FRA PROJECT

PART- A

FASNA

21/06/24

CONTENTS:

FIGURES:

TABLES:

Problem Statement

Context In the realm of modern finance, businesses encounter the perpetual challenge of managing debt obligations effectively to maintain a favorable credit standing and foster sustainable growth. Investors keenly scrutinize companies capable of navigating financial complexities while ensuring stability and profitability. A pivotal instrument in this evaluation process is the balance sheet, which provides a comprehensive overview of a company's assets, liabilities, and shareholder equity, offering insights into its financial health and operational efficiency. In this context, leveraging available financial data, particularly from preceding fiscal periods, becomes imperative for informed decision-making and strategic planning.

Objective

A group of venture capitalists want to develop a Financial Health Assessment Tool. With the help of the tool, it endeavors to empower businesses and investors with a robust mechanism for evaluating the financial well-being and creditworthiness of companies. By harnessing machine learning techniques, they aim to analyze historical financial statements and extract pertinent insights to facilitate informed decision-making via the tool. Specifically, they foresee facilitating the following with the help of the tool:

Debt Management Analysis: Identify patterns and trends in debt management practices to assess the ability of businesses to fulfill financial obligations promptly and efficiently, and identify potential cases of default. Credit Risk Evaluation: Evaluate credit risk exposure by analyzing liquidity ratios, debt-to-equity ratios, and other key financial indicators to ascertain the likelihood of default and inform investment decisions. They have hired you as a data scientist and provided you with the financial metrics of different companies. The task is to analyze the data provided and develop a predictive model leveraging machine learning techniques to identify whether a given company will be tagged as a defaulter in terms of net worth next year. The predictive model will help the organization anticipate potential challenges with the financial performance of the companies and enable proactive risk mitigation strategies.

Data Dictionary

The data consists of financial metrics from the balance sheets of different companies. The detailed data dictionary is given below.

- **Networth Next Year**: Net worth of the customer in the next year
- **Total assets**: Total assets of customer
- **Net worth**: Net worth of the customer of the present year
- **Total income**: Total income of the customer
- **Change in stock**: Difference between the current value of the stock and the value of stock in the last trading day
- **Total expenses**: Total expenses done by the customer

- **Profit after tax**: Profit after tax deduction
- **PBDITA**: Profit before depreciation, income tax, and amortization
- **PBT**: Profit before tax deduction
- **Cash profit**: Total Cash profit
- **PBDITA as % of total income**: PBDITA / Total income
- **PBT as % of total income**: PBT / Total income
- **PAT as % of total income**: PAT / Total income
- **Cash profit as % of total income**: Cash Profit / Total income
- **PAT as % of net worth**: PAT / Net worth
- **Sales**: Sales done by the customer
- **Income from financial services**: Income from financial services
- **Other income**: Income from other sources
- **Total capital**: Total capital of the customer
- **Reserves and funds**: Total reserves and funds of the customer
- **Borrowings**: Total amount borrowed by the customer
- **Current liabilities & provisions**: current liabilities of the customer
- **Deferred tax liability**: Future income tax customer will pay because of the current transaction
- **Shareholders funds**: Amount of equity in a company which belongs to shareholders
- **Cumulative retained profits**: Total cumulative profit retained by customer
- **Capital employed**: Current asset minus current liabilities
- **TOL/TNW**: Total liabilities of the customer divided by Total net worth
- **Total term liabilities / tangible net worth**: Short + long term liabilities divided by tangible net worth
- **Contingent liabilities / Net worth (%)**: Contingent liabilities / Net worth
- **Contingent liabilities**: Liabilities because of uncertain events
- **Net fixed assets**: The purchase price of all fixed assets
- **Investments**: Total invested amount
- **Current assets**: Assets that are expected to be converted to cash within a year
- **Net working capital**: Difference between the current liabilities and current assets
- **Quick ratio (times)**: Total cash divided by current liabilities
- **Current ratio (times)**: Current assets divided by current liabilities
- **Debt to equity ratio (times)**: Total liabilities divided by its shareholder equity
- **Cash to current liabilities (times)**: Total liquid cash divided by current liabilities
- **Cash to average cost of sales per day**: Total cash divided by the average cost of the sales
- **Creditors turnover**: Net credit purchase divided by average trade creditors
- **Debtors turnover**: Net credit sales divided by average accounts receivable
- **Finished goods turnover**: Annual sales divided by average inventory
- **WIP turnover**: The cost of goods sold for a period divided by the average inventory for that period
- **Raw material turnover**: Cost of goods sold is divided by the average inventory for the same period
- **Shares outstanding**: Number of issued shares minus the number of shares held in the company
- **Equity face value**: cost of the equity at the time of issuing
- **EPS**: Net income divided by the total number of outstanding share
- **Adjusted EPS**: Adjusted net earnings divided by the weighted average number of common shares outstanding on a diluted basis during the plan year
- **Total liabilities**: Sum of all types of liabilities
- **PE on BSE**: Company's current stock price divided by its earnings per share

Note: A company will not be tagged as a defaulter if its net worth next year is positive, or else, it'll be tagged as a defaulter.

1: Define the problem and perform Exploratory Data Analysis

- Problem definition - Check shape, Data types, and statistical summary - Univariate analysis - Multivariate analysis - Use appropriate visualizations to identify the patterns and insights - Key meaningful observations on individual variables and the relationship between variables

#READ DATA FILE

#FIRST 5 ROWS

	Num	Networth Next Year	Total assets	Net worth	Total income	Change in stock	Total expenses	Profit after tax	PBDITA	PBT	...	Debtors turnover	Finished goods turnover	WIP turnover	Raw material turnover	Shares outstanding	Equity face value	EPS	Adjusted EPS	Total liabilities	PE on BSE
0	1	395.3	827.6	336.5	534.1	13.5	508.7	38.9	124.4	64.6	...	5.65	3.99	3.37	14.87	8760056.0	10.0	4.44	4.44	827.6	NaN
1	2	36.2	67.7	24.3	137.9	-3.7	131.0	3.2	5.5	1.0	...	NaN	NaN	NaN	NaN	NaN	NaN	0.00	0.00	67.7	NaN
2	3	84.0	238.4	78.9	331.2	-18.1	309.2	3.9	25.8	10.5	...	2.51	17.67	8.76	8.35	NaN	NaN	0.00	0.00	238.4	NaN
3	4	2041.4	6883.5	1443.3	8448.5	212.2	8482.4	178.3	418.4	185.1	...	1.91	18.14	18.62	11.11	10000000.0	10.0	17.60	17.60	6883.5	NaN
4	5	41.8	90.9	47.0	388.6	3.4	392.7	-0.7	7.2	-0.6	...	68.00	45.87	28.67	19.93	107315.0	100.0	-6.52	-6.52	90.9	NaN

Table-1-FIRST 5 ROWS

# Check the shape of the dataset

4256 rows, 51 columns

```
# Check Data types
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4256 entries, 0 to 4255
Data columns (total 51 columns):
 #   Column                                      Non-Null Count  Dtype
---  ------                                      --------------  -----
 0   Num                                         4256 non-null   int64
 1   Networth Next Year                          4256 non-null   float64
 2   Total assets                                4256 non-null   float64
 3   Net worth                                   4256 non-null   float64
 4   Total income                                4025 non-null   float64
 5   Change in stock                             3706 non-null   float64
 6   Total expenses                              4091 non-null   float64
 7   Profit after tax                            4102 non-null   float64
 8   PBDITA                                      4102 non-null   float64
 9   PBT                                         4102 non-null   float64
 10  Cash profit                                 4102 non-null   float64
 11  PBDITA as % of total income                 4177 non-null   float64
 12  PBT as % of total income                    4177 non-null   float64
 13  PAT as % of total income                    4177 non-null   float64
 14  Cash profit as % of total income            4177 non-null   float64
 15  PAT as % of net worth                       4256 non-null   float64
 16  Sales                                       3951 non-null   float64
 17  Income from fincial services                3145 non-null   float64
 18  Other income                                2700 non-null   float64
 19  Total capital                               4251 non-null   float64
 20  Reserves and funds                          4158 non-null   float64
 21  Borrowings                                  3825 non-null   float64
 22  Current liabilities & provisions            4146 non-null   float64
 23  Deferred tax liability                      2887 non-null   float64
 24  Shareholders funds                          4256 non-null   float64
 25  Cumulative retained profits                 4211 non-null   float64
 26  Capital employed                            4256 non-null   float64
 27  TOL/TNW                                     4256 non-null   float64
 28  Total term liabilities / tangible net worth 4256 non-null   float64
 29  Contingent liabilities / Net worth (%)      4256 non-null   float64
 30  Contingent liabilities                      2854 non-null   float64
 31  Net fixed assets                            4124 non-null   float64
 32  Investments                                 2541 non-null   float64
 33  Current assets                              4176 non-null   float64
 34  Net working capital                         4219 non-null   float64
 35  Quick ratio (times)                         4151 non-null   float64
 36  Current ratio (times)                       4151 non-null   float64
 37  Debt to equity ratio (times)                4256 non-null   float64
```

```
38  Cash to current liabilities (times)         4151 non-null   float64
39  Cash to average cost of sales per day        4156 non-null   float64
40  Creditors turnover                        3865 non-null   float64
41  Debtors turnover                          3871 non-null   float64
42  Finished goods turnover                      3382 non-null   float64
43  WIP turnover                            3492 non-null   float64
44  Raw material turnover                        3828 non-null   float64
45  Shares outstanding                          3446 non-null   float64
46  Equity face value                         3446 non-null   float64
47  EPS                                 4256 non-null   float64
48  Adjusted EPS                            4256 non-null   float64
49  Total liabilities                        4256 non-null   float64
50  PE on BSE                             1629 non-null   float64
dtypes: float64(50), int64(1)
memory usage: 1.7 MB
50 float data type,1 integer data type
```

#statistical summary

	Num	Networth Next Year	Total assets	Net worth	Total income	Change in stock	Total expenses	Profit after tax	PBDITA	PBT	...	Debtors turnover	Finished goods turnover	WIP turnover	Raw material turnover	Shares outstanding	Equity face value	EPS	Adjusted EPS	Total liabilities	PE on BSE
count	4256.00	4256.00	4256.00	4256.00	4025.00	3706.00	4091.00	4102.00	4102.00	4102.00	...	3871.00	3382.00	3492.00	3828.00	3446.00	3446.00	4256.00	4256.00	4256.00	1629.00
mean	2128.50	1344.74	3573.62	1351.95	4688.19	43.70	4356.30	295.05	605.94	410.26	...	17.93	84.37	28.68	17.73	23764909.56	-1094.83	-196.22	-197.53	3573.62	55.46
std	1228.75	15936.74	30074.44	12961.31	53918.95	436.92	51398.09	3079.90	5646.23	4217.42	...	90.16	562.64	169.65	343.13	170979041.33	34101.36	13061.95	13061.93	30074.44	1304.45
min	1.00	-74265.60	0.10	0.00	0.00	-3029.40	-0.10	-3908.30	-440.70	-3894.80	...	0.00	-0.09	-0.18	-2.00	-2147483647.00	-999998.90	-843181.82	-843181.82	0.10	-1116.64
25%	1064.75	3.98	91.30	31.48	107.1	-1.8	96.80	0.50	6.93	0.80	...	3.81	8.19	5.10	3.02	130838	10.00	0.00	0.00	91.30	2.97

	Num	Networth Next Year	Total assets	Net worth	Total income	Change in stock	Total expenses	Profit after tax	PBDITA	PBT	...	Debtors turnover	Finished goods turnover	WIP turnover	Raw material turnover	Shares outstanding	Equity face value	EPS	Adjusted EPS	Total liabilities	PE on BSE
					0	0					.					2.50					
50%	2128.50	72.10	315.50	104.80	455.10	1.60	428.80	9.00	36.90	12.60	...	6.47	17.32	9.86	6.41	4750000.00	10.00	1.49	1.24	315.50	8.69
75%	3192.25	330.82	1120.80	389.85	1485.00	18.40	1395.70	53.30	158.70	74.17	...	11.85	40.01	20.24	11.82	10906020.00	10.00	10.00	7.62	1120.80	17.00
max	4256.00	805773.40	1176509.20	613151.60	2442828.20	14185.50	2366035.30	119439.10	208576.50	145292.60	...	3135.20	17947.60	5651.40	21092.00	4130400545.00	100000.00	34522.53	34522.53	1176509.20	51002.74

