Finance and Risk Analytics
Bankruptcy prediction

By: Agnes Raja Kumari. E

PGP-Data Science and Business Analytics

PGPDSBA.O.MAY24.A

Contents

List of Figures 4

List of Tables 5

1. Introduction 6

1.1 Context 6

1.2 Objective 6

1.3 Problem definition 7

2. Data Background and Contents 7

2.1 Context and Variables 7

2.2 Types of Variables Included 7

2.3 Overview of the Dataset 8

2.3.1 Displaying first 5 rows of the dataset 8

2.3.2 Displaying last 5 rows of the dataset 8

2.3.3 Checking the shape of the dataset 8

2.3.4 Find out the data types of the columns 9

2.3.5 Statistical Summary of the Dataset 10

3. Exploratory Data Analysis 11

3.1 Univariate Analysis 11

3.1.1 Distribution of bankruptcy cases 11

3.1.2 Distribution and Outlier Detection of Numerical Variables 12

3.1.3 Distribution of Numerical Variables with Count Overlay 13

3.2 Bivariate Analysis 14

3.2.1 Boxplots for all the numerical Variable against target Variable 14

3.2.2 Calculate the correlation matrix 16

3.3 Data Preprocessing 17

3.3.1 Checking Outliers 17

3.3.2 Data Preparation for Modeling 18

3.3.3 Missing Values Detection and Treatment 19

3.3.4 Scaling the Data 20

4. Model Building 21

4.1 Metric of Choice 21

4.2 Model Selection and Justification 22

4.2.1 Logistic Regression 23

4.2.2 Random Forest 26

4.3 Model Performance Improvement.....28
4.3.1 Model Performance Improvement - Logistic Regression 29
4.3.2 Model Performance Improvement - Random Forest.....35
4.4 Model Comparison and Final Model Selection 38
4.4.1 Training Performance Comparison 38
4.4.2 Testing Performance Comparison.....39
4.4.3 Feature Importance of Tunes Random Forest Analysis.....41
5.Conclusions and Recommendations.....42

List of Figures

Figure 1 First 5 Rows of the dataset 8
Figure 2 Last 5 Rows of the dataset 8
Figure 3 Bar chart showing the distribution of bankruptcy cases 11
Figure 4 Distribution and Outlier Detection of Numerical Variables 12
Figure 5 Distribution of Numerical Variables with Count Overlay 13
Figure 6 Boxplot of numerical variables against 'Default' 14
Figure 7 Calculation of correlation matrix 16
Figure 8 First 5 Rows of Train Dataset 20
Figure 9 First 5 Rows of Test Dataset 20
Figure 10 Logistic Regression 23
Figure 11 Logistic Regression Model - Training Performance 24
Figure 12 Logistic Regression Model - Test Performance 25
Figure 13 Random Forest Model - Training Performance 26
Figure 14 Random Forest Model - Test Performance 27
Figure 15 Retraining Logistic Regression Model with new data 31
Figure 16 Receiver Operating Characteristic (ROC) 32
Figure 17 Logistic Regression Performance - Training Set 33
Figure 18 Logistic Regression Performance - Test Set 34
Figure 19 Model Performance Improvement - Random Forest 35
Figure 20 Random Forest Performance - Training Set 36
Figure 21 Random Forest Performance - Test Set 37
Figure 22 Training Performance Comparison 38
Figure 23 Testing Performance Comparison 39
Figure 24 Feature Importance of Tunes Random Forest Analysis 41

List of Tables

Table 1 Data types of the columns 9
Table 2 Statistical Summary of the dataset 10
Table 3 Checking Outliers 17
Table 4 Missing Values Detection on Training Dataset 19
Table 5 Missing Values Detection on Test Dataset 20
Table 6 Model performance classification of Train data 25
Table 7 Model performance classification of Test data 26
Table 8 Model performance classification of Train data 27
Table 9 Model performance classification of Test data 28
Table 10 Variance Inflation Factors 29
Table 11 List of High Variance Inflation Factors 30
Table 12 Model performance classification of Train data 33
Table 13 Model performance classification of Test data 34
Table 14 Model performance classification of Train data 36
Table 15 Model performance classification of Test data 37

1. Introduction

1.1 Context

Bankruptcy prediction is a crucial component of financial risk management that protects the interests of creditors, investors, and other stakeholders. Predicting a company's impending bankruptcy can help with timely interventions and smart decision-making, which can reduce losses and promote stability in the economy. Predictive modeling can benefit from the abundance of financial data provided by US corporations listed on major exchanges such as the New York Stock Exchange (NYSE) and NASDAQ, which are subject to regulatory scrutiny and strict financial reporting requirements. A firm is considered bankrupt, according to the Securities Exchange Commission (SEC), if it files for bankruptcy under the Bankruptcy Code's Chapter 11 (reorganization) or Chapter 7 (liquidation) provisions.

