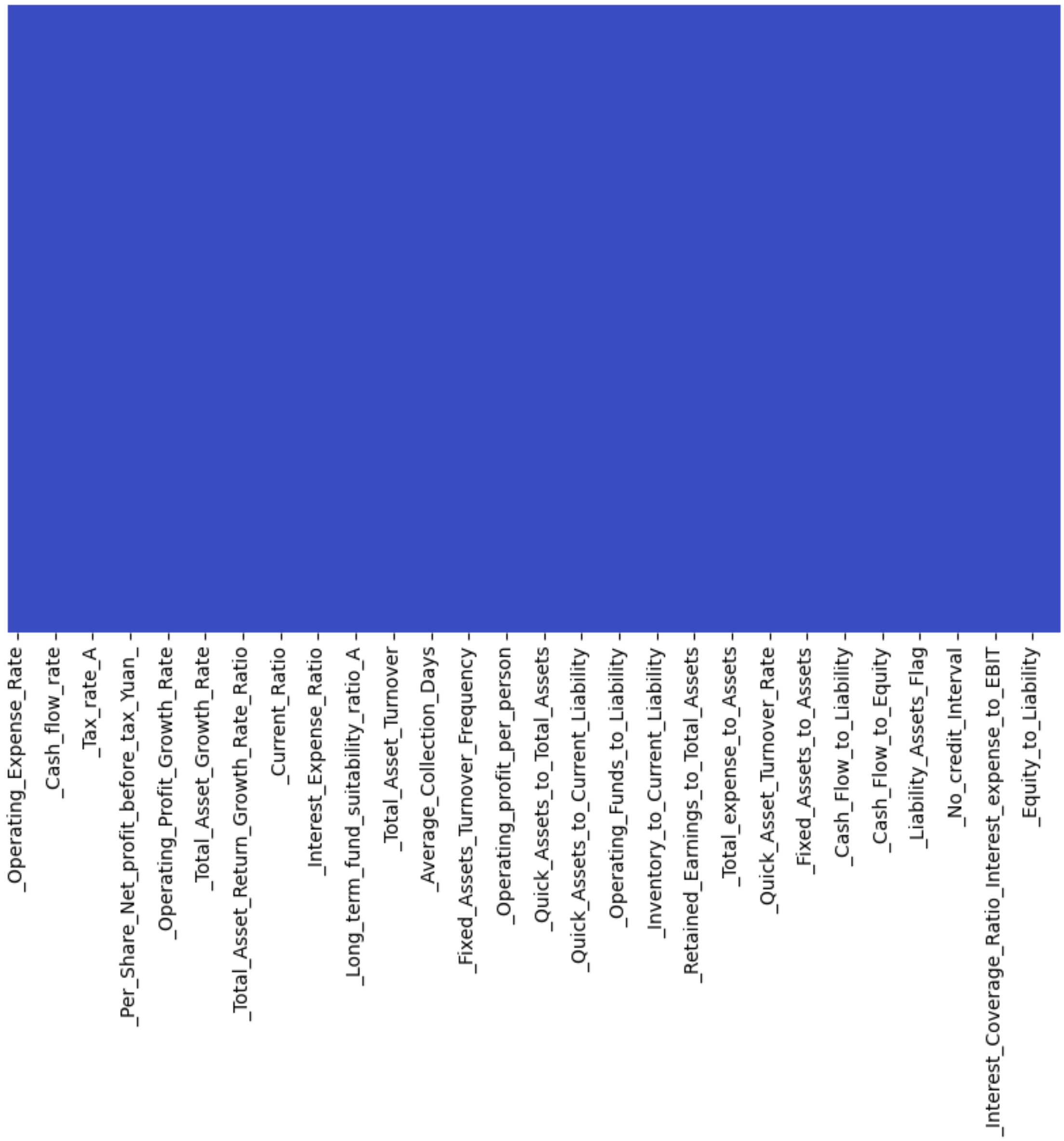


- **Color legend:** The legend on the right indicates that **red corresponds to hotter temperatures**, while **blue corresponds to colder temperatures**. This is a common color scheme used in temperature visualizations.

- **Temperature distribution:** Generally, the areas towards the bottom of the map (closer to the equator) appear red, indicating warmer temperatures. Conversely, areas towards the top of the map (closer to the poles) appear blue, indicating colder temperatures. This reflects the planet's temperature zones: warmer at the equator and colder at the poles.

- **Land vs. Ocean temperatures:** Oceans tend to have a moderating effect on temperature, so we can expect that land masses will generally show more extreme temperatures (both hotter and colder) than the oceans. This is because land heats and cools more quickly than water. On the heatmap, continents might show a wider range of colors compared to the oceans.

Overall, the heatmap provides a quick visual representation of the average temperature distribution on Earth. Since this is a global view, it is difficult to make out specific details or temperature values without zooming in or referring to a legend with specific temperature ranges.



- This plot above shows that there are no null values in the data after the treatment .

Variation Inflation factor

The Variance Inflation Factor (VIF) is a measure used in regression analysis to assess the severity of multicollinearity in a set of predictor variables. Multicollinearity occurs when two or more predictor variables in a regression model are highly correlated with each other, which can lead to issues with the interpretation of the model's coefficients.

The VIF quantifies the extent to which the variance of an estimated regression coefficient is increased due to multicollinearity. Specifically, the VIF for a predictor variable is calculated as the ratio of the variance of the coefficient estimate when that predictor is included in the model to the variance of the coefficient estimate when that predictor is excluded from the model.

Mathematically, the VIF for the ith predictor variable can be expressed as:

VIF⍰=11−⍰⍰2VIF*i*=1−*R****i*21

Where ⍰⍰2*R****i*2 is the coefficient of determination obtained by regressing the ith predictor variable on all other predictor variables in the model.

The P-values have the same interpretation, If the P_value is greeater than .05 then the variable is insignificant, and if the P value is less than .05 then the value is significant**

VIF talks about how good an independed varibales can be explained as a linear combination of other independednt variable, if the VIF is more than 5 then it means its more or less compensated by the other variable

variables	VIF
11	_Net_profit_before_tax_to_Paid_in_capital
5	*Per_Share_Net_profit_before_tax_Yuan*
31	_CFO_to_Assets
23	_Operating_Funds_to_Liability
21	_Quick_Assets_to_Current_Liability
2	_Cash_flow_rate
16	_Net_Worth_Turnover_Rate_times
8	_Current_Ratio
12	_Total_Asset_Turnover
9	_Quick_Ratio
30	_Cash_Flow_to_Total_Assets
7	_Cash_Reinvestment_perc
32	_Cash_Flow_to_Equity
33	_Current_Liability_to_Current_Assets
4	_Cash_Flow_Per_Share
37	_Equity_to_Liability
19	_Quick_Assets_to_Total_Assets
10	_Total_debt_to_Total_net_worth
22	_Cash_to_Current_Liability
20	_Cash_to_Total_Assets
24	_Inventory_to_Current_Liability
29	_Fixed_Assets_to_Assets
18	_Allocation_rate_per_person

variables	VIF
17	_Operating_profit_per_person
26	_Total_expense_to_Assets
15	_Fixed_Assets_Turnover_Frequency
13	_Average_Collection_Days
35	_Total_assets_to_GNP_price
25	_Long_term_Liability_to_Current_Assets
27	_Quick_Asset_Turnover_Rate
3	_Tax_rate_A
0	_Operating_Expense_Rate
1	_Research_and_development_expense_rate
14	_Inventory_Turnover_Rate_times
6	_Total_Asset_Growth_Rate
28	_Cash_Turnover_Rate
34	_Liability_Assets_Flag
36	_Net_Income_Flag

- many columns has VIF value of greater than 5%, we have keep dropping the columns with VIF greater than 5% until all the columns has vif less than 5%

variables	VIF
14	_Quick_Assets_to_Total_Assets
26	_Equity_to_Liability
24	_Current_Liability_to_Current_Assets
16	_Cash_to_Current_Liability
15	_Cash_to_Total_Assets
8	_Total_Asset_Turnover
7	_Total_debt_to_Total_net_worth
2	_Cash_flow_rate
5	*Per_Share_Net_profit_before_tax_Yuan*
22	_Fixed_Assets_to_Assets
13	_Allocation_rate_per_person
4	_Cash_Flow_Per_Share
12	_Operating_profit_per_person
17	_Inventory_to_Current_Liability
19	_Total_expense_to_Assets
11	_Fixed_Assets_Turnover_Frequency
9	_Average_Collection_Days

variables	VIF
25	_Total_assets_to_GNP_price
18	_Long_term_Liability_to_Current_Assets
23	_Cash_Flow_to_Equity
20	_Quick_Asset_Turnover_Rate
3	_Tax_rate_A
0	_Operating_Expense_Rate
1	_Research_and_development_expense_rate
10	_Inventory_Turnover_Rate_times
6	_Total_Asset_Growth_Rate

- now we have all the columns with less then 5% VIF.

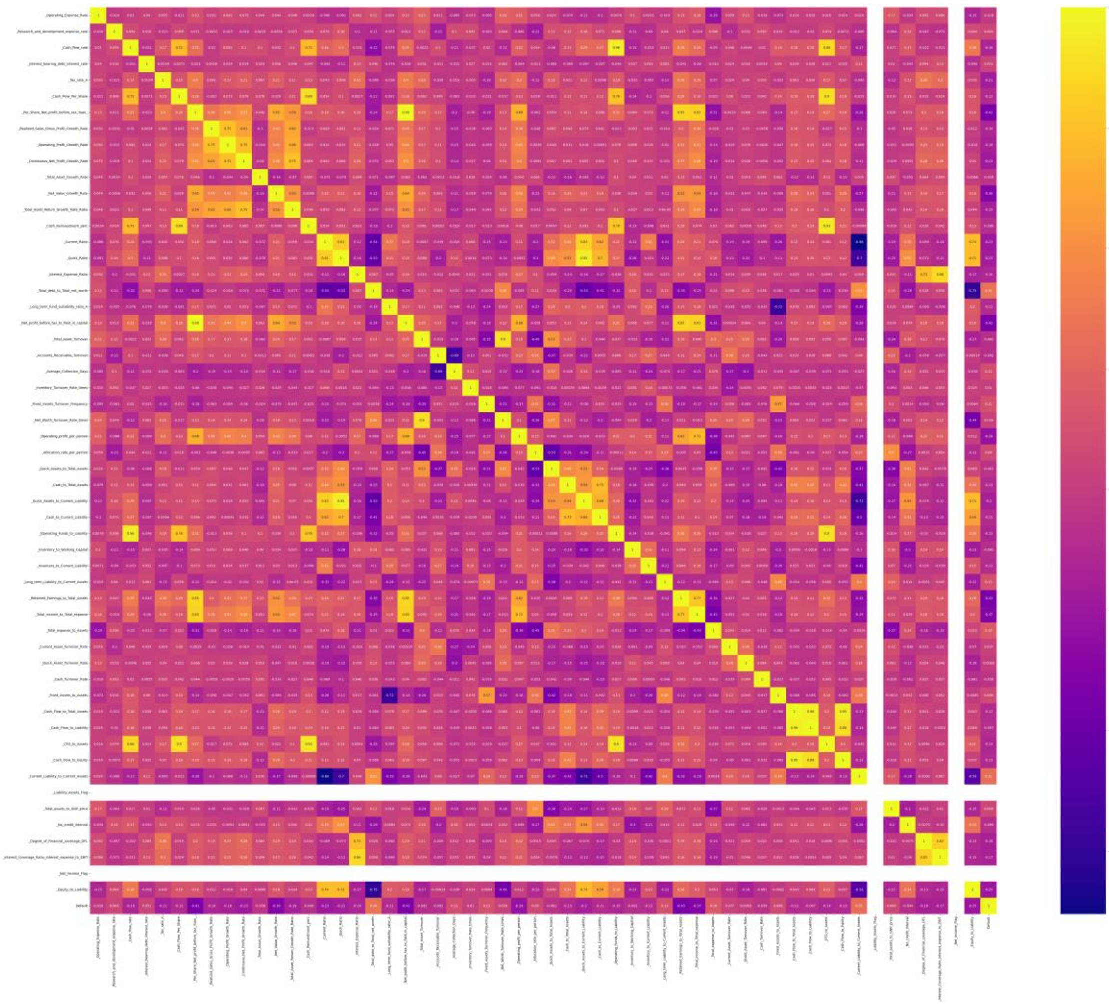
PART A: Train Test Split

- The treain and test split is doner with the ratio 67/33 and the random state is set to be 42,

```
print("Train dataset shape:", X_train.shape, y_train.shape)
print("Test dataset shape:", X_test.shape, y_test.shape)
```