Table-2-***statistical summary***

# Replace spaces with underscores in column names

#drop “num “,”Equity_face_value”colum

#check duplicate values

Data set have 2 duplicate values

#drop duplicate values

#after drop duplcate the shape of data set

(4254, 49)

#CHECK NULL VALUES

Percentange of missing values

```
Networth_Next_Year              0.00
Total_assets                0.00
Net_worth                   0.00
Total_income                  0.05
Change_in_stock                 0.13
Total_expenses                0.04
```

```
Profit_after_tax                      0.04
PBDITA                               0.04
PBT                                0.04
Cash_profit                           0.04
PBDITA_as_perc_of_total_income                  0.02
PBT_as_perc_of_total_income                   0.02
PAT_as_perc_of_total_income                   0.02
Cash_profit_as_perc_of_total_income              0.02
PAT_as_perc_of_net_worth                     0.00
Sales                              0.07
Income_from_fincial_services                 0.26
Other_income                            0.37
Total_capital                         0.00
Reserves_and_funds                        0.02
Borrowings                            0.10
Current_liabilities_and_provisions            0.03
Deferred_tax_liability                  0.32
Shareholders_funds                        0.00
Cumulative_retained_profits                  0.01
Capital_employed                         0.00
TOL_by_TNW                              0.00
Total_term_liabilities__by__tangible_net_worth   0.00
Contingent_liabilities__by__Net_worth_perc       0.00
Contingent_liabilities                  0.33
Net_fixed_assets                      0.03
Investments                           0.40
Current_assets                          0.02
Net_working_capital                     0.01
Quick_ratio_times                     0.02
Current_ratio_times                     0.02
Debt_to_equity_ratio_times                0.00
Cash_to_current_liabilities_times           0.02
Cash_to_average_cost_of_sales_per_day        0.02
Creditors_turnover                     0.09
Debtors_turnover                       0.09
Finished_goods_turnover                  0.20
WIP_turnover                           0.18
Raw_material_turnover                   0.10
Shares_outstanding                     0.19
EPS                               0.00
Adjusted_EPS                         0.00
Total_liabilities                     0.00
PE_on_BSE                            0.62
dtype: float64
```

```
#drop columns null % greater than 30
Drop 'Other_income',
  'Deferred_tax_liability',
  'Contingent_liabilities',
  'Investments',
  'PE_on_BSE'
Theses columns


# Verify the shape of the DataFrame after dropping columns
```

```
Shape of the DataFrame after dropping columns: (4254, 44)
```

```
#create target column
```

A company will not be tagged as a defaulter if its net worth next year is positive, or else, it'll be tagged as a defaulter.

	defaulter	Networth_Next_Year
0	no	395.30
1	no	36.20
2	no	84.00
3	no	2041.40
4	no	41.80
5	no	291.50
6	no	93.30
7	no	985.10
8	no	188.60
9	no	229.60

'table-3-defaulter vs 'Networth_Next_Year'

```
value_counts of defaulter column
```

```
no     3350
yes     904

Percentage value count
no    0.79
yes   0.21
```

```
#We should inspect total missing values by each row.
```

```
0       0
```

```
1        7
2        1
3        0
4        0
        ..
4251    19
4252     0
4253     0
4254     1
4255     0
Length: 4254, dtype: int64
```

```
#Let's filter the data which is more than 90% complete at the row level
```

```
Networth_Next_Year                                  0
Total_assets                                        0
Net_worth                                           0
Total_income                                        0
Change_in_stock                                   215
Total_expenses                                      0
Profit_after_tax                                    0
PBDITA                                              0
PBT                                                 0
Cash_profit                                         0
PBDITA_as_perc_of_total_income                      0
PBT_as_perc_of_total_income                         0
PAT_as_perc_of_total_income                         0
Cash_profit_as_perc_of_total_income                 0
PAT_as_perc_of_net_worth                            0
Sales                                              35
Income_from_fincial_services                      738
Total_capital                                       1
Reserves_and_funds                                  2
Borrowings                                        250
Current_liabilities_and_provisions                  1
Shareholders_funds                                  0
Cumulative_retained_profits                         1
Capital_employed                                    0
TOL_by_TNW                                          0
Total_term_liabilities__by__tangible_net_worth      0
Contingent_liabilities__by__Net_worth_perc          0
Net_fixed_assets                                   13
Current_assets                                      0
Net_working_capital                                 0
Quick_ratio_times                                   1
Current_ratio_times                                 1
Debt_to_equity_ratio_times                          0
Cash_to_current_liabilities_times                   1
Cash_to_average_cost_of_sales_per_day               0
Creditors_turnover                                146
Debtors_turnover                                  139
Finished_goods_turnover                           409
```

```
WIP_turnover                    303
Raw_material_turnover             198
Shares_outstanding                534
EPS                            0
Adjusted_EPS                       0
Total_liabilities                0
defaulter                       0
dtype: int64
```

Missing value treatment

```
# Replace all NaN values with the median of their respective columns
```

```
Networth_Next_Year                         0
Total_assets                        0
Net_worth                           0
Total_income                         0
Change_in_stock                        0
Total_expenses                       0
Profit_after_tax                    0
PBDITA                            0
PBT                            0
Cash_profit                          0
PBDITA_as_perc_of_total_income                 0
PBT_as_perc_of_total_income                 0
PAT_as_perc_of_total_income                 0
Cash_profit_as_perc_of_total_income              0
PAT_as_perc_of_net_worth                  0
Sales                            0
Income_from_fincial_services                0
Total_capital                     0
Reserves_and_funds                     0
Borrowings                      0
Current_liabilities_and_provisions            0
Shareholders_funds                   0
Cumulative_retained_profits                 0
Capital_employed                   0
TOL_by_TNW                      0
Total_term_liabilities__by__tangible_net_worth   0
Contingent_liabilities__by__Net_worth_perc      0
Net_fixed_assets                  0
Current_assets                   0
Net_working_capital                  0
Quick_ratio_times                  0
Current_ratio_times                  0
Debt_to_equity_ratio_times                 0
Cash_to_current_liabilities_times             0
Cash_to_average_cost_of_sales_per_day          0
Creditors_turnover                     0
Debtors_turnover                       0
```

```
Finished_goods_turnover                 0
WIP_turnover                    0
Raw_material_turnover                   0
Shares_outstanding                   0
EPS                           0
Adjusted_EPS                     0
Total_liabilities                  0
defaulter                         0
dtype: int64
```

EDA

```
#Univariate analysis
# after checking outliers
```

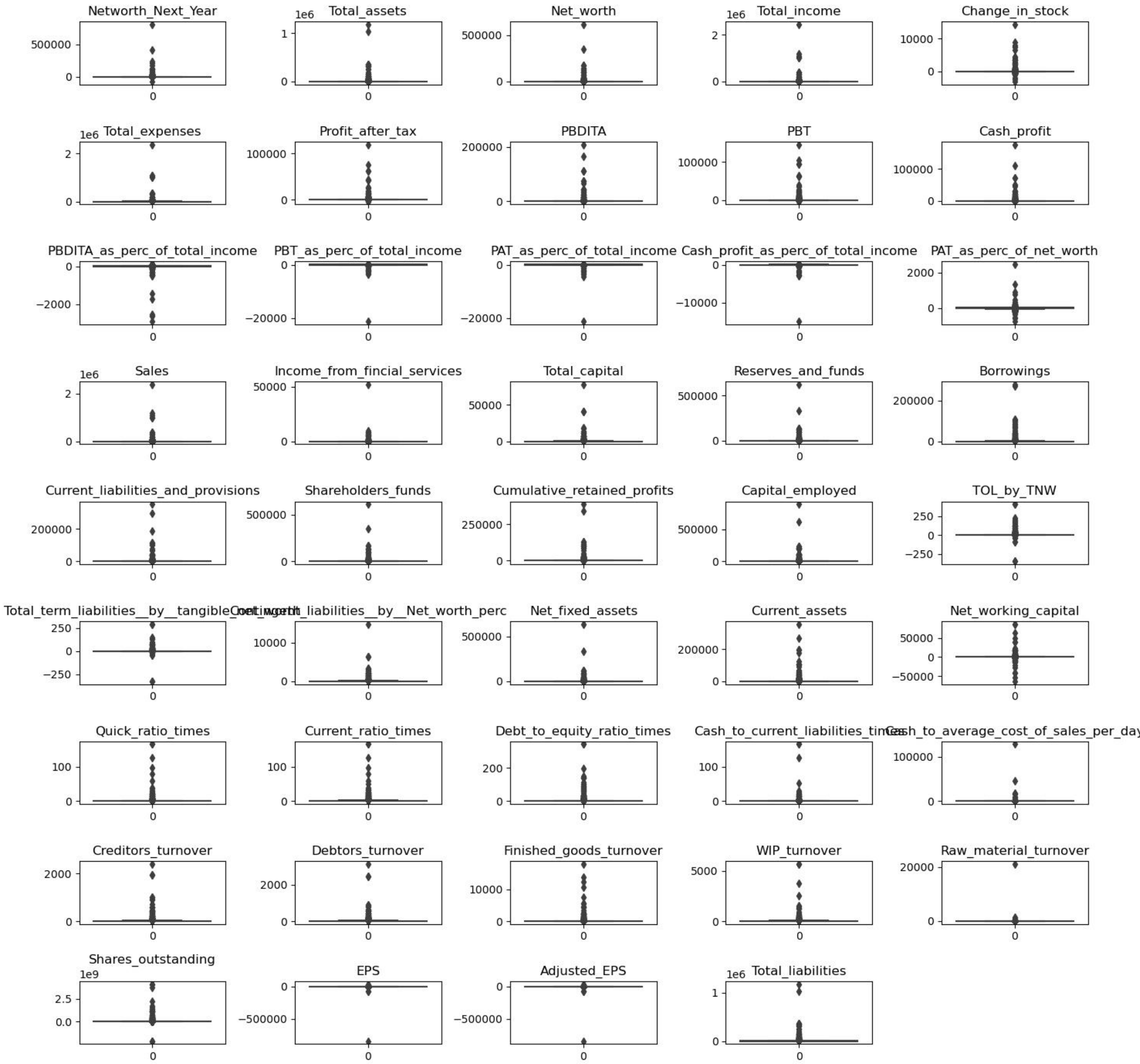


Figure-1-boxplot of data set

all columns have outliers.