1.2 Objective

A well-known financial analytics company wants to create a Bankruptcy Prediction Tool to help regulators, investors, and financial institutions assess the bankruptcy risk of US publicly traded corporations. The program will evaluate past financial data using cutting-edge machine learning algorithms to find important signs and trends related to bankruptcy. The following are this tool's main goals:

1. Bankruptcy Risk Assessment: Provide a probabilistic estimate of a company's likelihood of filing for bankruptcy within a specified time frame (e.g., one year), allowing stakeholders to make informed decisions and take preventive measures.
2. Early Warning System: Develop an early warning system that flags companies exhibiting financial distress signals, enabling proactive risk management and strategic planning.
3. Financial Health Analysis: Analyze various financial metrics to offer a comprehensive assessment of a company's financial health, highlighting areas of concern and potential vulnerabilities.

As part of the data science team in the firm, I have been provided with a dataset containing financial metrics of various companies. The task is to analyze the data and develop a predictive model using machine learning techniques to identify whether a given company is at risk of bankruptcy in the near future. The model will help the organization anticipate potential financial distress and enable proactive measures to manage risks effectively.

1.3 Problem definition

The goal is to develop a **Bankruptcy Prediction Tool** that can accurately predict whether a publicly traded US company is at risk of bankruptcy in the near future. Using historical financial data provided by companies listed on stock exchanges such as the NYSE and NASDAQ, the system should leverage machine learning algorithms to:

- **Classify companies** as "Bankrupt" or "Not Bankrupt" based on their financial indicators.
- **Quantify bankruptcy risk** through probabilistic scoring.
- **Detect early warning signs** of financial distress to enable proactive risk mitigation.

The project involves end-to-end data analysis, feature selection, model training, evaluation, and validation to ensure the tool provides reliable and actionable insights for investors, financial institutions, and regulators.

2. Data Background and Contents

2.1 Context and Variables

This dataset pertains to the **financial performance analysis of companies**, primarily intended for **credit risk assessment**, **bankruptcy prediction**, or **financial health evaluation**. The available variables encompass a range of financial metrics that collectively provide a comprehensive view of a company's operations and stability.

2.2 Types of Variables Included

The dataset contains a mix of financial indicators drawn from various financial statements:

Balance Sheet Metrics

- Current_assets, Total_assets, Inventory, Total_liabilities, Total_current_liabilities, Retained_earnings, Total_long_term_debt, etc.
- These provide insight into the company's liquidity, solvency, and capital structure.

Income Statement Metrics

- Net_sales, Gross_profit, Net_income, EBITDA, EBIT, Cost_of_goods_sold, Total_operating_expenses
- These help assess profitability and operational efficiency.

Target Variable

- **Bankrupt**
- A binary indicator:
- 1 → The company filed for **bankruptcy**
- 0 → The company did **not** file for bankruptcy
- This makes the task a **binary classification problem**, where the goal is to predict whether a company is likely to go bankrupt based on historical financial data.

Data Dictionary

The data consists of financial metrics from the balance sheets of different companies. The detailed data dictionary is available in the data dictionary file ('Bankruptcy_Data.csv').

File Name: **'Bankruptcy_Data.csv`**

2.3 Overview of the Dataset

The initial steps to get an overview of the dataset include:

1. Observing the first few rows to verify that the dataset has been loaded properly.
2. Checking the number of rows and columns to understand the dataset's size.
3. Identifying data types of each column to ensure correct data storage and format.
4. Reviewing the statistical summary to get insights into the numerical features.