the above shows the shape of the data after train/test split is done

the below is the correltion plot



Positive correlation: If there's a positive correlation, points will generally trend upwards from left to right. This would suggest that places with higher average precipitation tend to have higher average temperatures as well.

Negative correlation: If there's a negative correlation, points will generally trend downwards from left to right. This would suggest that places with higher average precipitation tend to have lower average temperatures.

No correlation: If there's no clear trend, the data points will be scattered randomly across the plot. This would suggest no relationship between average temperature and precipitation

PART A: Build Logistic Regression Model (using statsmodels library) on most important variables on train dataset and choose the optimum cut-off. Also showcase your model building approach

Model Building using Logistic Regression for 'Probability at default'
The equation of the Logistic Regression by which we predict the corresponding probabilities and then go on predict a discrete target variable is
y = 11+�−�

Note: z = ⍰0 +∑⍰⍰=1(⍰⍰⍰1)
Creating logistic regression equation & storing it in f_1
model = SM.logit(formula='Dependent Variable ~ Σ⍰⍰⍰⍰⍰⍰⍰⍰⍰⍰⍰ ⍰⍰⍰⍰⍰⍰⍰⍰⍰ (⍰)'

we fit a logistic regression model (`model_1`) to the training data (`Default_train`) based on the specified formula (`m1`). After fitting the model, you can use `model_1` to make predictions on new data or to analyze the relationship between the dependent and independent variables in the model.

Summary of model 1

Dep. Variable:
Model:
Method:
Date:
Time:
converged:
Covariance Type:

Intercept
_Operating_Expense_Rate
_Research_and_development_expense_rate
_Cash_flow_rate
_Tax_rate_A
_Cash_Flow_Per_Share
Per_Share_Net_profit_before_tax_Yuan
_Total_Asset_Growth_Rate
_Total_debt_to_Total_net_worth
_Total_Asset_Turnover
_Average_Collection_Days
_Inventory_Turnover_Rate_times
_Fixed_Assets_Turnover_Frequency
_Operating_profit_per_person
_Allocation_rate_per_person
_Quick_Assets_to_Total_Assets
_Cash_to_Total_Assets
_Cash_to_Current_Liability
_Inventory_to_Current_Liability

_Long_term_Liability_to_Current_Assets
_Total_expense_to_Assets
_Quick_Asset_Turnover_Rate
_Cash_Turnover_Rate
_Fixed_Assets_to_Assets
_Cash_Flow_to_Equity
_Current_Liability_to_Current_Assets
_Total_assets_to_GNP_price
_Equity_to_Liability

- If the coef is positive and if value increases then will the prop of default increases prob of default, if coef is -ve and value increases then it will decrease the probability of default
  -1378 - The number of data points used to fit the model.
  -R-squared: 0.4163 - This is a measure of the goodness of fit of the model, indicating the
  proportion of variance explained by the model relative to a null model.

- 273.06 - The log-likelihood value represents the goodness of fit of the model, with higher values indicating better fit.

- Variables with p-values less than the chosen significance level (typically 0.05) are considered statistically significant predictors of the outcome.

- For instance, the variable '*Operating_Expense_Rate' has a coefficient of 0.1583 with a p-value of 0.230, suggesting that it is not statistically significant at the conventional significance level of 0.05. Therefore, it may not have a significant impact on predicting the likelihood of 'Default'.*

- Conversely, the variable '_Research_and_development_expense_rate' has a coefficient of 0.4418 with a p-value less than 0.001, indicating statistical significance. Thus, a one-unit increase in this variable is associated with an increase in the log odds of 'Default' by 0.4418 units, all else being equal.

After we dropped few columns and performed **23 model**, this is the final result where The pvalues are all 0's

```
Optimization terminated successfully.
         Current function value: 0.209689
         Iterations 8
```

Dep. Variable:

Dep. Variable:

Model:

Method:

Date:

Time:

converged:

Covariance Type:

Intercept

_Research_and_development_expense_rate

Per_Share_Net_profit_before_tax_Yuan

_Total_debt_to_Total_net_worth

_Average_Collection_Days

_Quick_Assets_to_Total_Assets

- The optimization terminated successfully after 8 iterations.
- The current function value, representing the negative log-likelihood of the model, is approximately 0.2097.
- The model was estimated using a maximum likelihood estimation (MLE) method.

1. **Model Information**:
   - Dependent Variable: `Default`
   - Number of Observations: 1378
   - Model: Logit (Logistic Regression)
   - Degrees of Freedom (Residuals): 1372
   - Degrees of Freedom (Model): 5
   - Pseudo R-squared: 0.3824
   - Log-Likelihood: -288.95
   - LL-Null: -467.84 (Log-Likelihood of the null model)
   - Convergence: The model converged successfully.
   - Covariance Type: Non-robust (Assuming homoscedasticity)

_Research_and_development_expense_rate: For a one-unit increase in the research and development expense rate, the log-odds of the Default variable increase by 0.3338.

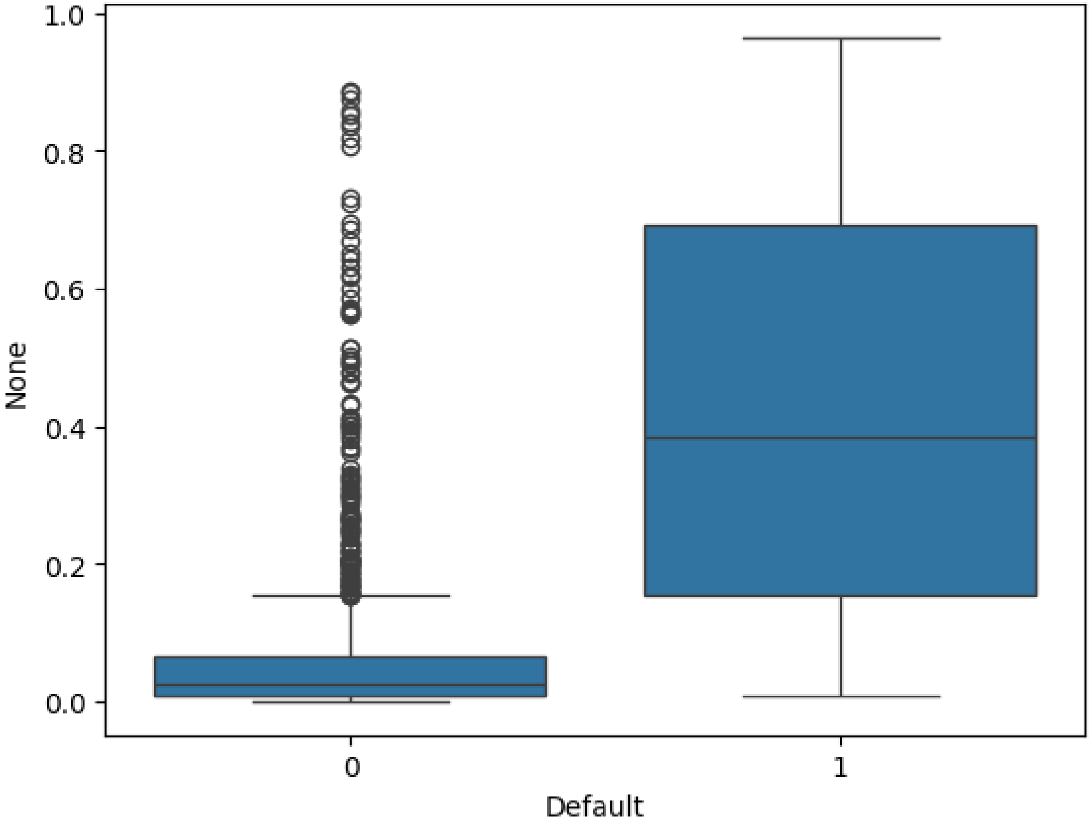
Per_Share_Net_profit_before_tax_Yuan: For a one-unit increase in the per-share net profit before tax (in Yuan), the log-odds of the Default variable decrease by 1.4702.

_Total_debt_to_Total_net_worth: For a one-unit increase in the total debt to total net worth ratio, the log-odds of the Default variable increase by 0.7801.

_Average_Collection_Days: For a one-unit increase in the average collection days, the log-odds of the Default variable increase by 0.4710.

_Quick_Assets_to_Total_Assets: For a one-unit increase in the quick assets to total assets ratio, the log-odds of the Default variable decrease by 0.6475.