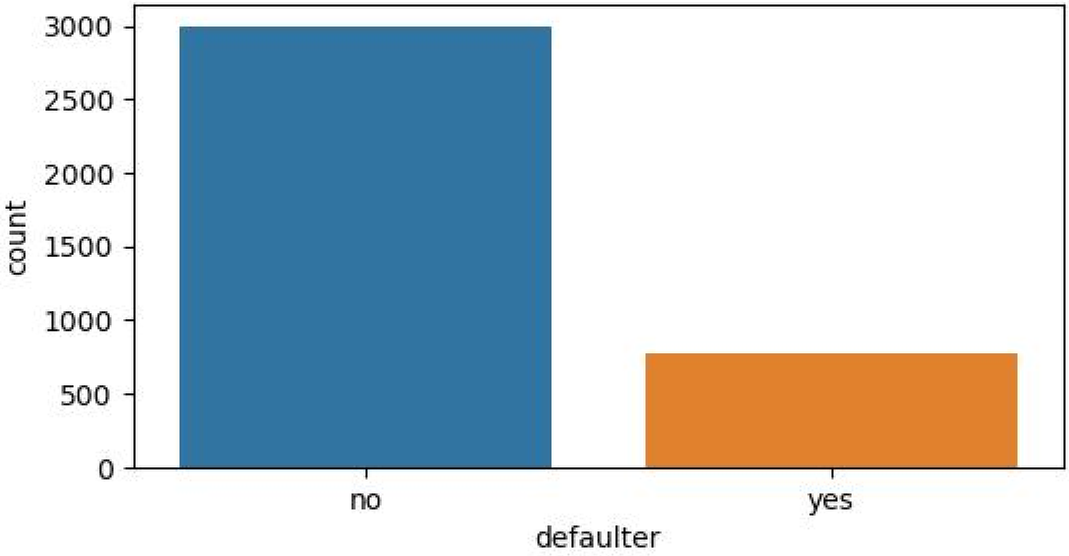


Figure-2-Count plot of Defaulter column

more costomers are non defaulters, 21% defaulters.

```
# Analyzing the target variable 'Networth Next Year'
```

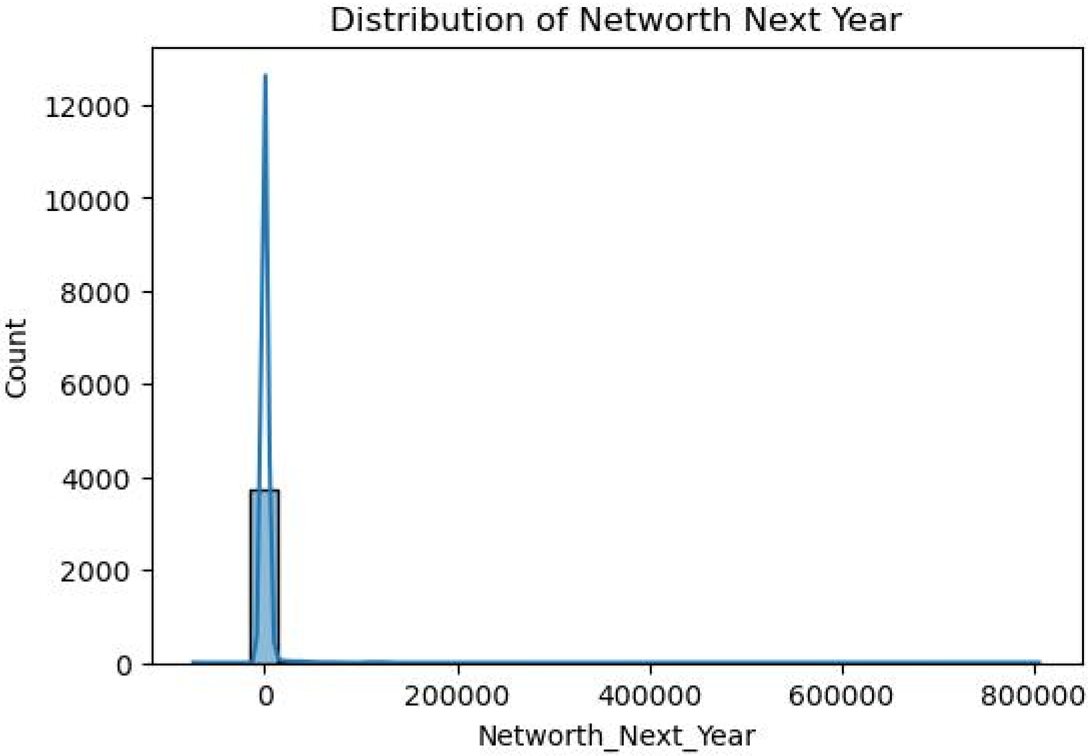


Figure-3-histogram of networth next year

75% values are low,max networth of next year is 805773

```
# Correlation matrix to identify relationships between numerical features
```

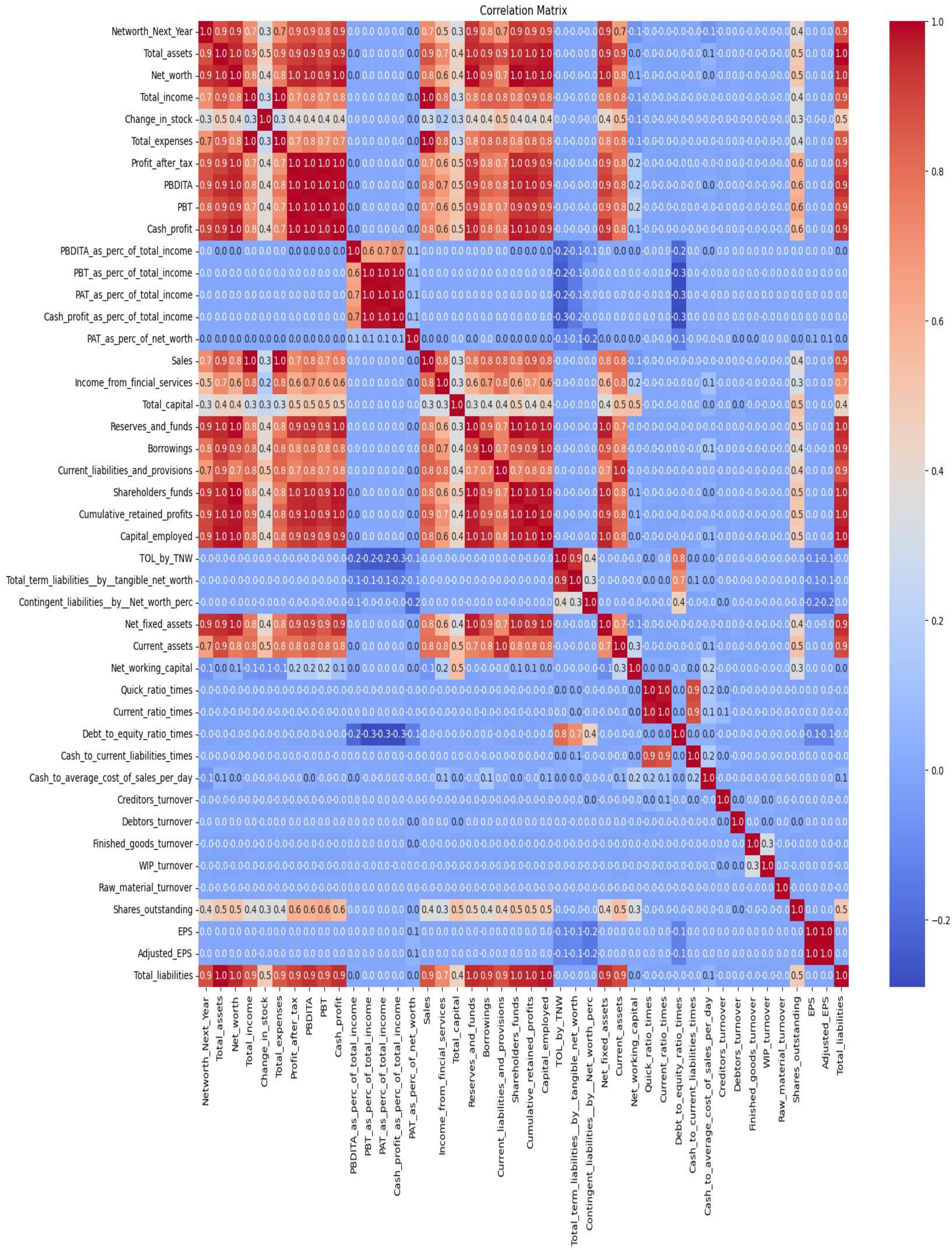


Figure-4-heatmap

Complex Relationships:

The presence of both positive and negative correlations suggests intricate dependencies. Further analysis could reveal hidden patterns or opportunities.

Strong Negative Correlations:

more variables are negative correlation. Some variables show negative correlations. For instance, when one metric rises, another falls. Understanding these inverse relationships is crucial for hedging strategies.
we can seen multi colinarity

```
# Pairplot to visualize relationships between selected features
```

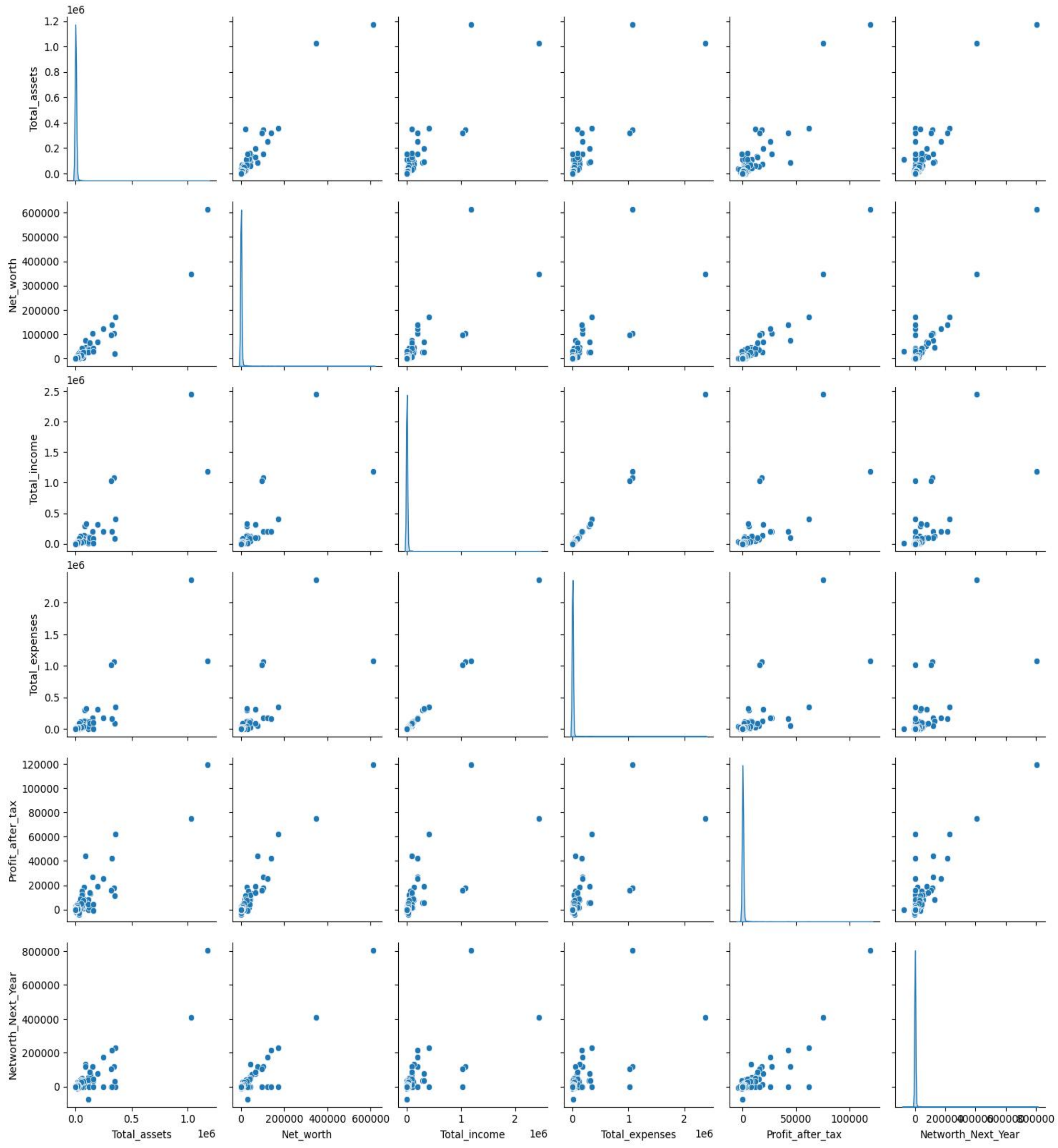


Figure-5-pairplot

INSIGHTS

1.

Net Worth vs. Total Assets:

2.

- A positive correlation between net worth and total assets suggests that as assets increase, net worth tends to rise as well.

- This aligns with the concept that net worth is influenced by the value of owned assets.

3.

Total Income vs. Total Expenses:

4.

- The relationship between total income and total expenses is essential for financial management.
- If expenses consistently exceed income, it could impact profitability.

5.

Profit After Tax vs. Net Worth Next Year:

6.

- Examining the relationship between profit after tax and projected net worth for the next year can reveal growth potential.
- Positive profit after tax contributes to net worth growth.

7.

Distribution of Features (Diagonal Plots):

8.