2.3.1 Displaying first 5 rows of the dataset

	Company_id	Current_assets	Cost_of_goods_sold	Depreciation_and_amortization	EBITDA	Inventory	Net_income	Total_receivables	Market_value	Net_sales	Total_assets	Total_lon
0	C_3	6.494	15.700	0.761	0.381	3.488	-1.738	2.677	2.9667	24.051	8.635	
1	C_6	6540.000	19419.000	1404.000	-456.000	822.000	-1762.000	1414.000	3445.0155	18963.000	32841.000	
2	C_8	189.471	125.059	11.600	44.563	48.423	21.354	49.409	626.5350	232.808	310.252	
3	C_12	34.617	104.017	3.165	5.766	13.184	0.812	11.964	15.1588	117.326	90.941	
4	C_14	90.558	148.628	0.909	7.759	37.818	4.245	41.500	65.1701	178.154	105.173	

Figure 1 First 5 Rows of the dataset

2.3.2 Displaying last 5 rows of the dataset

	Company_id	Current_assets	Cost_of_goods_sold	Depreciation_and_amortization	EBITDA	Inventory	Net_income	Total_receivables	Market_value	Net_sales	Total_assets
1978	C_7172	0.042	0.062	0.000	-0.048	0.000	-0.047	0.000	1.0813	0.062	0.046
1979	C_7176	6.385	11.054	0.201	0.421	0.000	0.175	3.678	23.9363	13.788	6.708
1980	C_7180	1135.848	1370.382	71.231	187.991	982.000	11.442	93.704	1626.4270	2517.639	1950.615
1981	C_7186	9.453	7.894	0.467	-1.672	0.880	-3.047	1.049	15.6110	6.222	16.247
1982	C_7288	2.753	5.436	0.679	-0.610	1.419	-5.163	1.073	5.9271	8.627	4.072

Figure 2 Last 5 Rows of the dataset

2.3.3 Checking the shape of the dataset

- The number of rows (observations) is 1983
- The number of columns (variables) is 20

2.3.4 Find out the data types of the columns

S.No	Column Name	Data Type
1	Company_id	object
2	Current_assets	float64
3	Cost_of_goods_sold	float64
4	Depreciation_and_amortization	float64
5	EBITDA	float64
6	Inventory	float64
7	Net_income	float64
8	Total_receivables	float64
9	Market_value	float64
10	Net_sales	float64
11	Total_assets	float64
12	Total_long_term_debt	float64
13	EBIT	float64
14	Gross_profit	float64
15	Total_current_liabilities	float64
16	Retained_earnings	float64
17	Total_revenue	float64
18	Total_liabilities	float64
19	Total_operating_expenses	float64
20	Bankrupt	int64

Table 1 Data types of the columns

Key Observations:

- The dataset is numerically rich and well-suited for statistical and machine learning models.
- Since all predictor variables (except ID) are numeric, this reduces the need for preprocessing like encoding.
- Scaling, feature selection, and multicollinearity checks (e.g., VIF) are appropriate preprocessing steps.
- The target is binary, making classification algorithms such as Logistic Regression, Random Forest suitable for predicting bankruptcy.

2.3.5 Statistical Summary of the Dataset

Variable	Count	Mean	Std Dev	Min	25%	50%	75%	Max
Current_assets	1983	485.49	1790.17	0.002	14.19	63.28	234.69	36105
Cost_of_goods_sold	1983	1024.31	4316.52	0	14.87	72.38	338.1	76809
Depreciation_and_amortization	1983	86.55	400.23	0	1.49	6.44	30.21	9338
EBITDA	1983	162.43	880.11	-5743	-2.52	5.31	55.98	18632
Inventory	1983	119.28	506.21	0	0	5.7	44.21	8923
Net_income	1983	-53	1536.72	-56122	-14.92	-0.8	9.55	8560
Total_receivables	1983	171.7	882.46	0	2.44	13.83	61.84	28813
Market_value	1983	1630.98	8159.3	0.0384	18.23	96.38	588.22	180090.4
Net_sales	1983	1426.31	5627.38	0	23.99	115	535.47	97863
Total_assets	1983	1641.77	6932.95	0.004	29.45	131.43	574.62	165282
Total_long_term_debt	1983	422.77	1902.19	0	0.04	5.61	125.98	45247
EBIT	1983	75.88	697.46	-10537	-7.78	0.98	28.23	16295
Gross_profit	1983	401.99	1763.76	-1317	6.18	35.99	174.95	41000
Total_current_liabilities	1983	364.71	1513.95	0.004	7.61	27.44	114.96	25427
Retained_earnings	1983	133.58	2235.73	-57158	-3.87	6.04	64.18	33896
Total_revenue	1983	1426.31	5627.38	0	23.99	115	535.47	97863
Total_liabilities	1983	1030.6	4515.32	0.004	11.03	49.13	315.27	110042
Total_operating_expenses	1983	1263.88	4970.26	0.067	31.26	111.38	488.99	81832
Bankrupt (Target)	1983	0.2088	0.4065	0	0	0	0	1