PART A: Validate the Model on Test Dataset and state the performance metrics. Also state interpretation from the model

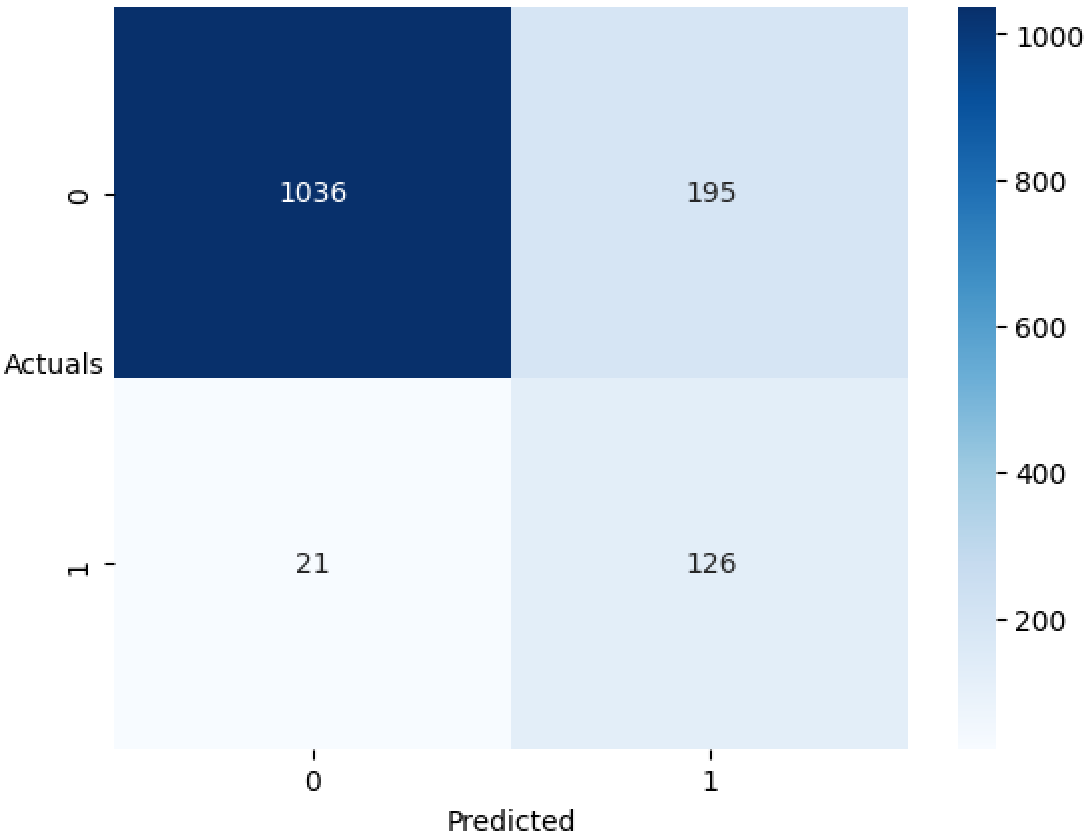


```
2011    0.110
697     0.009
160     0.064
1273    0.040
541     0.104
        ...
1386    0.015
```

```
1127    0.006
950     0.020
1058    0.025
562     0.249
Length: 1378, dtype: float64
```

```
the optimal threshold = 0.1072017973601675
```

logistic regression model for train data

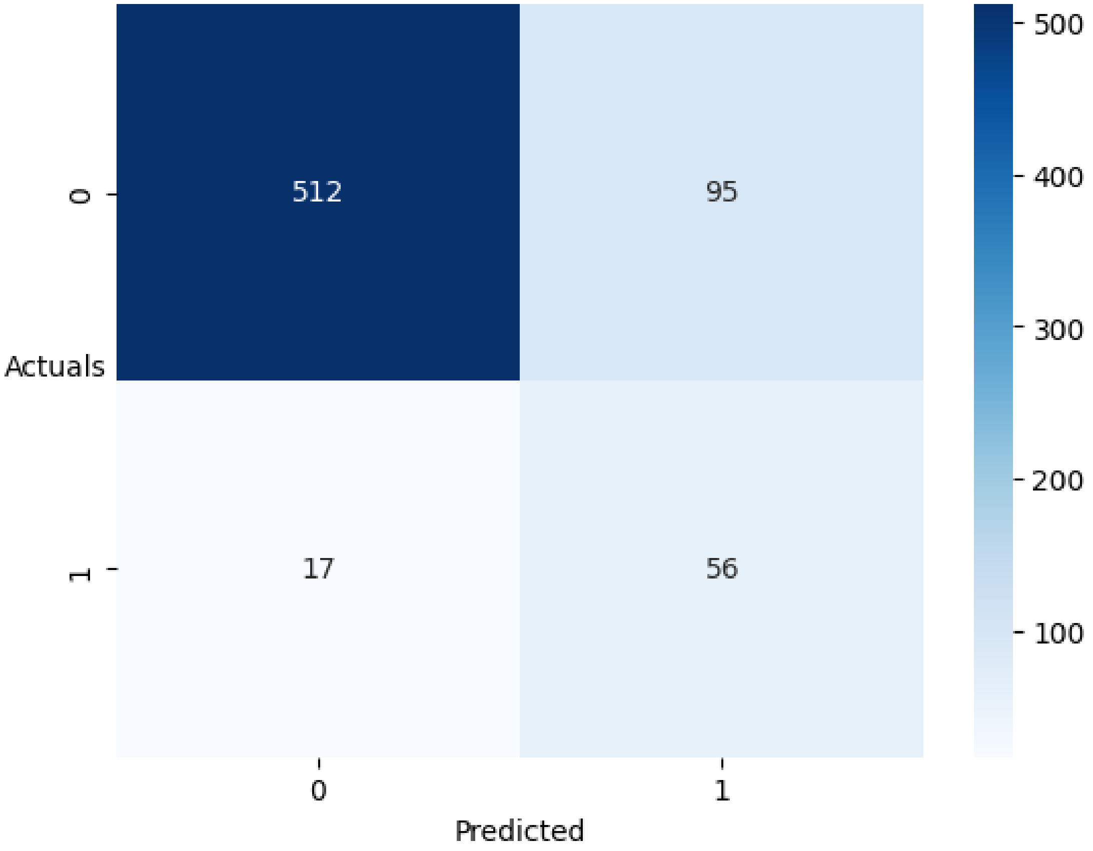


- High Recall (85%):
- The model has a high recall, indicating that it effectively identifies most of the instances of 'Default' in the dataset. This suggests that the model is good at capturing instances of 'Default', minimizing false negatives. In other words, it correctly identifies a large portion of companies that are likely to default on their loans.
- Low Precision (39%):
- Despite the high recall, the model has a relatively low precision. This means that among the instances predicted as 'Default' by the model, only 39% of them are true

positives. In other words, there is a significant number of false positives among the instances predicted as 'Default' by the model.

- The high recall indicates that the model is sensitive to identifying instances of 'Default' and is effective in capturing most of them.
- However, the low precision suggests that the model may be overly aggressive in predicting 'Default', leading to a considerable number of false positives.
- Therefore, while the model is good at identifying companies at risk of defaulting, it also misclassifies a substantial number of non-default cases as default, which could lead to unnecessary actions or interventions.

Logistic regresssion model for test data



recall is = 0.7671232876712328

precision is = 0.3708609271523179

- Similar to the training data, the model in the test data demonstrates high recall, indicating good performance in capturing actual positive cases.

- However, the precision remains low, implying that the model's positive predictions have a significant proportion of false positives.

PART A: Build a Random Forest Model on Train Dataset. Also showcase your model building approach

The parameters are

```
{'max_depth': 7,
 'min_samples_leaf': 10,
 'min_samples_split': 15,
 'n_estimators': 25}
```

We utilized Grid Search Cross-Validation to determine the best hyperparameters for the Random Forest Classifier. This involved exploring different combinations of hyperparameters such as 'max_depth', 'min_samples_leaf', 'min_samples_split', and 'n_estimators'. After thorough exploration, the optimal parameter values were identified as follows: 'max_depth' set to 7, 'min_samples_leaf' set to 10, 'min_samples_split' set to 30, and 'n_estimators' set to 50. These selected parameter values will be employed to construct the Random Forest Classifier model, ensuring the highest performance and accuracy in predicting the outcome.

- Classification report of train data

```
              precision    recall  f1-score   support

           0       0.94      1.00      0.97      1231
           1       0.95      0.49      0.65       147

    accuracy                           0.94      1378
   macro avg       0.94      0.74      0.81      1378
weighted avg       0.94      0.94      0.93      1378
```

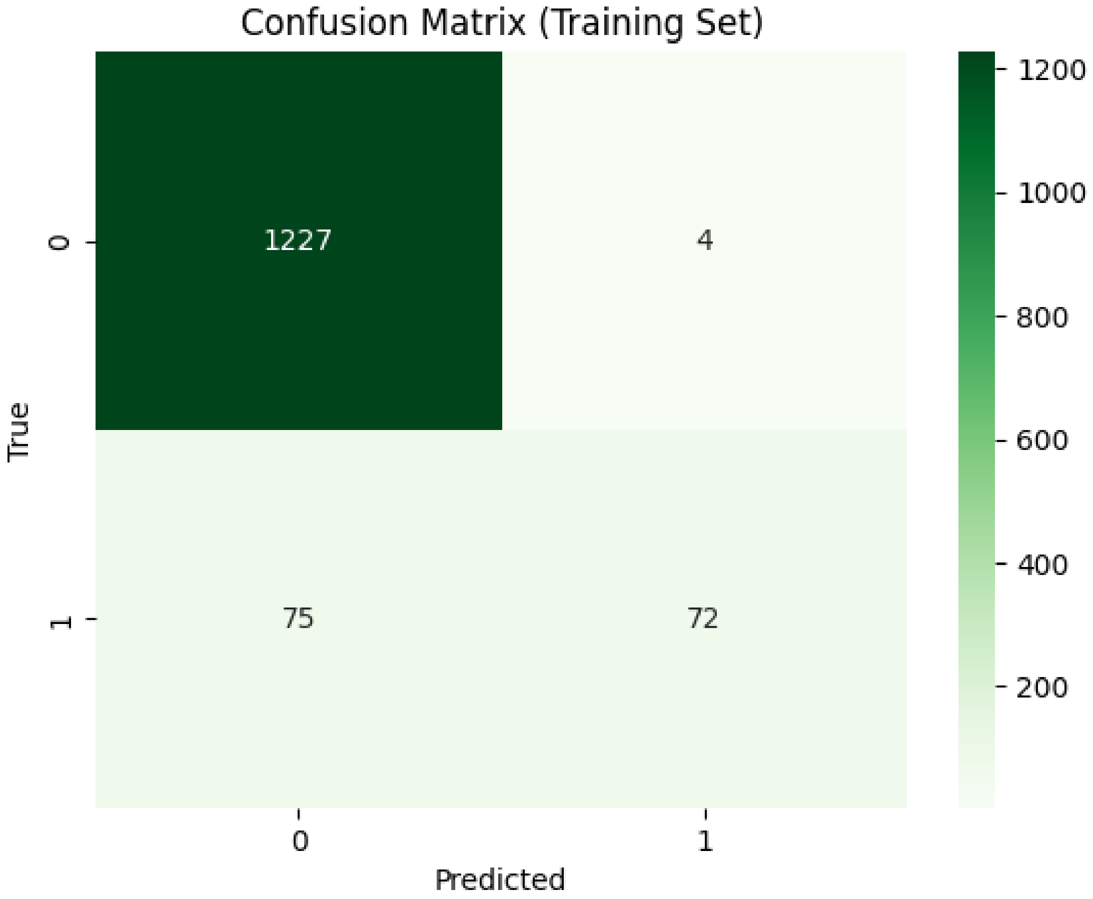
The model shows strong performance in correctly identifying negative cases (class 0), with 94% precision and 100% recall, indicating it rarely misclassifies negatives. However, it struggles to identify positive cases (class 1), with only 49% recall, meaning it misses nearly half of the actual positives. Consequently, the F1-score for class 1 is lower at 65%, highlighting the trade-off between precision and recall. The overall accuracy is high at 94%, but it's crucial to consider the class imbalance. The macro average precision, recall, and F1-score are 94%, 74%, and 81%, respectively, with equal weight to all classes. Weighted average precision, recall, and F1-score are 94%, 94%, and 93%, respectively, reflecting the influence of class distribution. Despite strong performance in predicting negatives, the model's effectiveness in capturing positives requires improvement, especially considering the application's specific needs and class distribution.