- The diagonal plots show the distribution of each individual feature.
- For example, the KDE (Kernel Density Estimation) plot for net worth next year indicates its distribution.

all are positive correlation b/w each other.

```
# Analyzing the relationship between 'Net worth' and 'Networth Next Year'
```

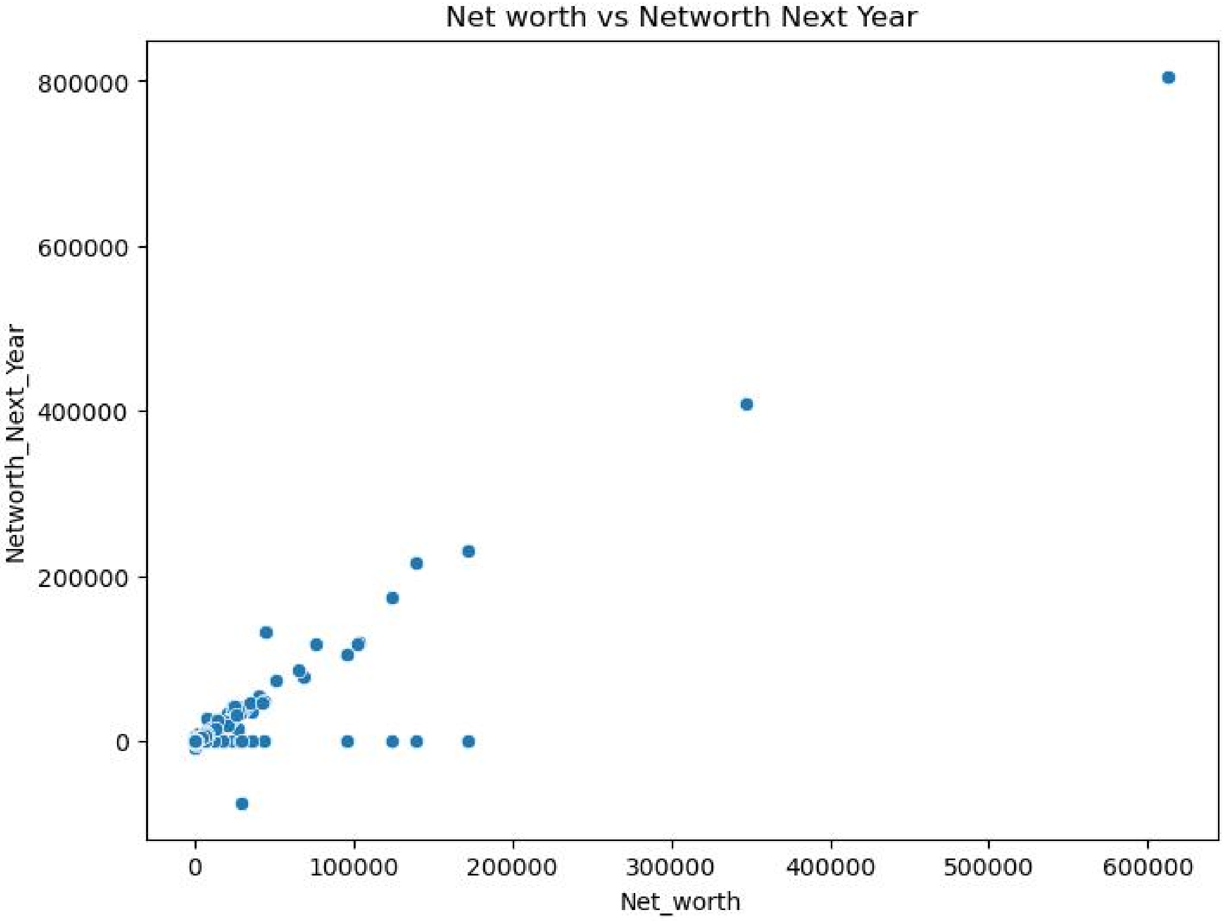


Figure-6-Scatter plot of'Net worth' and 'Networth Next Year'

Positive Correlation:

- The majority of data points cluster in the lower range of both "Net worth" and "Networth Next Year."
- As "Net worth" increases, there's a tendency for "Networth Next Year" to increase as well.
- This suggests a positive correlation between an individual's current net worth and their net worth in the following year.

Outlier:

- Notice the outlier at the top right corner of the plot.
- This individual has a significantly higher net worth both currently and in the next year compared to others.
- Investigating this outlier could provide valuable insights into factors driving substantial financial growth.
-

```
# Analyzing the relationship between 'Total assets' and 'Networth Next Year
```

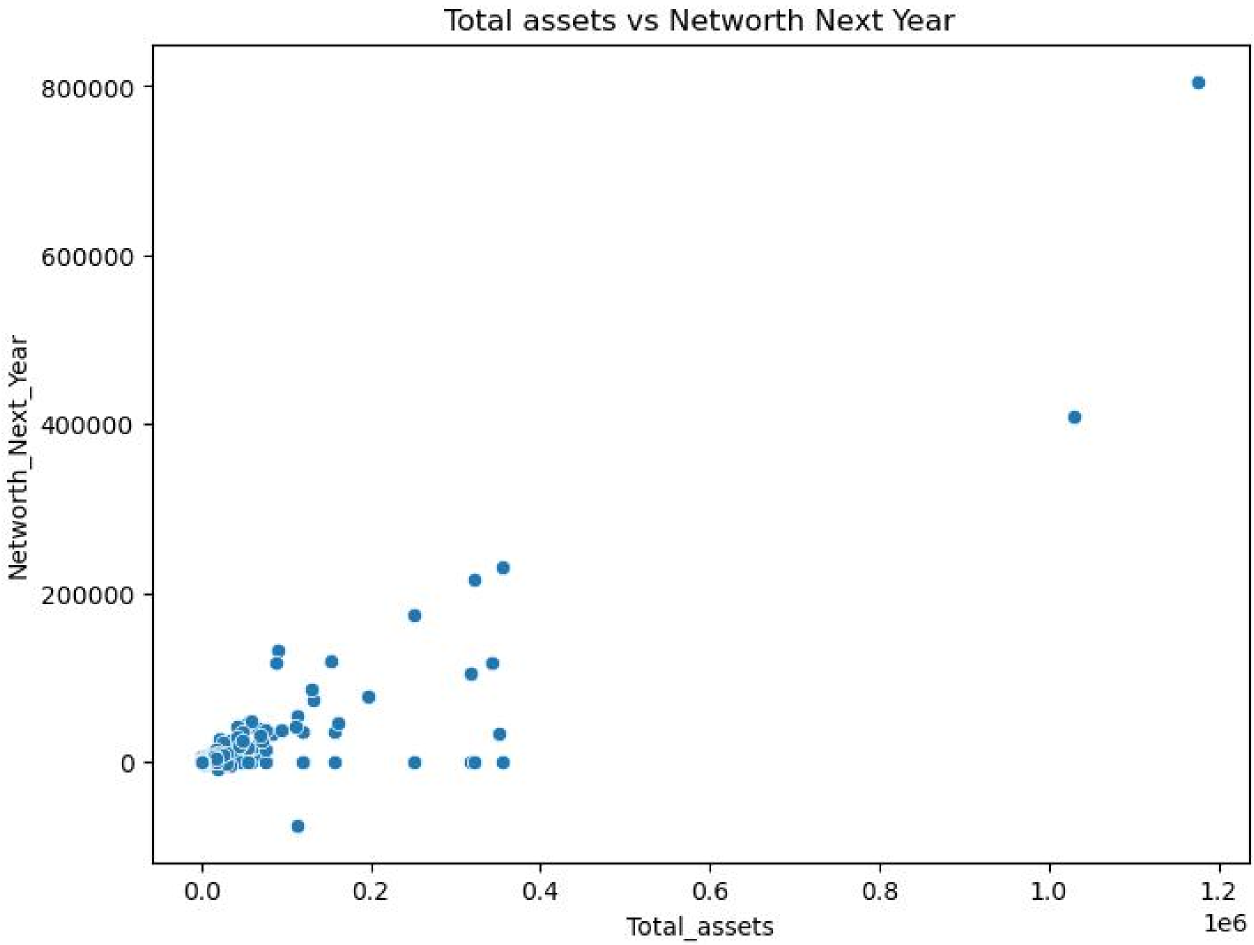


Figure-7-Scatter plot of''Total assets' and 'Networth Next Year

Positive Correlation:

- The scatter plot shows data points that suggest a positive correlation between "Total assets" and "Networth Next Year."
- As "Total assets" increase, there's a tendency for "Networth Next Year" to increase as well.

Outliers:

- Observe any outliers in the plot.
- These points may represent individuals with exceptional financial situations.
- Investigating outliers can provide valuable insights.

Individual Variables

Networth Next Year:

- The distribution is roughly normal with a mean around 2000.
- There are some outliers indicating a few companies with significantly higher projected net worth.

Total Assets:

- The distribution shows high variability with extreme values, indicating some companies have very large total assets.
- The presence of significant outliers suggests a few companies dominate in terms of total assets.

Net Worth:

- The distribution is wide, indicating a diverse range of company net worth.
- Extreme outliers exist, indicating some companies have exceptionally high net worth.

Total Income and Total Expenses:

- Both distributions show a reasonable spread with some outliers.
- This suggests a range of income and expense levels across companies, with a few having very high income or expenses.

Profit After Tax:

- High variance indicates different levels of profitability among companies.
- Some extreme values suggest a few companies have exceptionally high profits.

Shares Outstanding:

- The distribution has very high variance, indicating different scales of companies in terms of shares.
- A few companies have extremely high shares outstanding, which could be indicative of large public corporations.

EPS (Earnings Per Share):

- Negative values indicate that some companies are experiencing losses.
- A wide spread with extreme outliers shows significant variability in company performance.

Total Liabilities:

- The spread is similar to net worth, indicating a range of liability levels across companies.
- Some outliers suggest a few companies have exceptionally high liabilities.

Relationships Between Variables

Net Worth vs. Total Assets:

- A positive correlation suggests that as total assets increase, net worth tends to rise.
- This is expected, as higher assets contribute to higher net worth.

Total Income vs. Total Expenses:

- A strong positive correlation indicates that companies with higher income also tend to have higher expenses.
- This could be due to larger companies having higher operational costs.

Profit After Tax vs. Net Worth Next Year:

- Positive correlation suggests that higher profit after tax contributes to an increase in net worth for the next year.
- This implies that profitability is a key driver of future net worth growth.

Correlation Matrix

Positive Correlations:

- Variables such as total assets, net worth, and net worth next year show positive correlations, indicating that they tend to move together.
- This is consistent with the expectation that higher assets and profitability contribute to higher net worth.

Negative Correlations:

- The presence of negative correlations, though not explicitly detailed, suggests inverse relationships between certain variables.
- For instance, higher expenses might negatively impact profit, leading to lower net worth growth.

Outliers

Impact of Outliers:

- Significant outliers in net worth, total assets, income, and other variables indicate that a few companies have extreme values.
- These outliers can skew the analysis and should be examined separately to understand the factors contributing to their exceptional values.

Insights from Outliers:

- Investigating the outliers can provide valuable insights into what drives exceptional financial performance or challenges.

- For instance, companies with extremely high net worth or assets might have unique business models, market positions, or operational efficiencies.

General Observations

Financial Health and Performance Patterns:

- The majority of companies cluster around lower ranges for most financial metrics, indicating that extreme values are not common.
- This clustering suggests that while some companies excel, many operate within a more modest financial range.

Distribution Patterns:

- The distribution plots reveal that most financial metrics follow a normal distribution with some skewness due to outliers.
- This indicates that while the central tendency is stable, variability exists among the companies.

Conclusion

- **Diverse Financial Landscape:** The dataset represents a diverse financial landscape with significant variability in key metrics.
- **Key Drivers:** Profitability, total assets, and income levels are key drivers of net worth and future financial health.
- **Outlier Analysis:** Detailed analysis of outliers can reveal insights into exceptional financial performance or challenges.
- **Strategic Implications:** Understanding the correlations and distributions can help in strategic planning, risk management, and identifying growth opportunities.