Table 2 Statistical Summary of the dataset

Key Observations:

1. Target Imbalance (Bankrupt Column):

- Only ~20.9% of companies in the dataset are bankrupt.
- This confirms class imbalance, which should be addressed using:
- Class weights
- SMOTE/undersampling
- Proper evaluation metrics (Recall, F1)

2. Wide Range and Skewness:

- Most features have large gaps between min and max, suggesting high skewness and outliers.
- Examples:
- Net_income: Min = -56121.9 vs. Max = 8560
- Market_value: Max = 180,090 → High variance
- Log transformation or standardization may be helpful.

3. Negative Means:

- Net_income and Retained_earnings have negative means, which may indicate:
- Financial stress is common in the sample.
- Bankruptcy correlates with these poor financial indicators.

4. Zero Values:

- Some features like Inventory, Depreciation, and Total_long_term_debt have minimum values of 0, possibly due to:
- Missing data recorded as 0
- Some companies truly having zero in these accounts.

3. Exploratory Data Analysis

3.1 Univariate Analysis

3.1.1 Distribution of bankruptcy cases

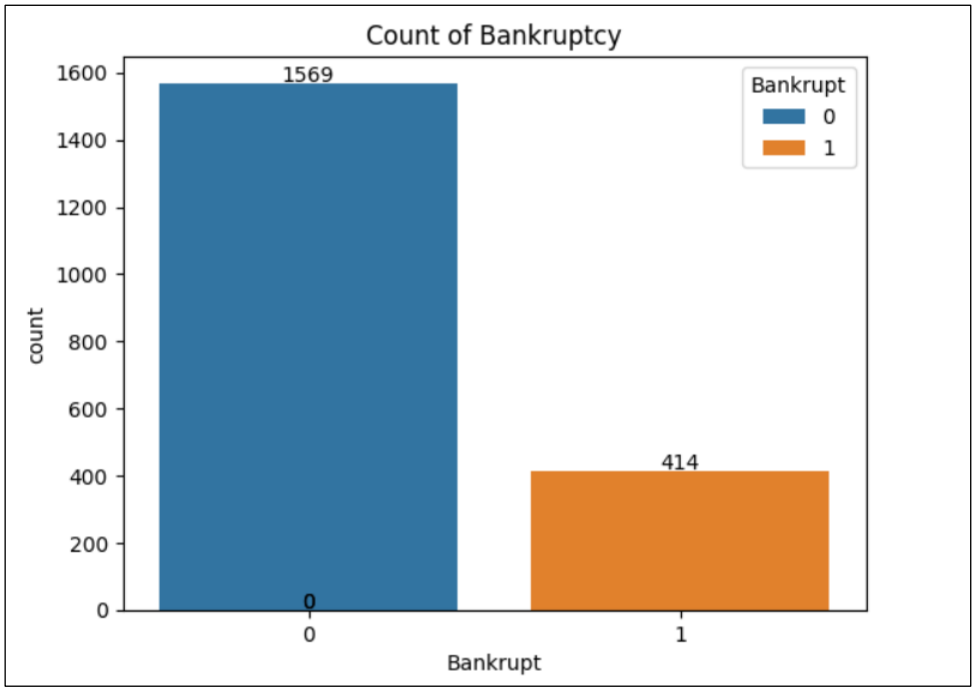

Figure 3 Bar chart showing the distribution of bankruptcy cases

Key Observations:

Class Imbalance:

- There are **1,569** non-bankrupt instances (Bankrupt = 0).
- There are **414** bankrupt instances (Bankrupt = 1).
- This indicates a **class imbalance**, with the non-bankrupt cases being significantly more frequent than bankrupt ones (approximately a 3.8:1 ratio).

Implications:

- The imbalance may bias predictive models towards the majority class (non-bankrupt).
- Techniques like **resampling (e.g., SMOTE, undersampling)** or **using class weights** should be considered during model building to handle this imbalance effectively.