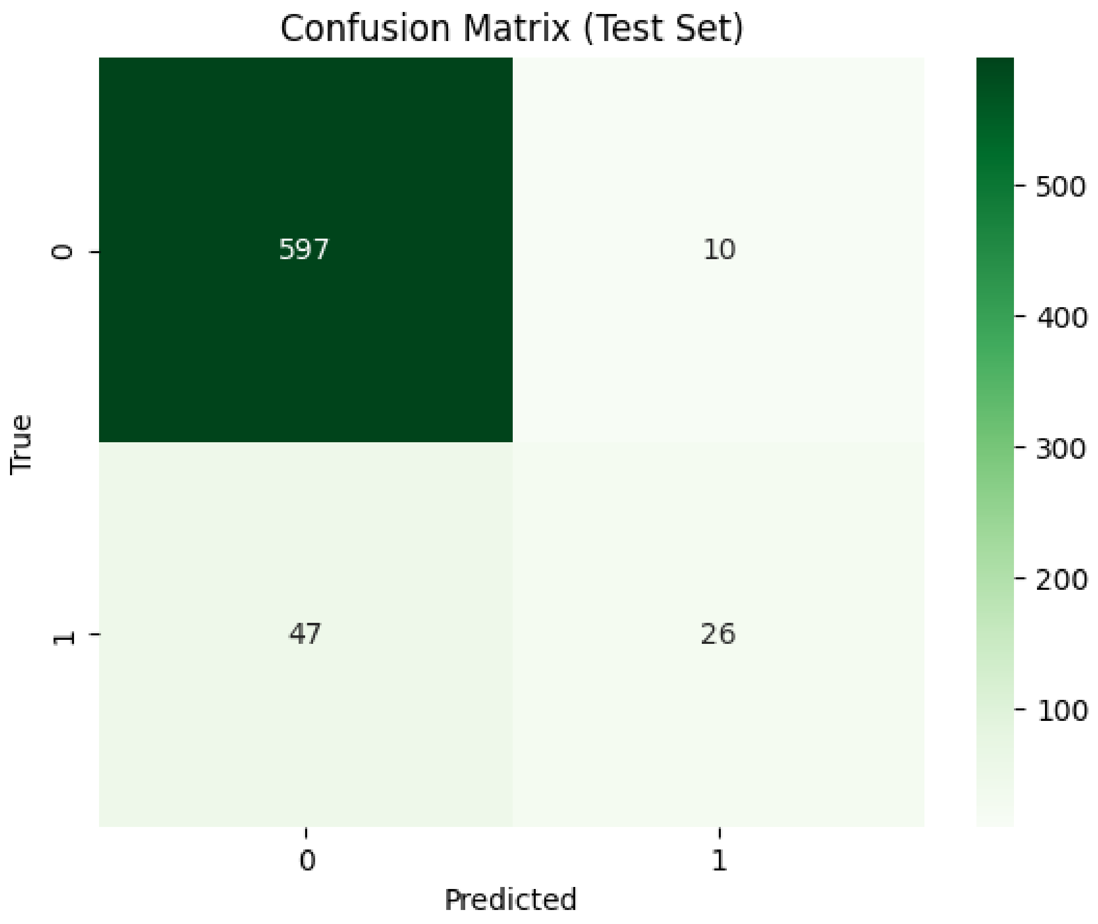
- Classification report of test data

```
          precision    recall  f1-score   support

         0       0.93      0.98      0.95       607
         1       0.72      0.36      0.48        73

  accuracy                           0.92       680
```

macro avg 0.82 0.67 0.72 680
weighted avg 0.91 0.92 0.90 680

The classification report for the Random Forest model on the test data reveals that it performs reasonably well, with 93% precision for class 0 (negative) and 72% precision for class 1 (positive). However, it demonstrates lower recall for class 1 at 36%, indicating it misses a significant portion of actual positives. The F1-score for class 1 is 48%, reflecting the balance between precision and recall. The overall accuracy is 92%, showcasing the model's ability to correctly classify instances. The macro average precision, recall, and F1-score are 82%, 67%, and 72%, respectively, suggesting some imbalance in class performance. When comparing with the training data, the model shows similar trends of high precision for negatives but struggles to capture positives effectively, indicating room for improvement in identifying positive cases.



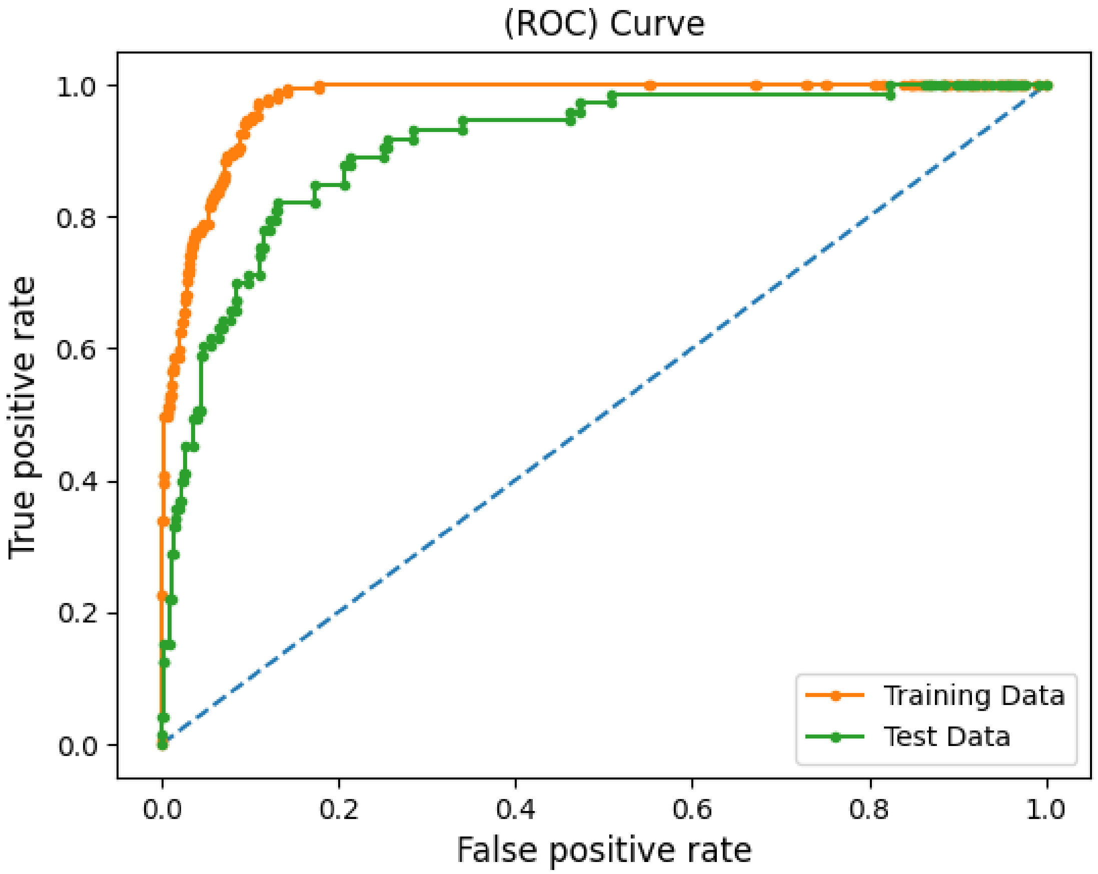


Here after performing logistic regression, the model which we are working on turned out to be overfitting as the recall in training showed 54% recall where as the recall in the test has showed only 31% recall

- This also says that the model not accurate and can't be hightly relied on

- The precesion looks pretty good in both the train and test data

- among the 2 models we have performed above Logistic regression seems to have a bette performance.



ROC curve (Receiver Operating Characteristic Curve). It is a visual representation of the performance of a classification model at various classification thresholds. The x-axis represents the false positive rate (FPR), and the y-axis represents the true positive rate (TPR).

- The diagonal line represents where the TPR (true positive rate) is equal to the FPR (false positive rate). The further the curve is from this diagonal line, the better the performance of the model at classifying the data.
- The area under the ROC curve (AUC) is a numerical measure of the performance of a classifier. A larger AUC indicates better performance.
- In the specific ROC curve you sent, the AUC appears to be high, which suggests that the model is performing well at classifying between positive and negative instances.
- The curve starts at a point slightly above (0,0) and ends at a point slightly below (1,1), which suggests that the model is returning some true positives and some false positives.
- It is difficult to say definitively from this image how well-calibrated the model is, but a well-calibrated model would ideally produce a smooth curve.

- The ROC curve can also be used to compare the performance of two or more models on the same classification task.
- By plotting the ROC curves of two models on the same graph, you can visually compare their performance and see which model performs better at classifying the data.
- ROC curves are typically used in machine learning tasks where the goal is to binary classification.
- In conclusion, the ROC curve you sent is a helpful visualization of the performance of a classification model. The high AUC suggests that the model is performing well.

PART A: Build a LDA Model on Train Dataset. Also showcase your model building approach

LDA, or Linear Discriminant Analysis, is a supervised learning technique used for dimensionality reduction and classification. It aims to find a linear combination of features that maximizes class separability. By projecting data onto a lower-dimensional space, LDA preserves class discriminatory information. Unlike PCA, LDA focuses on maximizing class separability rather than variance. Assumptions include normally distributed features and identical covariance matrices across classes. LDA finds applications in face recognition, bioinformatics, and text classification, among others, for tasks like pattern recognition and feature extraction.