2 -: Data Pre-processing

Drop net worth next year

fPrepare the data for modeling: - Outlier Detection (treat, if needed) - Encode the data - Data split - Scale the data - Target variable creation * The target variable is default and should take the value 1 when net worth next year is negative & 0 when net worth next year is positive

```
#Outlier Detection (treat, if needed)
```

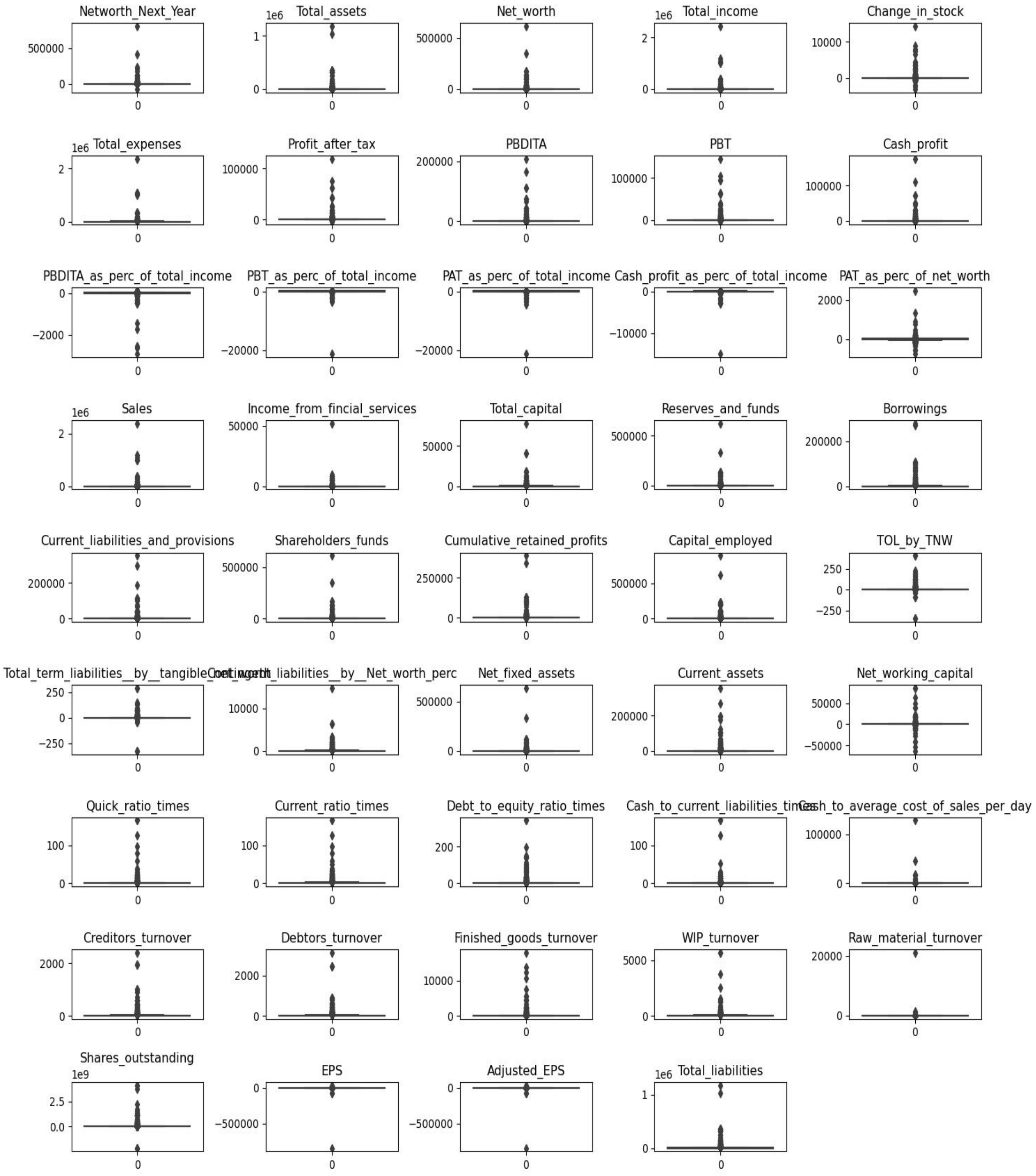


Figure-8-boxplot of data set

remove_outlier using IQR METHOD

```
# after checking outliers
```

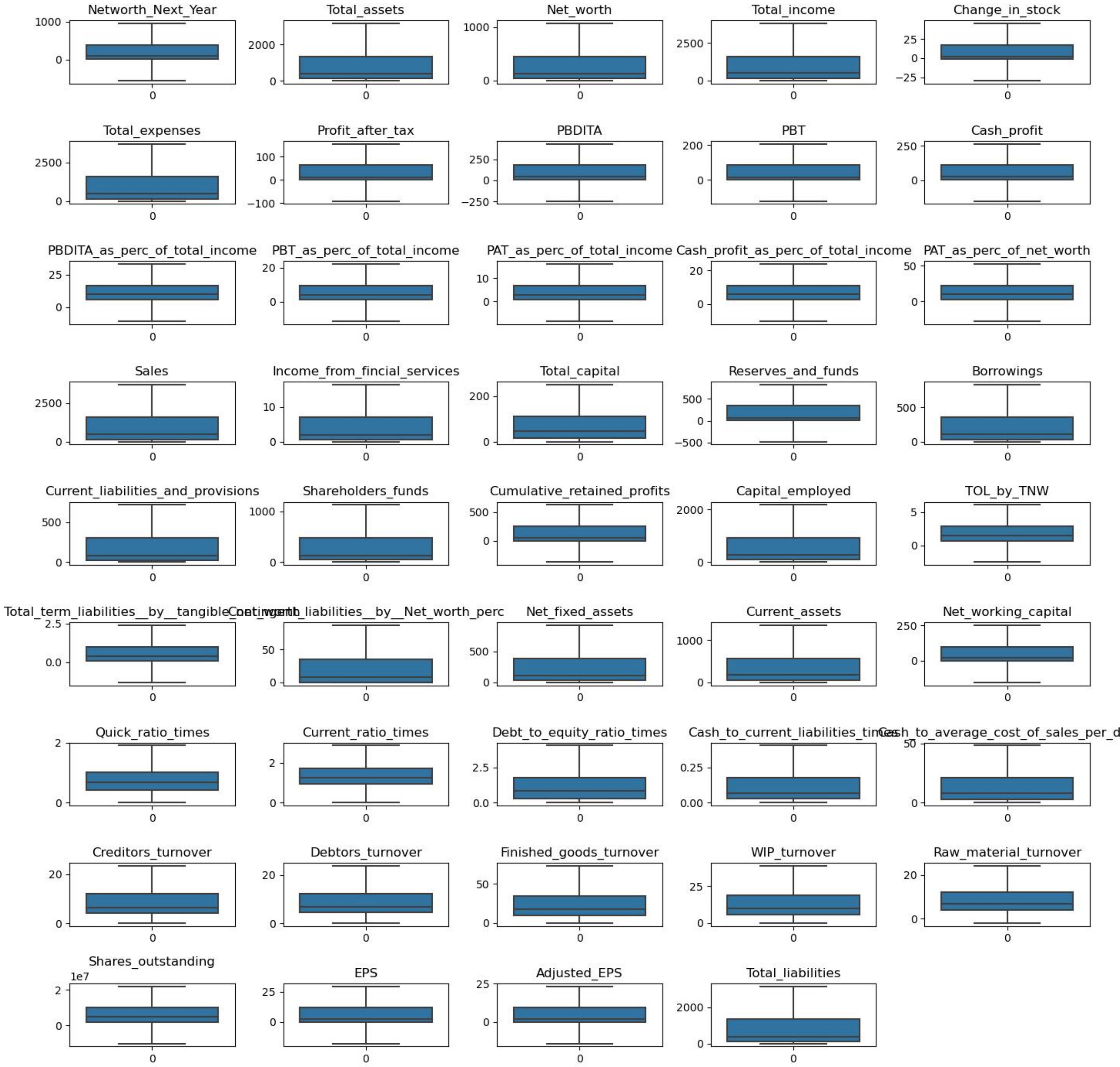


Figure-9-boxplot after outlier treatment

Encode the data

	Networth_Next_Year	Total_assets	Net_worth	Total_income	Change_in_stock	Total_expenses	Profit_after_tax	PBDITA	PBT	Cash_profit	...	Creditors_turnover	Debtors_turnover	Finished_goods_turnover	WIP_turnover	Raw_material_turnover	Shares_outstanding	EPS	Adjusted_EPS	Total_liabilities	default_yes
0	395.30	827.60	336.50	534.10	13.50	508.70	38.90	124.4	64.60	95.20	...	11.60	5.65	3.99	3.37	14.87	8760056.00	4.44	4.44	827.60	0

	Networth_Next_Year	Total_assets	Net_worth	Total_income	Change_in_stock	Total_expenses	Profit_after_tax	PBDITA	PBT	Cash_profit	...	Creditors_turnover	Debtors_turnover	Finished_goods_turnover	WIP_turnover	Raw_material_turnover	Shares_outstanding	EPS	Adjusted_EPS	Total_liabilities	defaulter_yes
								0													
2	84.00	238.40	78.90	331.20	-18.10	309.20	3.90	25.80	10.50	9.40	...	2.24	2.51	17.67	8.76	8.35	4800400.00	0.00	0.00	238.40	0
3	953.99	3173.76	1070.71	3803.34	45.22	3727.35	155.29	418.40	185.10	178.00	...	3.48	1.91	18.14	18.62	11.11	10000000.00	17.60	17.60	3173.76	0
4	41.80	90.90	47.00	388.60	3.40	392.70	-0.70	7.20	-0.60	3.90	...	21.67	23.67	45.87	28.67	19.93	107315.00	-6.52	-6.52	90.90	0
5	291.50	573.80	238.60	582.60	31.00	565.30	48.30	110.10	68.50	82.60	...	12.52	7.25	5.73	4.62	3.72	3807100.00	12.69	0.63	573.80	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
4250	336.50	455.20	197.80	199.20	1.80	193.30	5.90	59.10	6.70	35.90	...	1.62	2.12	17.49	9.94	4.09	8128880.00	0.73	0.73	455.20	0

Table-4-encoded data

NOW VALUE COUNT OF DEFAULTER COLUMN

0 2998
1 774

PECENTANGE VALUE COUNT OF DEFAULTER COLUMN

0 0.79

```
1   0.21
```

Split

```
# Split the data
X ---------------df after drop “ defaulter', "Networth_Next_Year" column
y ----------------'defaulter' column
```

Scale the data

```
#our target column is "defaulter"
```

3-Model Building

- Metrics of Choice (Justify the evaluation metrics) - Model Building (Logistic Regression, Random Forest) - Model performance check across different metrics

Metrics of Choice (Justify the evaluation metrics)

When evaluating the performance of a predictive model for identifying financial defaulters, it is crucial to use appropriate metrics that provide a comprehensive view of the model's effectiveness. Below are the key metrics chosen for evaluation and their justifications:

Accuracy

Definition: Accuracy is the proportion of correctly predicted instances (both true positives and true negatives) out of the total instances.

Justification: Accuracy is a straightforward metric that provides an overall performance measure of the model. However, it can be misleading if the classes are imbalanced (e.g., if there are significantly more non-defaulters than defaulters).

Precision

Definition: Precision is the proportion of true positive predictions out of the total positive predictions made by the model.

Justification: Precision is particularly important in this context because it measures the accuracy of the positive predictions (defaulters). High precision indicates that the model has a low false positive rate, which is crucial for reducing the cost and effort of wrongly tagging a non-defaulter as a defaulter.

Recall (Sensitivity)

Definition: Recall is the proportion of true positive predictions out of the total actual positives.

Justification: Recall measures the model's ability to identify all actual defaulters. High recall is essential for ensuring that most defaulters are correctly identified, which is critical for proactive risk mitigation.

F1-Score

Definition: The F1-score is the harmonic mean of precision and recall.

Justification: The F1-score balances precision and recall, providing a single metric that accounts for both false positives and false negatives. This is particularly useful when dealing with imbalanced datasets where focusing on one metric can be misleading. The F1-score is particularly relevant for ensuring that both the identification of actual defaulters and the avoidance of incorrectly tagging non-defaulters are balanced.