- classification on train data

```
        precision    recall  f1-score   support

      0       0.94      0.97      0.95      1231
      1       0.64      0.50      0.56       147

accuracy                          0.92      1378
```

macro avg 0.79 0.73 0.76 1378
weighted avg 0.91 0.92 0.91 1378

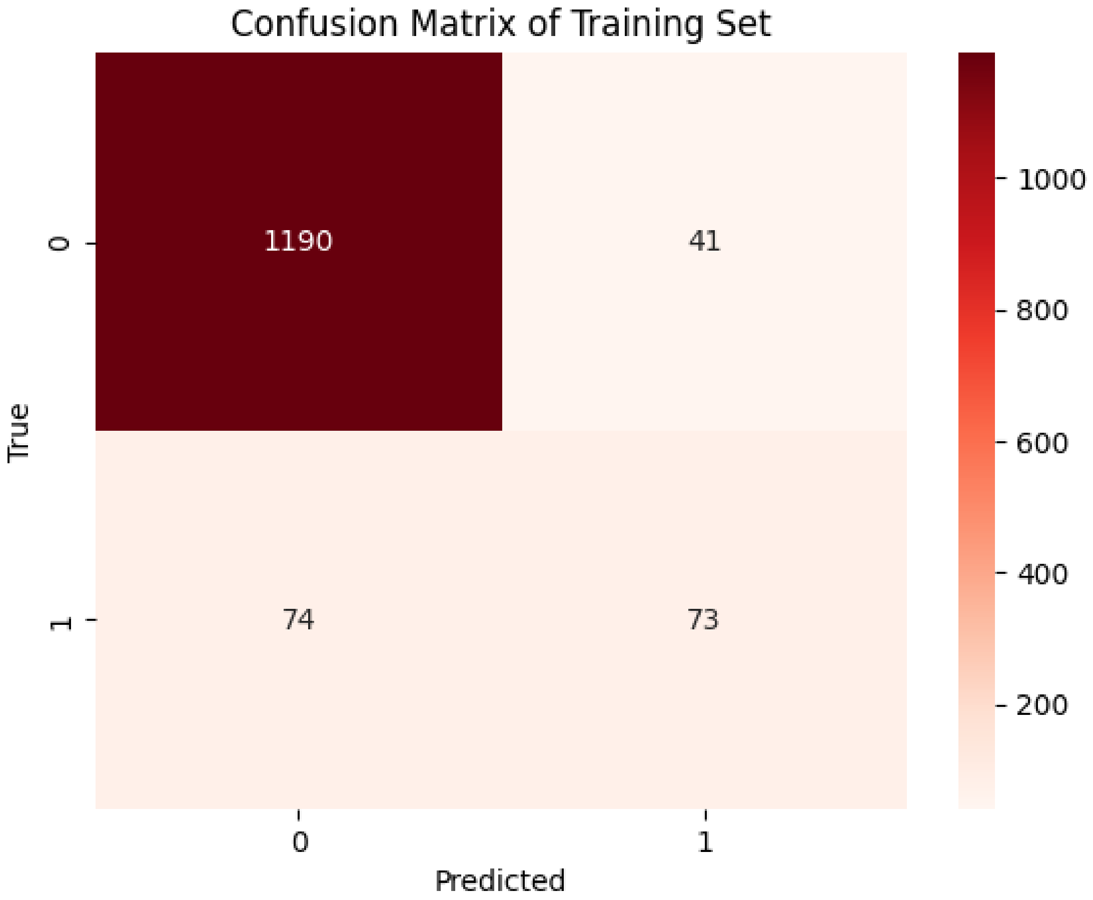
The classification report for the LDA model on the training data suggests a strong performance overall. It achieves high precision of 94% for class 0 (negative) and moderate precision of 64% for class 1 (positive). However, the recall for class 1 is relatively lower at 50%, indicating that the model misses some actual positive cases. The F1-score for class 1 is 56%, reflecting the balance between precision and recall. The overall accuracy is 92%, demonstrating the model's ability to correctly classify instances. The macro average precision, recall, and F1-score are 79%, 73%, and 76%, respectively, indicating a reasonably balanced performance across classes. This suggests that while the model performs well overall, there may be a need for further optimization to improve the identification of positive cases.

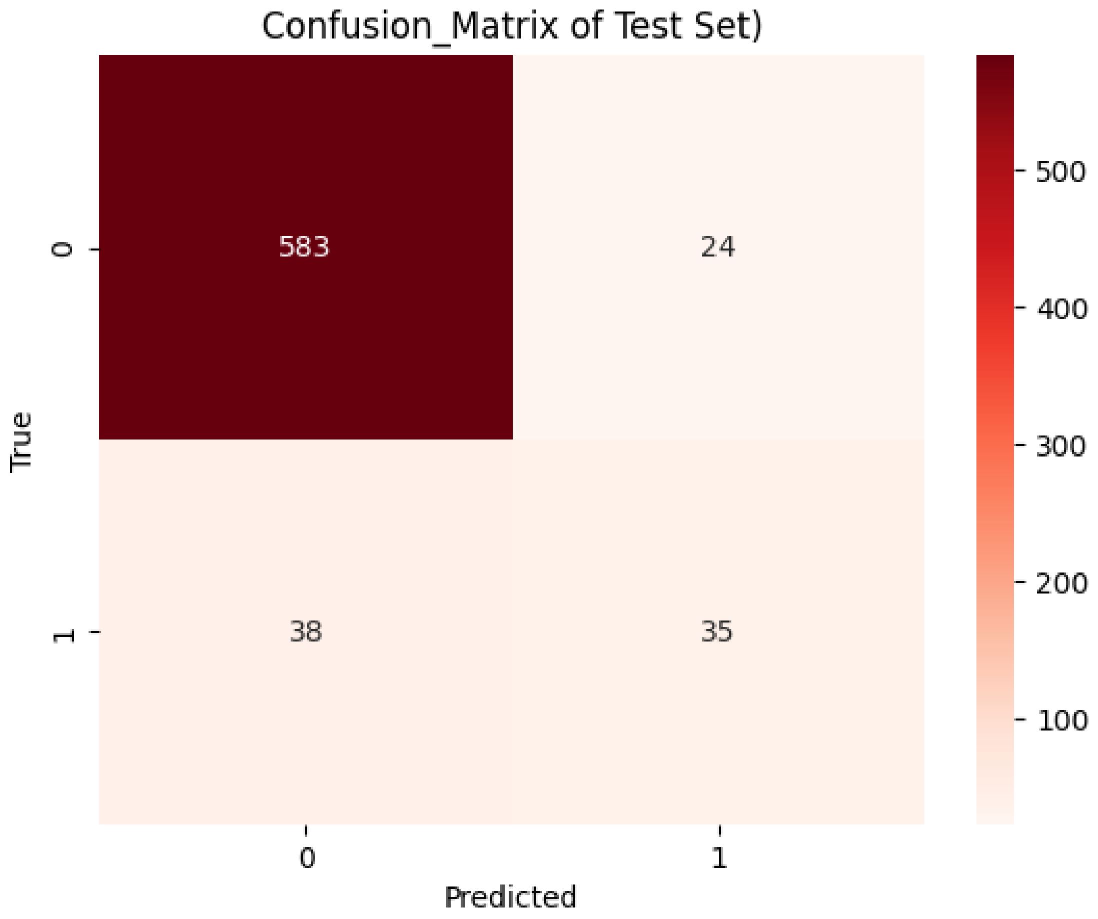
- classification on test data

```
          precision    recall  f1-score   support

       0       0.94      0.96      0.95       607
       1       0.59      0.48      0.53        73

accuracy                           0.91       680
```

macro avg 0.77 0.72 0.74 680
weighted avg 0.90 0.91 0.90 680

The classification report for the LDA model on the test data indicates that it performs relatively well. It achieves high precision of 94% for class 0 (negative) and moderate precision of 59% for class 1 (positive). However, the recall for class 1 is lower at 48%, suggesting that the model misses a significant portion of actual positives. The F1-score for class 1 is 53%, reflecting the balance between precision and recall. The overall accuracy is 91%, demonstrating the model's ability to correctly classify instances. The macro average precision, recall, and F1-score are 77%, 72%, and 74%, respectively, indicating a reasonably balanced performance across classes. This suggests that while the model performs well overall, there is room for improvement, particularly in correctly identifying positive cases.

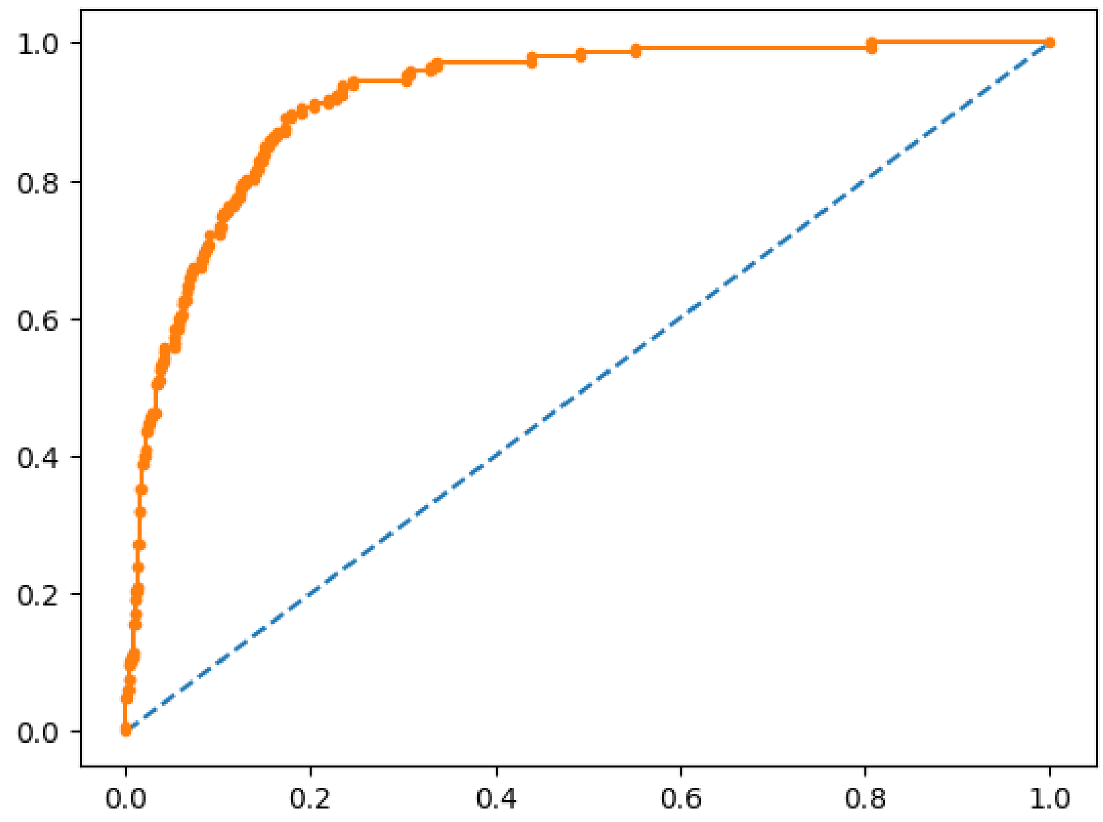

Confusion Matrix of Training Set
True
0
1
1190
41
74
73
Predicted
0
1
1000
800
600
400
200



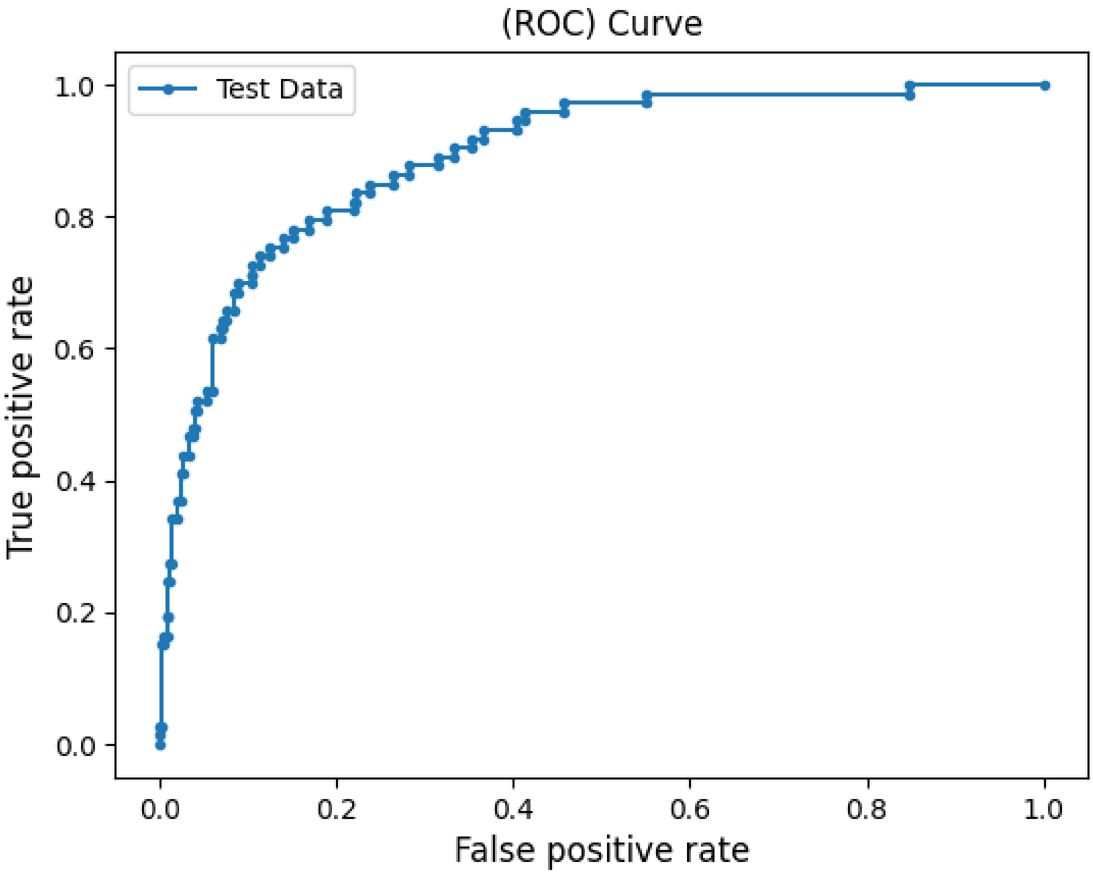
- The recall on the train set is approximately 49.66%, indicating that the model correctly identifies about 49.66% of the actual positive cases in the train set.
- The precision on the train set is around 64.04%, suggesting that among the instances predicted as positive by the model, approximately 64.04% are indeed true positives.
- On the test set, the recall is approximately 47.95%, showing that the model identifies about 47.95% of the actual positive cases in the test set.
- The precision on the test set is about 59.32%, indicating that among the instances predicted as positive by the model on the test set, approximately 59.32% are true positives.

Overall, while the model demonstrates some capability in correctly identifying positive cases, there's room for improvement, particularly in recall, which suggests the model could benefit from better identifying actual positive cases.

Auc curve of train data



curve of test data



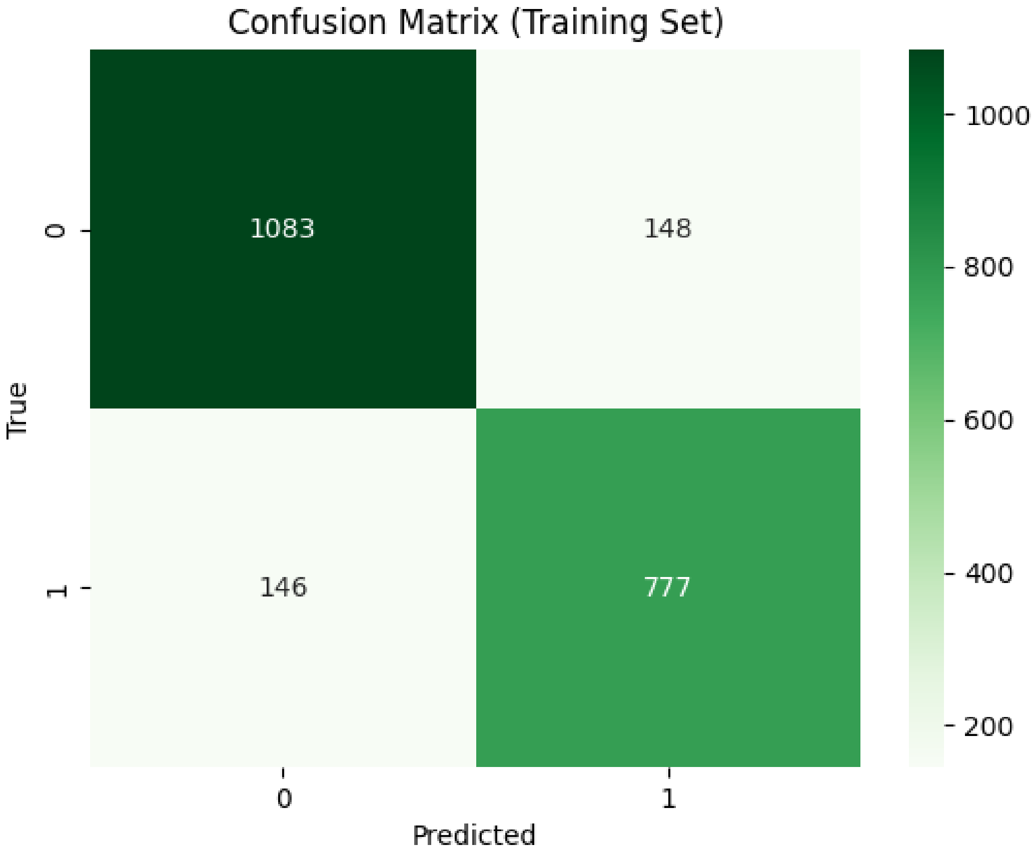
- **0.921 for training data:** This is a very good AUC score, indicating that the model is performing well on the data it was trained on. It can correctly classify positive and negative instances most of the time on the training data.
- **0.893 for test data:** This is still a good AUC score, but it is lower than the training data. This means that the model is not generalizing as well as it could. It is possible that the model is overfitting the training data and has not learned the underlying patterns that generalize well to unseen data.

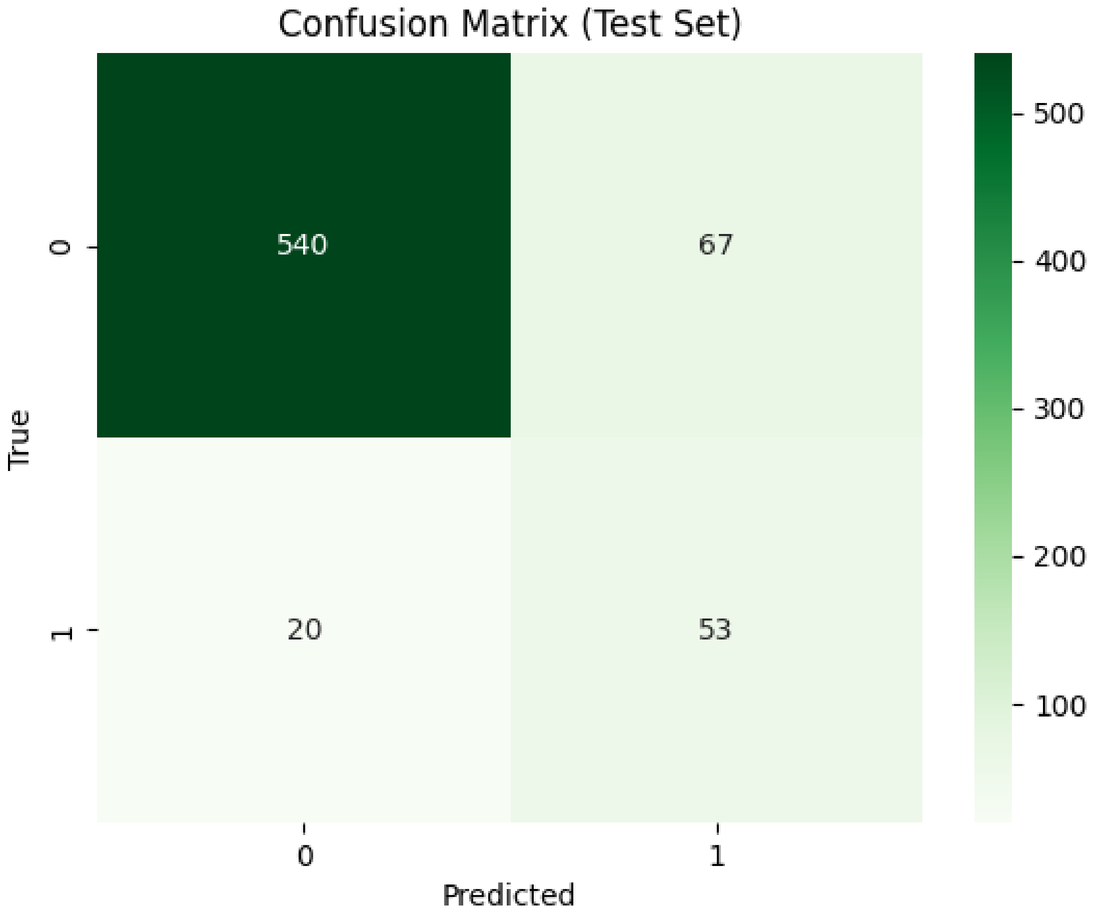
Here are some additional things to keep in mind:

- The difference between the training AUC and the test AUC is relatively small (0.028). This suggests that the model may not be severely overfitting.
- It is important to use a validation set to monitor the performance of a model during training to avoid overfitting.
- Other factors, such as the size and quality of the data, can also affect the performance of a model.

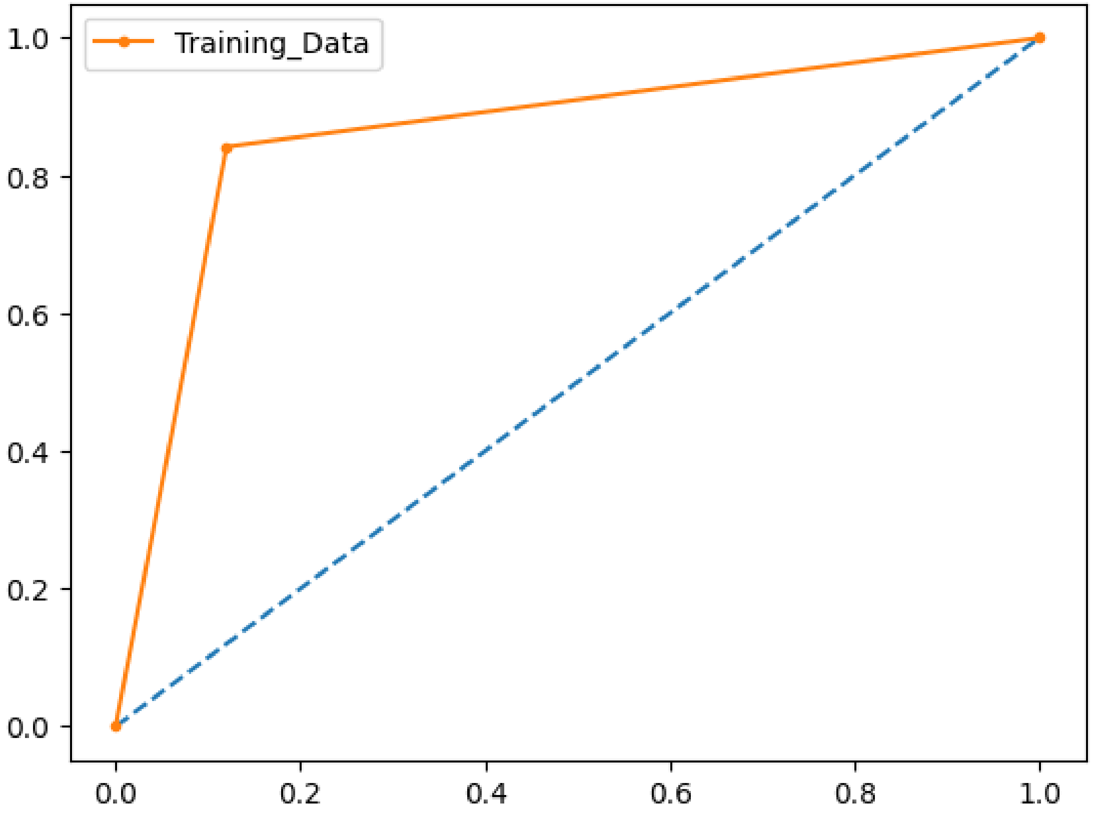
Overall, the AUC scores suggest that the model is performing well, but there is some room for improvement. By addressing potential overfitting issues, the performance of the model on unseen data could be improved.