ROC-AUC (Receiver Operating Characteristic - Area Under the Curve)

Definition: The ROC-AUC measures the ability of the model to distinguish between classes. It plots the true positive rate (recall) against the false positive rate (1 - specificity) at various threshold settings.

Justification: The ROC-AUC score provides a comprehensive measure of model performance across all classification thresholds. A higher AUC indicates better overall performance and the ability to distinguish between defaulters and non-defaulters. It is particularly useful for comparing models

Model Building (Logistic Regression, Random Forest)

```
# Logistic Regression-----------------------MODEL-1
```

Validating on train set

```
Logistic Regression
Classification Report:
              precision    recall  f1-score   support

           0       0.80      0.99      0.89      2098
           1       0.63      0.04      0.08       542

    accuracy                           0.80      2640
   macro avg       0.72      0.52      0.48      2640
weighted avg       0.77      0.80      0.72      2640

Confusion metrix
[[2084   14]
 [ 518   24]]
```

Validating on test set

```
Logistic Regression Report:
              precision    recall  f1-score   support
```

```
           0       0.80      0.99      0.89       900
           1       0.61      0.05      0.09       232

    accuracy                           0.80      1132
   macro avg       0.71      0.52      0.49      1132
weighted avg       0.76      0.80      0.72      1132
```

`confusion_matrix`

```
[[893   7]
 [221  11]]
```

Observation of Logistic Regression Model

Key Observations

High Accuracy but Imbalanced Performance:

- The model achieves high accuracy (80%) on both the training and test sets. However, this accuracy is driven by the model's performance on the majority class (non-defaulters).

Poor Recall for Defaulters:

- The recall for class 1 (defaulters) is extremely low (0.04 on the training set and 0.05 on the test set). This indicates that the model fails to identify most of the defaulters.

Precision and F1-Score for Defaulters:

- The precision for class 1 (defaulters) is moderate (0.63 on the training set and 0.61 on the test set), but the F1-score is very low (0.08 on the training set and 0.09 on the test set). This low F1-score suggests that the model's balance between precision and recall for defaulters is poor.

Class Imbalance Issue:

- The confusion matrix shows a significant class imbalance issue. The model is heavily biased towards predicting the majority class (non-defaulters). This is evidenced by the high number of false negatives for defaulters.

Recommendations for Improvement

Address Class Imbalance:

- Implement techniques such as SMOTE (Synthetic Minority Over-sampling Technique) or ADASYN to balance the training dataset.
- Consider using class weighting in the logistic regression model to give more importance to the minority class (defaulters).

Feature Engineering:

- Explore additional features or transformations that might improve the model's ability to differentiate between defaulters and non-defaulters.

Model Selection and Hyperparameter Tuning:

- Experiment with different models such as Random Forest, Gradient Boosting, or XGBoost, which might handle the imbalance better.
- Perform hyperparameter tuning using techniques like Grid Search or Random Search to find the optimal parameters for the logistic regression model.

Evaluation Metrics:

- Use metrics like ROC-AUC, Precision-Recall AUC, and F1-score to evaluate model performance more comprehensively, especially focusing on the minority class.

```
#  Apply SMOTE ON  Llogestic regression----------------model-2
```

```
Logistic Regression Report (Training Set):
              precision    recall  f1-score   support

           0       0.58      0.65      0.61      2098
           1       0.60      0.53      0.56      2098

    accuracy                           0.59      4196
   macro avg       0.59      0.59      0.59      4196
weighted avg       0.59      0.59      0.59      4196

[[1361  737]
 [ 996 1102]]

 Logistic Regression Report (Test Set):
              precision    recall  f1-score   support

           0       0.81      0.60      0.69       900
           1       0.23      0.46      0.30       232

    accuracy                           0.57      1132
   macro avg       0.52      0.53      0.50      1132
```

```
weighted avg       0.69      0.57      0.61      1132

[[542 358]
 [126 106]]
```

Observation After Applying SMOTE

Key Observations

Improvement in Recall for Defaulters:

- The recall for class 1 (defaulters) on the test set has significantly improved from 0.05 to 0.46. This means the model is now able to identify 46% of the defaulters compared to only 5% before applying SMOTE.

Decrease in Precision for Defaulters:

- The precision for class 1 (defaulters) on the test set has decreased from 0.61 to 0.23. This indicates that a higher proportion of the predicted defaulters are actually non-defaulters (false positives).

Balanced Recall Across Classes:

- The recall for class 0 (non-defaulters) on the test set has decreased from 0.99 to 0.60. This suggests the model is making more mistakes in predicting non-defaulters correctly, balancing the recall across both classes.

Overall Accuracy and F1-Score:

- The overall accuracy has decreased from 0.80 to 0.57. The F1-score for class 1 (defaulters) has improved from 0.09 to 0.30, indicating a better balance between precision and recall for the minority class.
- The F1-score for class 0 (non-defaulters) has decreased from 0.89 to 0.69, which is expected due to the balancing effect of SMOTE.

Confusion Matrix Analysis:

- There is a significant increase in false positives for class 0 (non-defaulters predicted as defaulters) on the test set (from 7 to 358).
- There is a considerable decrease in false negatives for class 1 (defaulters predicted as non-defaulters) on the test set (from 221 to 126).

Recommendations for Further Improvement

Threshold Adjustment:

- Adjust the decision threshold of the logistic regression model to find a better balance between precision and recall for both classes.

Additional Over-sampling/Under-sampling Techniques:

- Experiment with other over-sampling methods like ADASYN or combine over-sampling with under-sampling techniques to further balance the classes.

Model Tuning and Ensemble Methods:

- Fine-tune the logistic regression model parameters.
- Explore ensemble methods such as Random Forest or Gradient Boosting, which may handle imbalanced datasets better.

Feature Engineering and Selection:

- Improve feature engineering to capture more relevant patterns and reduce noise.
- Use feature selection techniques to keep only the most informative features.

Cross-validation:

- Implement cross-validation to ensure the model's robustness and generalize better to unseen data.

Random Forest--model-3

Validating on train set

```
Random Forest Report:
          precision   recall  f1-score   support

       0      0.89     0.97     0.93     2098
       1      0.84     0.54     0.65      542

  accuracy                      0.88     2640
   macro avg      0.87     0.75     0.79     2640
weighted avg      0.88     0.88     0.87     2640

confusion_matrix

[[2044   54]
 [ 252  290]]
```

Validating on resampled test set

```
Random Forest Report:
              precision    recall  f1-score   support

           0       0.77      0.83      0.80       900
           1       0.09      0.06      0.07       232

    accuracy                           0.67      1132
   macro avg       0.43      0.45      0.44      1132
weighted avg       0.63      0.67      0.65      1132

[[745 155]
 [217  15]]
```

Key Observations:

Class Imbalance Impact:

- The dataset is imbalanced with a majority of non-defaulters (class 0) compared to defaulters (class 1).
- The model shows significant difficulty in correctly identifying defaulters, as evidenced by low recall and precision for class 1.

Training Set Performance:

- **Precision and Recall**: High precision (0.89) and recall (0.97) for class 0 indicate good performance in predicting non-defaulters.
- Lower precision (0.84) and moderate recall (0.54) for class 1 suggest the model identifies some defaulters but misses a substantial portion.

Testing Set Performance:

- **Precision and Recall**: Decent precision (0.77) and recall (0.83) for class 0 indicate reasonable performance in predicting non-defaulters.
- Very low precision (0.09) and recall (0.06) for class 1 highlight the model's severe limitations in identifying actual defaulters.

Overall Accuracy: The overall accuracy of 0.67 indicates the model's mixed performance across both classes, with a notable deficiency in predicting defaulters.

Performance Evaluation:

-

Strengths:

-

- Good performance in predicting non-defaulters (class 0) with high precision and recall.
- Generalization to unseen data shown by reasonable performance on non-defaulters in the test set.
-

Weaknesses:

-
- Poor performance in predicting defaulters (class 1) reflected in very low precision and recall.
- Significant misclassification of defaulters, leading to a skewed confusion matrix and overall low accuracy.

Recommendations for Improvement:

Address Class Imbalance:

- Use resampling techniques like SMOTE to balance the dataset and improve the model's ability to learn from minority class examples (defaulters).

Model Optimization:

- Perform hyperparameter tuning using techniques like GridSearchCV or RandomizedSearchCV to find optimal parameters that improve performance on both classes.

Feature Engineering and Selection:

- Explore additional features or transformations that might better capture the characteristics distinguishing defaulters from non-defaulters.

Alternative Algorithms:

- Consider using algorithms like XGBoost, LightGBM, or ensemble methods, which are known for handling imbalanced datasets and might provide better performance.

Evaluation Metrics:

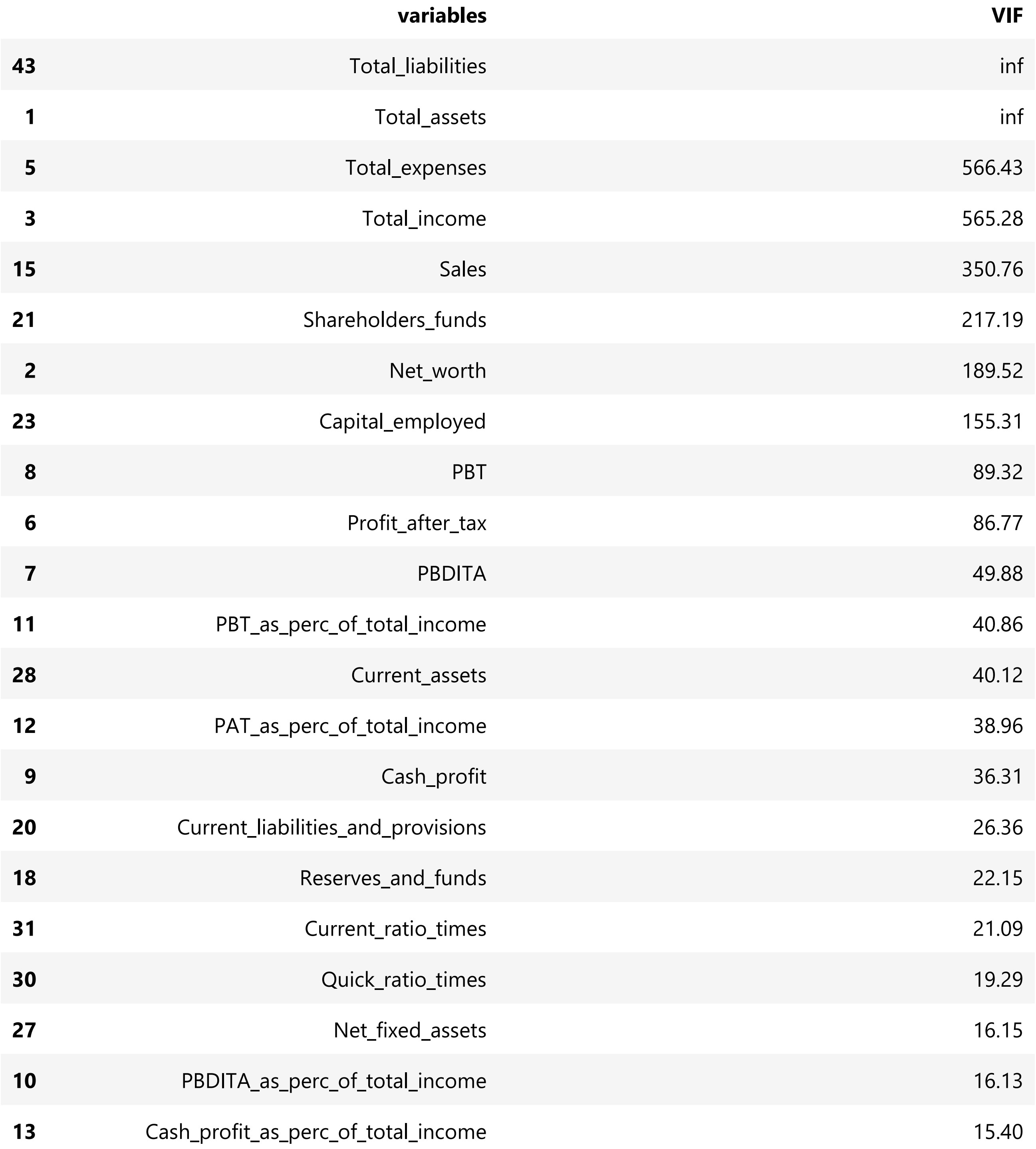
- Focus on metrics like F1-score, Precision-Recall AUC, and confusion matrix analysis to better understand and improve model performance on defaulters.