PART A: Validate the LDA Model on test Dataset and state the performance metrics. Also state interpretation from the model



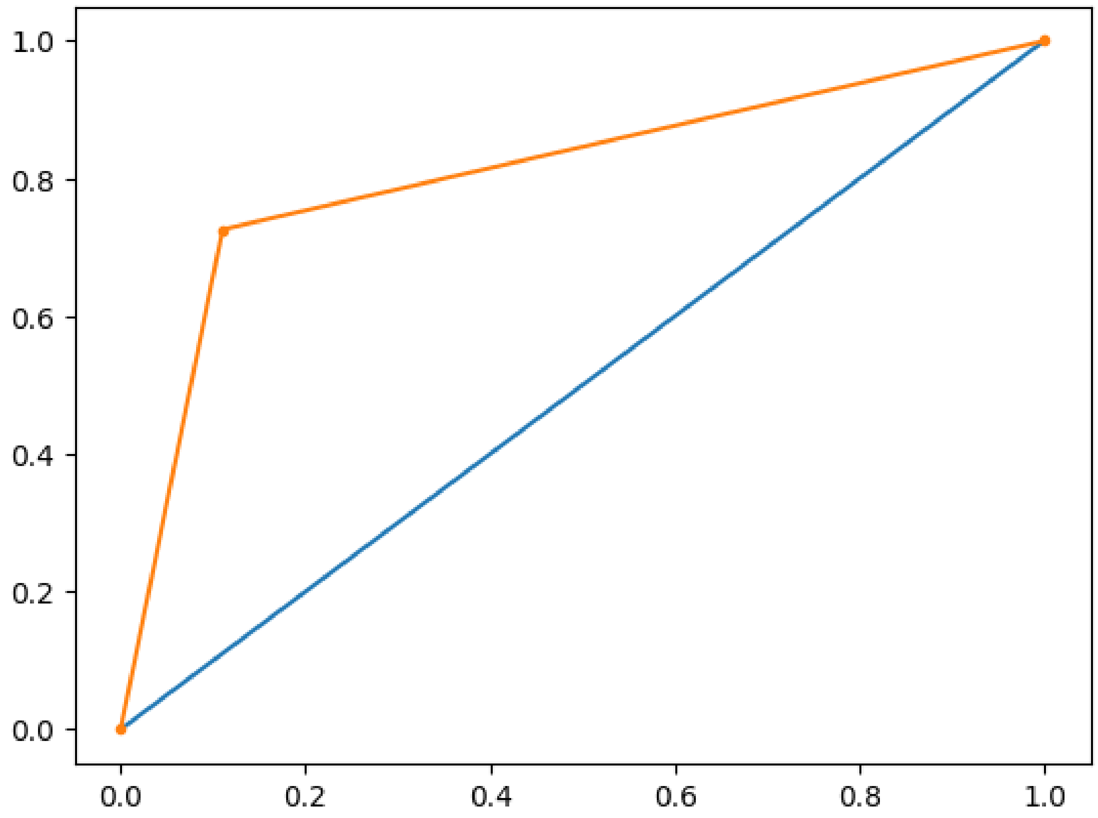


Confusion Matrix (Test Set)



The curve starts at the top left corner and goes down towards the right, which is a typical characteristic of a precision-recall curve.

As the curve progresses to the right, the precision generally decreases while the recall increases. This suggests a trade-off between these two metrics. The model might be good at filtering out negative instances (resulting in high precision at the beginning of the curve), but it may miss some positive cases (resulting in lower recall) at this precision level. As the classification threshold is adjusted to capture more positive cases (increasing recall on the right side of the curve), the precision suffers (meaning more false positives are included).



PART A: Compare the performances of Logistic Regression, Random Forest, and LDA models (include ROC curve)

Logistic Regression:-This model showed good performance in identifying positive cases, with an 85% recall rate on the training data and 76% on the test data. However, its precision, which measures its ability to avoid false positives, was moderate at around 40% on the training data and 37% on the test data. The model's AUC values were decent, indicating its ability to distinguish between classes.

Random Forest:-This model demonstrated excellent precision, especially on the training data, with a rate of 96%. However, its recall, which measures its ability to capture all positive cases, was comparatively lower at 54% on the training data and 31% on the test data. Nonetheless, its AUC values were impressive, indicating strong discriminatory power and generalization.

LDA:-The LDA model's performance fell between Logistic Regression and Random Forest. It showed moderate recall and precision rates, with slightly lower recall than Logistic Regression and slightly higher precision than Random Forest. Its AUC values were good, indicating effective discrimination between classes.

Overall, Random Forest emerged as the top performer, excelling in terms of AUC, recall, and precision. Logistic Regression and LDA showed comparable performance, with Logistic Regression having slightly better recall and LDA slightly better precision. The choice of model depends on specific needs; for instance, Logistic Regression might be preferred for high recall, while Random Forest could be chosen for high precision.

PART A: Conclusions and Recommendations

Conclusions:

- The Logistic Regression model, with a threshold of 0.1076, shows consistent performance. It predicts defaults with an accuracy of 84% on the training data and 83.5% on the testing data, indicating stable predictions across different datasets.
- Both precision and recall values for non-default and default cases remain steady, indicating the model's reliability in identifying instances from both classes.
- The AUC scores confirm the model's ability to distinguish between default and non-default cases consistently, highlighting its stable discriminative power.

Recommendations:

- Increase Data Collection: Gathering more data, particularly on default cases, can improve the model's exposure to such instances and enhance its predictive performance.
- Explore Feature Engineering: Further exploration of the model through feature engineering techniques may enhance its ability to identify default cases more accurately. This involves modifying or creating new features from existing data.
- Consider Ensemble Techniques: Incorporating ensemble techniques like boosting or bagging could improve the model's performance in identifying default cases. These methods involve combining multiple models to enhance predictive accuracy and robustness.
- Overall, while the Logistic Regression model demonstrates stable and reliable performance in predicting defaults, there's room for improvement through data collection, feature engineering, and the adoption of ensemble techniques. These steps can enhance the model's accuracy in identifying default cases effectively.

Problem Statement-B:

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 using Python.

The data is properluy read and below is the first 5 rows of the data set

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
1	07-04-2014	257	68	458	276	70	5728	610	279	84
2	14-04-2014	254	68	454	270	68	5649	607	279	83
3	21-04-2014	253	68	488	283	68	5692	604	274	83
4	28-04-2014	256	65	482	282	63	5582	611	238	79

- (314, 11) is the shape of the dataset, that is it has 314 rows and 11 columns.

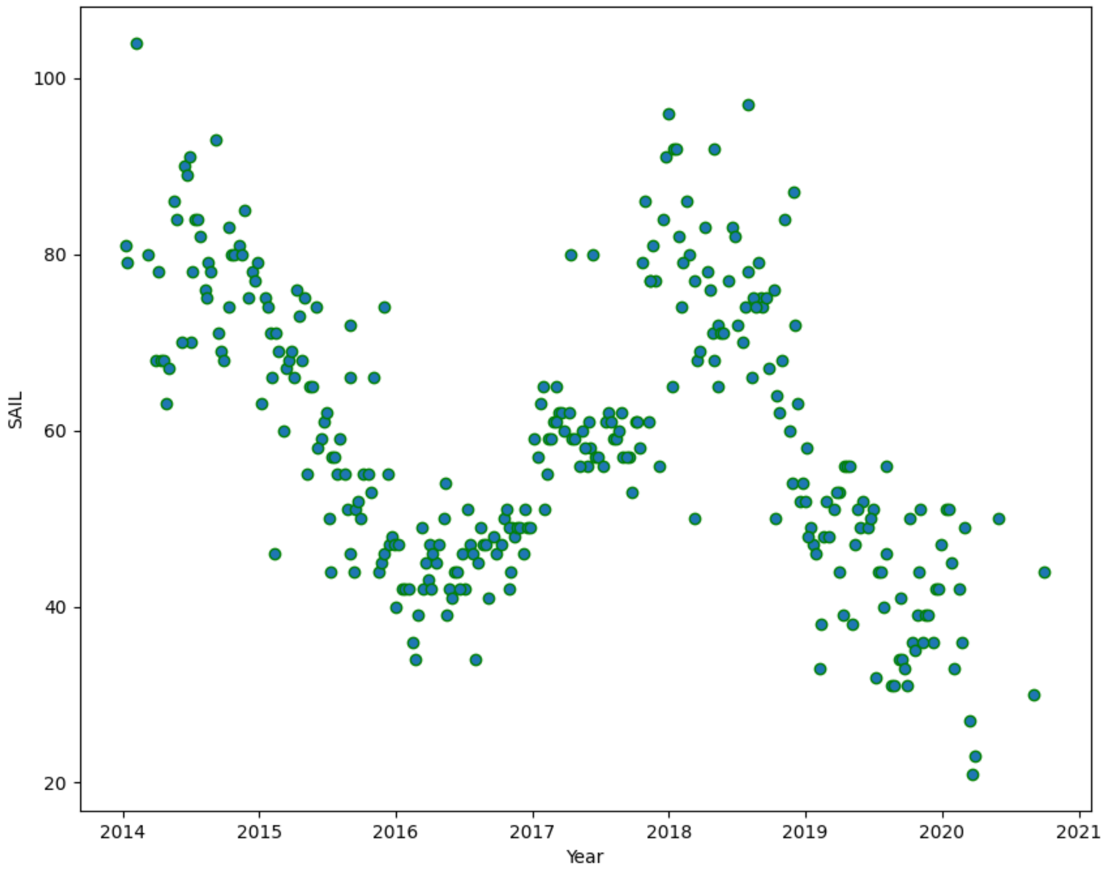
- there are no null are missing values in the dataset

```
#   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)
memory usage: 27.1+ KB
```

- The data has 10 int and 1 object datatype.

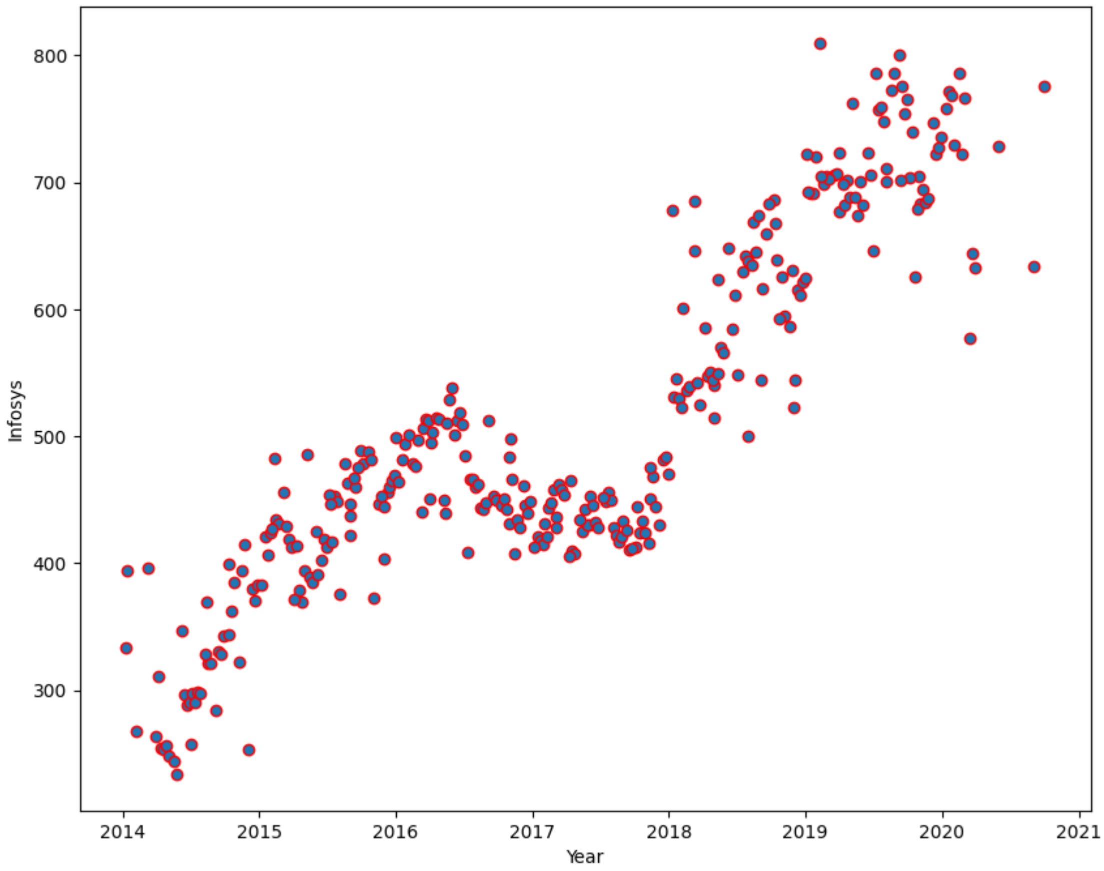
Infosys	Indian Hotel	Mahindra & Mahindra	Axis Bank	SAIL	Shree Cement	Sun Pharma	Jindal Steel	Idea Vodafone	Jet Airways
count	314.000000	314.000000	314.000000	314.000000	314.000000	314.000000	314.000000	314.000000	314.000000
mean	511.340764	114.560510	636.678344	540.742038	59.095541	14806.410828	633.468153	147.627389	53.713376
std	135.952051	22.509732	102.879975	115.835569	15.810493	4288.275085	171.855893	65.879195	31.248985
min	234.000000	64.000000	284.000000	263.000000	21.000000	5543.000000	338.000000	53.000000	3.000000
25%	424.000000	96.000000	572.000000	470.500000	47.000000	10952.250000	478.500000	88.250000	25.250000
50%	466.500000	115.000000	625.000000	528.000000	57.000000	16018.500000	614.000000	142.500000	53.000000
75%	630.750000	134.000000	678.000000	605.250000	71.750000	17773.250000	785.000000	182.750000	82.000000
max	810.000000	157.000000	956.000000	808.000000	104.000000	24806.000000	1089.000000	338.000000	117.000000

PART B: Draw Stock Price Graph(Stock Price vs Time) for any 2 given stocks with inference



Assuming "SAIL" refers to the number of cases: There seems to be a positive correlation between the number of cases and the year. This would mean that the number of cases has been increasing over time from 2014 to 2021.

Alternative interpretation: If "SAIL" does not represent the number of cases, it's possible there's a different relationship between the variable on the y-axis and the year.



based on the general shape of the line, we can observe some trends:

- There appear to be periods of both price increases and price decreases.
- The price seems to be more volatile in some parts of the time series compared to others

PART B: Calculate Returns for all stocks with inference

Infosys Indian_Hotel Mahindra_and_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

The dataset comprises 314 rows and 10 columns, representing the log returns of stock prices for ten different companies over a certain period. Log returns offer valuable insights into the daily percentage changes in stock prices for each company. Negative log returns signify a decline in stock prices, whereas positive log returns indicate an increase. It's worth noting that the log return for the first day is NaN since there is no previous day's data available to compute the return.

PART B: Calculate Stock Means and Standard Deviation for all stocks with inference

Means of stocks

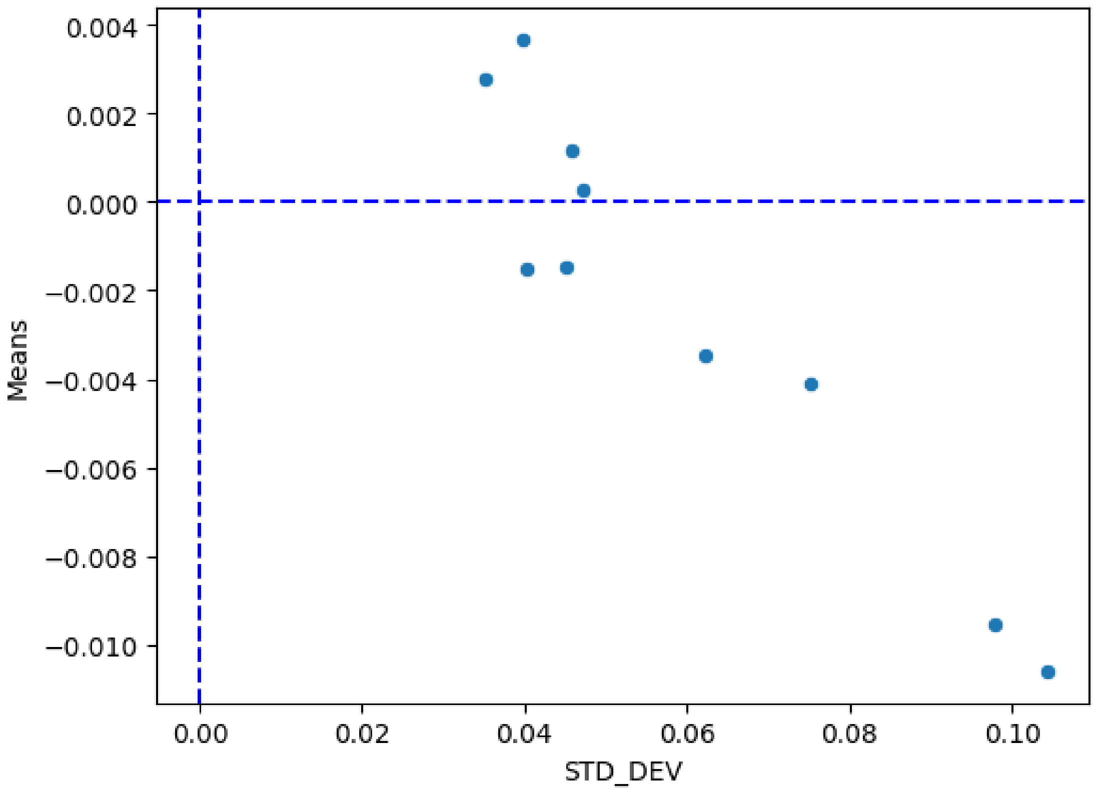
```
Infosys                   0.00279
Indian_Hotel              0.00027
Mahindra_and_Mahindra    -0.00151
Axis_Bank                 0.00117
SAIL                     -0.00346
Shree_Cement              0.00368
Sun_Pharma               -0.00145
Jindal_Steel             -0.00412
Idea_Vodafone            -0.01061
Jet_Airways              -0.00955
dtype: float64
```

Std_dev of all stocks

```
Infosys                   0.03507
Indian_Hotel              0.04713
Mahindra_and_Mahindra     0.04017
Axis_Bank                 0.04583
SAIL                      0.06219
Shree_Cement              0.03992
Sun_Pharma                0.04503
Jindal_Steel              0.07511
Idea_Vodafone             0.10432
Jet_Airways               0.09797
dtype: float64
```

The average returns offer an understanding of the typical daily performance exhibited by each stock, with certain stocks displaying gains while others exhibit losses. Meanwhile, the standard deviations serve as a measure of the volatility or uncertainty linked to each stock, wherein elevated standard deviations signify greater price fluctuations experienced by the stock over time.

PART B: Draw a plot of Stock Means vs Standard Deviation and state your inference



The scatter plot visually illustrates the relationship between mean returns and volatility across different companies. Notably, Infosys and Shree Cement emerge as the top performers, boasting the highest mean returns among the companies analyzed. Conversely, Idea Vodafone and Jet Airways register as the least profitable options, exhibiting the lowest mean returns.

Delving deeper into the analysis, it's discerned that Shree Cement, despite sharing the spotlight with Infosys in terms of mean returns, showcases slightly lower volatility. This characteristic suggests that Shree Cement presents a comparatively more stable investment avenue when juxtaposed with Infosys. Moreover, when scrutinizing the two companies with the lowest mean returns, Jet Airways emerges with marginally lower volatility compared to Idea Vodafone.

In essence, the scatter plot not only identifies the standout performers and laggards in terms of mean returns but also sheds light on the varying levels of volatility associated with these companies. This nuanced understanding can aid investors in making informed decisions tailored to their risk tolerance and investment objectives.

PART B: Conclusions and Recommendations

In analyzing the stock data across various companies, several notable insights emerge. Particularly, Infosys and Shree Cement emerge as standout options, showcasing the highest average returns, suggesting promising investment prospects. Conversely, Idea Vodafone and Jet Airways exhibit the lowest average returns, signaling a need for caution when considering these entities for investment ventures.

For investors seeking to optimize their profitability, it is advisable to explore opportunities within companies like Infosys and Shree Cement. These firms consistently demonstrate superior returns over the observed period, presenting attractive investment avenues for individuals aiming to maximize their financial gains.

Conversely, investors inclined towards risk aversion should exercise prudence when contemplating investments in companies such as Idea Vodafone and Jet Airways, given their track record of lower mean returns.

Furthermore, it's essential to acknowledge the presence of short-term fluctuations inherent in the market. Holding onto investments for extended durations can serve as a strategy to mitigate the impact of market volatility and potentially yield higher returns over time.

Regular monitoring of investment performance and staying abreast of market trends are indispensable practices for investors. Remaining informed enables investors to make well-informed decisions and adapt their strategies to evolving market conditions effectively.

*** THE END ***