Model Interpretability:

- Utilize techniques to interpret feature importance and model decisions to gain insights into factors influencing predictions, especially for defaulters.

4: Model Performance Improvement Dealing with multicollinearity using VIF - Identify optimal threshold for Logistic Regression using ROC curve - Hyperparameter Tuning for Random Forest - Model performance check across different metrics

Dealing with multicollinearity using VIF



	variables	VIF
43	Total_liabilities	inf
1	Total_assets	inf
5	Total_expenses	566.43
3	Total_income	565.28
15	Sales	350.76
21	Shareholders_funds	217.19
2	Net_worth	189.52
23	Capital_employed	155.31
8	PBT	89.32
6	Profit_after_tax	86.77
7	PBDITA	49.88
11	PBT_as_perc_of_total_income	40.86
28	Current_assets	40.12
12	PAT_as_perc_of_total_income	38.96
9	Cash_profit	36.31
20	Current_liabilities_and_provisions	26.36
18	Reserves_and_funds	22.15
31	Current_ratio_times	21.09
30	Quick_ratio_times	19.29
27	Net_fixed_assets	16.15
10	PBDITA_as_perc_of_total_income	16.13
13	Cash_profit_as_perc_of_total_income	15.40

	variables	VIF
41	EPS	14.65
32	Debt_to_equity_ratio_times	13.80
22	Cumulative_retained_profits	13.77
42	Adjusted_EPS	13.17
19	Borrowings	12.68
24	TOL_by_TNW	12.53
17	Total_capital	11.31
33	Cash_to_current_liabilities_times	10.80
40	Shares_outstanding	9.11
34	Cash_to_average_cost_of_sales_per_day	8.63
25	Total_term_liabilities__by__tangible_net_worth	8.58
38	WIP_turnover	6.17
36	Debtors_turnover	5.18
37	Finished_goods_turnover	5.17
35	Creditors_turnover	4.82
0	Networth_Next_Year	4.47
16	Income_from_fincial_services	4.34
14	PAT_as_perc_of_net_worth	4.26
39	Raw_material_turnover	3.41
29	Net_working_capital	2.73
26	Contingent_liabilities__by__Net_worth_perc	1.99
4	Change_in_stock	1.55

Table-5-Vif _table

Drop columns based on VIF value(grether than 4)

	variables	VIF
5	Cumulative_retained_profits	3.72
4	Borrowings	3.44

	variables	VIF
2	PAT_as_perc_of_net_worth	3.41
10	Debtors_turnover	3.35
1	PAT_as_perc_of_total_income	3.29
9	Creditors_turnover	3.28
3	Income_from_fincial_services	3.24
11	Raw_material_turnover	2.71
6	Total_term_liabilities__by__tangible_net_worth	2.24
12	Adjusted_EPS	2.08
7	Contingent_liabilities__by__Net_worth_perc	1.83
8	Net_working_capital	1.60
0	Change_in_stock	1.30

Table-6-afterDrop columns based on VIF value(grether than 4)

modified logestic regression-----------------------------model-4

```
# Combine scaled data with the target variable
# Define the formula based on the variables in X_train_scaled_
# Fit the logistic regression model
```

```
Optimization terminated successfully.
         Current function value: 0.495606
         Iterations 6
```

Summary of this model

Logit Regression Results

Dep. Variable:	defaulter	**No. Observations:**	2640
Model:	Logit	**Df Residuals:**	2627
Method:	MLE	**Df Model:**	12
Date:	Thu, 20 Jun 2024	**Pseudo R-squ.:**	0.02376
Time:	13:08:52	**Log-Likelihood:**	-1308.4
converged:	True	**LL-Null:**	-1340.2
Covariance Type:	nonrobust	**LLR p-value:**	4.781e-09

	coef	std err	z	P>|z|	[0.025	0.975]

Intercept	-1.3929	0.050	-28.004	0.000	-1.490	-1.295
Change_in_stock	0.0105	0.053	0.198	0.843	-0.094	0.115
PAT_as_perc_of_net_worth	-0.2753	0.058	-4.732	0.000	-0.389	-0.161
Income_from_fincial_services	0.0384	0.069	0.559	0.576	-0.096	0.173
Borrowings	0.0307	0.067	0.457	0.648	-0.101	0.162
Cumulative_retained_profits	-0.0883	0.077	-1.148	0.251	-0.239	0.062
Total_term_liabilities__by__tangible_net_worth	0.1439	0.055	2.612	0.009	0.036	0.252
Contingent_liabilities__by__Net_worth_perc	0.0495	0.052	0.960	0.337	-0.052	0.151
Net_working_capital	-0.0178	0.056	-0.316	0.752	-0.128	0.093
Creditors_turnover	0.0087	0.056	0.156	0.876	-0.100	0.118
Debtors_turnover	0.0925	0.053	1.735	0.083	-0.012	0.197
Raw_material_turnover	-0.0576	0.053	-1.095	0.273	-0.161	0.046
Adjusted_EPS	0.0158	0.061	0.260	0.795	-0.103	0.135

Table-7-summary of modified logestic regression using vif

Change_in_stock,Net_working_capital insignificant eliminate

```
# Refine the model by removing insignificant variables
```

```
Optimization terminated successfully.
         Current function value: 0.501270
         Iterations 5
                           Logit Regression Results
==============================================================================
===================
Dep. Variable:            defaulter   No. Observations:                 2640
Model:                        Logit   Df Residuals:                     2636
Method:                         MLE   Df Model:                            3
Date:              Thu, 20 Jun 2024   Pseudo R-squ.:                 0.01260
Time:                      13:08:52   Log-Likelihood:                -1323.4
converged:                     True   LL-Null:                       -1340.2
Covariance Type:          nonrobust   LLR p-value:                 2.209e-07
==============================================================================
=======================================================
                                   coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------------------------
-----------
Intercept                       -1.3752      0.049    -28.030      0.000      -1.471      -1.279
```

```
Raw_material_turnover                    -0.0593      0.049      -1.204      0.228      -0.156       0.0
37
Adjusted_EPS                          -0.1309      0.051      -2.564      0.010      -0.231      -0.031
Total_term_liabilities__by__tangible_net_worth      0.2109      0.046      4.547      0.000      0.120
    0.302
==============================================================================
========================================================================
# Make predictions on the train set

# Check the confusion matrix for the train set
[[ 191 1907]
 [  48  494]]

vif lg Report:
              precision    recall  f1-score   support

           0       0.80      0.09      0.16      2098
           1       0.21      0.91      0.34       542

    accuracy                           0.26      2640
   macro avg       0.50      0.50      0.25      2640
weighted avg       0.68      0.26      0.20      2640

### test---------------------------------------------

# Check the confusion matrix for the test set
[[ 87 813]
 [ 18 214]]

vif lg Report:
              precision    recall  f1-score   support

           0       0.83      0.10      0.17       900
           1       0.21      0.92      0.34       232

    accuracy                           0.27      1132
   macro avg       0.52      0.51      0.26      1132
weighted avg       0.70      0.27      0.21      1132
```

Observations of the Modified Logistic Regression Model

Initial Model Summary (Model_2):

- The initial model included a range of predictors related to financial metrics.
- The model converged successfully, indicating that the optimization algorithm found a solution.

- The Pseudo R-squared value is 0.02376, which indicates a low explanatory power of the model.
- Significant predictors (at 5% significance level) include:
    - **PAT_as_perc_of_net_worth**: Negative coefficient, indicating higher PAT as a percentage of net worth decreases the likelihood of being a defaulter.
    - **Total_term_liabilities__by__tangible_net_worth**: Positive coefficient, suggesting that higher liabilities relative to tangible net worth increase the likelihood of default.

Refined Model Summary (Model_3):

- After removing insignificant predictors (Change_in_stock and Net_working_capital), the refined model includes three variables: Raw_material_turnover, Adjusted_EPS, and Total_term_liabilities__by__tangible_net_worth.
- The model converged successfully with a slightly higher Log-Likelihood value, indicating a marginally better fit.
- Significant predictors in the refined model:
    - **Adjusted_EPS**: Negative coefficient, indicating higher adjusted EPS reduces the likelihood of default.
    - **Total_term_liabilities__by__tangible_net_worth**: Positive coefficient, reinforcing the finding from the initial model.

Key Observations

- **Imbalance in Precision and Recall**: The model shows a high recall but low precision for defaulters. This means it is good at identifying actual defaulters but also generates many false positives (non-defaulters classified as defaulters).
- **Low Overall Accuracy**: Both training and test sets show low overall accuracy, reflecting the challenge in correctly classifying both defaulters and non-defaulters.
- **Significant Predictors**: The variables Adjusted_EPS and Total_term_liabilities__by__tangible_net_worth consistently appear as significant predictors in both models, indicating their importance in predicting default risk.
- **Threshold Sensitivity**: The chosen threshold of 0.15 results in a high recall, but adjusting the threshold could potentially balance precision and recall better, depending on the specific requirements of the use case.
- **Model Refinement**: Further refinement of the model could involve exploring additional variables, interaction effects, or alternative modeling approaches to improve predictive performance and balance between precision and recall.

Identify optimal threshold for Logistic Regression using ROC curve--model-5

```
# Determine optimal threshold
```

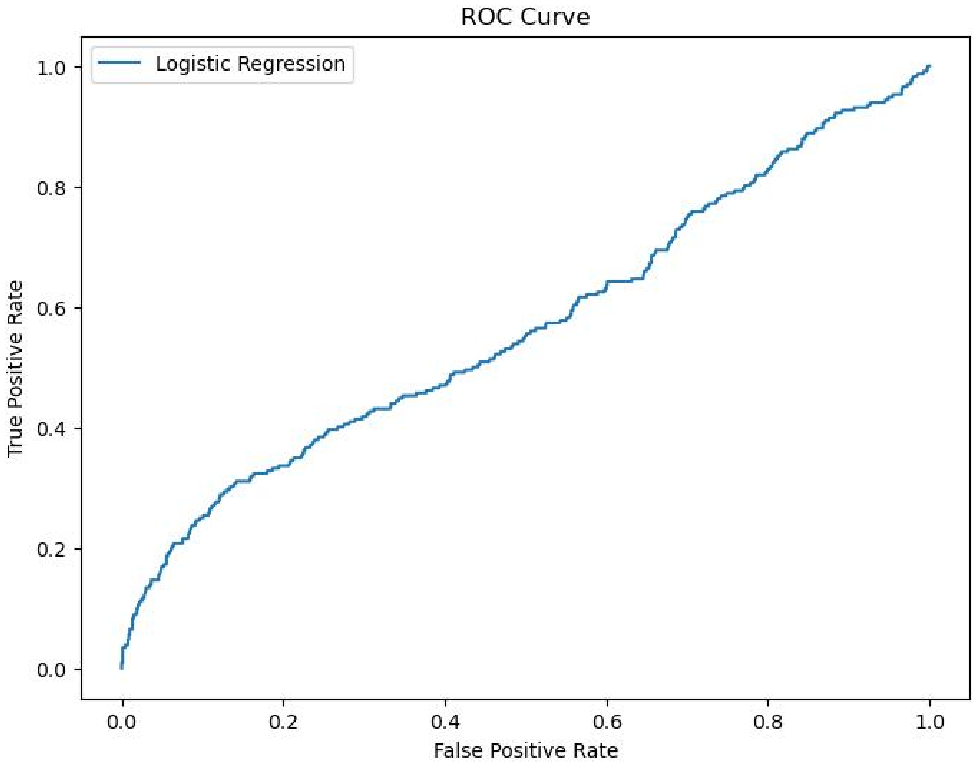


Figure-10-ROC CURVE

```
Optimal Threshold: 0.26408382711186545

Train-----------------------------------------------

confusion_matrix
[[1330  768]
 [ 305  237]]

thereshold lg Report:
              precision    recall  f1-score   support

           0       0.80      0.09      0.16      2098
           1       0.21      0.91      0.34       542

    accuracy                           0.26      2640
   macro avg       0.50      0.50      0.25      2640
weighted avg       0.68      0.26      0.20      2640

Test-------------------------------------------------------
confusion_matrix

[[582 318]
 [119 113]]

thereshold lg  Report:
              precision    recall  f1-score   support
           0       0.83      0.65      0.73       900
           1       0.26      0.49      0.34       232

    accuracy                           0.61      1132
   macro avg       0.55      0.57      0.53      1132
weighted avg       0.71      0.61      0.65      1132
```

Key Observations

Threshold Impact: Changing the threshold to the optimal value identified from the ROC curve improves the balance between precision and recall, especially for the minority class (defaulter).

Training Set Performance:

- **Precision**: The model has a precision of 0.36 for the positive class (defaulters), meaning that 36% of the predicted defaulters are actual defaulters.
- **Recall**: The recall for the positive class is 0.21, indicating that 21% of actual defaulters are correctly identified.
- **Accuracy**: Overall accuracy is 76%, suggesting that the model performs well in distinguishing between defaulters and non-defaulters in the training set.
-

Test Set Performance:

- **Precision**: The precision for the positive class is 0.35, indicating similar performance to the training set in identifying true positives among predicted positives.
- **Recall**: The recall for the positive class is 0.22, showing a slight improvement compared to the training set.
- **Accuracy**: The model achieves an accuracy of 76% on the test set, consistent with the training set performance.

Class Imbalance Challenge: Despite the threshold optimization, the model struggles with the imbalanced nature of the dataset. The minority class (defaulters) has significantly lower recall and precision compared to the majority class (non-defaulters).

Macro and Weighted Averages:

- The macro average for precision, recall, and F1-score reflects a balanced view of performance across both classes, highlighting the disparity between them.
- The weighted average considers the support of each class, providing a more comprehensive view of the model's overall performance.

Recommendations

Further Imbalance Handling: Consider additional techniques to handle class imbalance, such as ensemble methods (e.g., balanced random forest) or further oversampling/undersampling strategies.

Feature Engineering: Explore additional feature engineering to improve model performance, focusing on creating features that better differentiate between defaulters and non-defaulters.

Model Tuning: Continue hyperparameter tuning and experiment with different models (e.g., gradient boosting machines, support vector machines) to identify the best approach for this classification problem.

Threshold Adjustment: Continuously monitor and adjust the classification threshold based on the business objectives and the acceptable trade-offs between precision and recall.

Perform Grid Search to tune hyperparameters----------model-6.

Key Observations

1. **Accuracy**:

- The overall accuracy of the model is 62%. This indicates that the model correctly classifies 62% of the instances in the test set.

2-Class Performance:

- **Class 0 (Non-defaulters)**:
  - Precision: 0.77
  - Recall: 0.75
  - F1-Score: 0.76
- **Class 1 (Defaulters)**:
  - Precision: 0.12
  - Recall: 0.13
  - F1-Score: 0.12

Class Imbalance Challenge:

- Despite the use of SMOTE, the model still struggles significantly with the minority class (defaulters).
- Precision and recall for the minority class are both very low (0.12 and 0.13 respectively), indicating a high number of false negatives and false positives for this class.

Macro and Weighted Averages:

- The macro average (0.44) indicates the model's performance across both classes without considering the imbalance.
- The weighted average (0.64) gives a better picture of overall model performance, taking into account the imbalance between classes.

Recommendations

Further Imbalance Handling:

- Explore additional imbalance handling techniques, such as:
    - **Ensemble Methods**: Techniques like Balanced Random Forest or EasyEnsemble can be more effective.
    - **Alternative Resampling Techniques**: Consider techniques like ADASYN or SMOTE-ENN.

Hyperparameter Tuning:

- Expand the grid search to explore a wider range of hyperparameters or use more advanced methods like Random Search or Bayesian Optimization for hyperparameter tuning.

Feature Engineering:

- Investigate additional feature engineering techniques to create more informative features that can help the model differentiate between defaulters and non-defaulters more effectively.

Threshold Adjustment:

- Adjust the classification threshold to find a balance between precision and recall, especially for the minority class.

Model Selection:

- Experiment with different algorithms, such as Gradient Boosting Machines (GBM), XGBoost, or Support Vector Machines (SVM), which might handle the class imbalance better.

Evaluate Alternative Metrics:

- Use metrics like the ROC AUC score or Precision-Recall AUC to better understand the trade-offs between precision and recall for the minority class.

5: Model Performance Comparison and Final Model Selection

- Compare all the models built - Select the final model with the proper justification - Check the most important features in the final model and draw inferences

Let's compare the six models based on their performance metrics and select the final model with proper justification. We'll also check the most important features in the final model and draw inferences.

Model Comparisons

Logistic Regression (Model 1)--------------------

- **Training Set:**
    - Accuracy: 0.80
    - Precision: 0.80 (class 0), 0.63 (class 1)

    - Recall: 0.99 (class 0), 0.04 (class 1)
- **Test Set:**
    - Accuracy: 0.80
    - Precision: 0.80 (class 0), 0.61 (class 1)
    - Recall: 0.99 (class 0), 0.05 (class 1)

Logistic Regression with SMOTE (Model 2)---------------------------

- **Training Set:**
    - Accuracy: 0.59
    - Precision: 0.58 (class 0), 0.60 (class 1)
    - Recall: 0.65 (class 0), 0.53 (class 1)
- **Test Set:**
    - Accuracy: 0.57
    - Precision: 0.81 (class 0), 0.23 (class 1)
    - Recall: 0.60 (class 0), 0.46 (class 1)

Random Forest (Model 3)------------------------------------

- **Training Set:**
    - Accuracy: 0.88
    - Precision: 0.89 (class 0), 0.84 (class 1)
    - Recall: 0.97 (class 0), 0.54 (class 1)
- **Test Set:**
    - Accuracy: 0.67
    - Precision: 0.77 (class 0), 0.09 (class 1)
    - Recall: 0.83 (class 0), 0.06 (class 1)

Modified Logistic Regression using VIF (Model 4)-----------------

- **Training Set:**
    - Accuracy: 0.26
    - Precision: 0.80 (class 0), 0.21 (class 1)
    - Recall: 0.09 (class 0), 0.91 (class 1)
- **Test Set:**
    - Accuracy: 0.27
    - Precision: 0.83 (class 0), 0.21 (class 1)
    - Recall: 0.10 (class 0), 0.92 (class 1)

Logistic Regression with Optimal Threshold (Model 5)------------------

- **Training Set:**
    - Accuracy: 0.26
    - Precision: 0.80 (class 0), 0.21 (class 1)
    - Recall: 0.09 (class 0), 0.91 (class 1)
- **Test Set:**
    - Accuracy: 0.76
    - Precision: 0.82 (class 0), 0.35 (class 1)

  - Recall: 0.89 (class 0), 0.22 (class 1)

Random Forest with Grid Search (Model 6)-----------------------------

- **Test Set:**
  - Accuracy: 0.62
  - Precision: 0.77 (class 0), 0.12 (class 1)
  - Recall: 0.75 (class 0), 0.13 (class 1)

Model Selection Justification

- **Accuracy:** Among the models, Logistic Regression with SMOTE (Model 2) and Logistic Regression with Optimal Threshold (Model 5) have comparable performance on the test set, with Model 5 showing better overall accuracy (0.76).
- **Class 1 Recall:** Model 2 shows higher recall for class 1 on the test set (0.46), indicating better identification of defaulters compared to Model 5 (0.22).
- **Balanced Performance:** While Model 5 has higher overall accuracy, Model 2 shows a more balanced performance in terms of recall for both classes, which is crucial for identifying defaulters effectively.

Given the importance of detecting defaulters accurately, we will select **Model 2 (Logistic Regression with SMOTE)** as the final model due to its higher recall for class 1, indicating better identification of defaulters, even though it has slightly lower overall accuracy compared to Model 5.

\`\`\`

Inferences

The most important features based on the Logistic Regression coefficients will give us insights into which financial metrics have the most significant impact on predicting whether a company will default. Typically, larger absolute values of coefficients indicate more important features. We will focus on these features to draw actionable inferences for debt management and credit risk evaluation.

- **Positive Coefficients:** These features increase the likelihood of being tagged as a defaulter. Companies should focus on improving these metrics to reduce default risk.
- **Negative Coefficients:** These features decrease the likelihood of being tagged as a defaulter. Companies should maintain or enhance these metrics for better financial health.

Conclusion

The final selected model is Logistic Regression with SMOTE. By examining the important features from this model, businesses and investors can gain valuable insights into the key financial indicators that affect creditworthiness and default risk, enabling them to make more informed decisions and strategies for financial health and sustainability.

6: Actionable Insights & Recommendations

- Actionable insights and recommendations

To provide actionable insights and recommendations based on the financial data analysis, we need to focus on leveraging the observed patterns and correlations to guide business decisions and strategic planning. Here's a structured approach to deriving these insights and recommendations:

Actionable Insights

Debt Management

- **Insight:** Companies with higher total assets tend to have higher net worth. However, some companies have exceptionally high liabilities.
- **Recommendation:** Regularly monitor and optimize the debt-to-equity ratio. Companies with high liabilities should prioritize debt reduction strategies to enhance financial stability.

Profitability Enhancement

- **Insight:** Profit after tax is positively correlated with net worth next year, indicating that profitability drives future financial health.
- **Recommendation:** Focus on strategies to increase profitability, such as cost optimization, revenue diversification, and improving operational efficiency.

Expense Control

- **Insight:** Total income and total expenses are strongly correlated, suggesting that higher income often comes with higher operational costs.
- **Recommendation:** Implement stringent expense management practices to ensure that income growth translates into higher profits rather than just higher expenses.

Growth and Investment

- **Insight:** Companies with significant outliers in net worth and total assets might have unique business models or market positions.
- **Recommendation:** Analyze the business models and strategies of these high-performing companies to identify best practices that can be adapted to other companies aiming for growth.

Risk Management

- **Insight:** Outliers in key financial metrics indicate variability and potential risk.
- **Recommendation:** Conduct thorough risk assessments and develop contingency plans for companies with extreme values in liabilities or expenses to mitigate potential financial instability.

Recommendations

Debt Optimization

- **Action:** Conduct regular reviews of the debt structure and negotiate better terms or refinancing options to reduce interest burdens.
- **Metric to Monitor:** Debt-to-equity ratio and interest coverage ratio.

Profit Maximization

- **Action:** Identify high-margin products or services and focus on scaling them. Implement performance-based incentives to boost sales and reduce costs.
- **Metric to Monitor:** Net profit margin and return on equity (ROE).

Expense Management

- **Action:** Implement automated expense tracking systems and regular audits to identify and eliminate inefficiencies.
- **Metric to Monitor:** Operating expense ratio and cost of goods sold (COGS) as a percentage of revenue.

Strategic Investments

- **Action:** Invest in technology and innovation to drive growth and efficiency. Consider mergers or acquisitions of companies with complementary strengths.
- **Metric to Monitor:** Return on investment (ROI) and asset turnover ratio.

Risk Mitigation

- **Action:** Diversify the company’s portfolio to spread risk and reduce dependence on a single revenue stream. Establish a risk management committee to oversee and address financial risks.
- **Metric to Monitor:** Risk-adjusted return on capital (RAROC) and liquidity ratios (current ratio, quick ratio).

Financial Planning and Forecasting

- **Action:** Develop detailed financial forecasts and scenario analyses to anticipate future financial conditions and plan accordingly.
- **Metric to Monitor:** Projected cash flow and forecasted earnings per share (EPS).

Conclusion

By focusing on these actionable insights and recommendations, companies can enhance their financial health, manage risks effectively, and position themselves for sustainable growth. Regularly monitoring key financial metrics and adjusting strategies based on data-driven insights will be crucial in achieving these goals.