3.1.2 Distribution and Outlier Detection of Numerical Variables

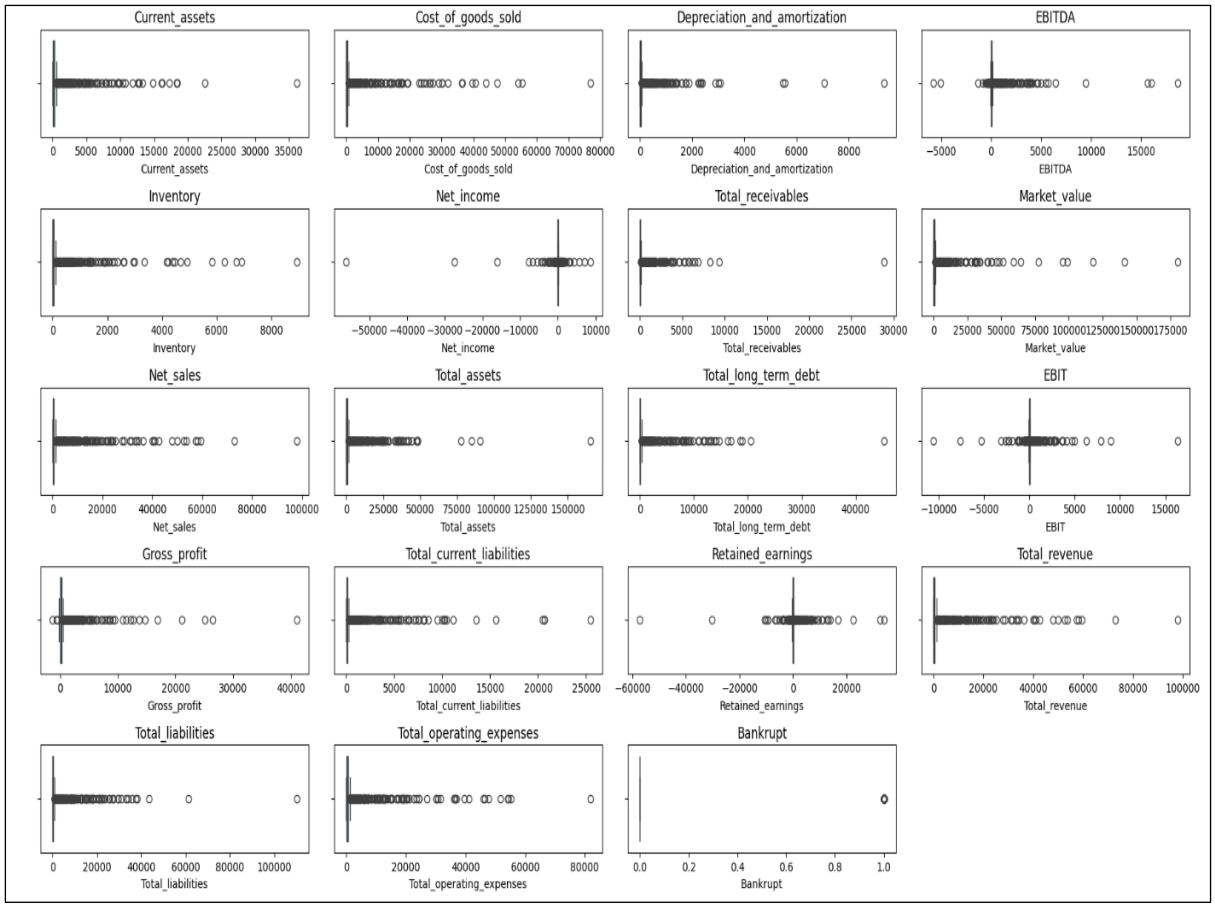

Figure 4 Distribution and Outlier Detection of Numerical Variables

Key Observations:

1. **Presence of Outliers:**
   - Most variables exhibit extreme outliers, indicating high variability and potential skewness in financial data.
   - For example: Market_value, Net_sales, Total_assets, and Total_revenue have long right tails with many extreme values.
2. **Skewed Distributions:**
   - Many variables are positively skewed, with the majority of data points clustered near the lower end and a few high values stretching the scale.
3. **Bankruptcy Indicator (Bankrupt):**
   - The Bankrupt variable is binary (0 or 1) and shows a sharp class separation, supporting the class imbalance noted earlier.
4. **Scaling Issue:**
   - The vast range differences among variables suggest the need for feature scaling (e.g., StandardScaler or MinMaxScaler) before model building.

3.1.3 Distribution of Numerical Variables with Count Overlay

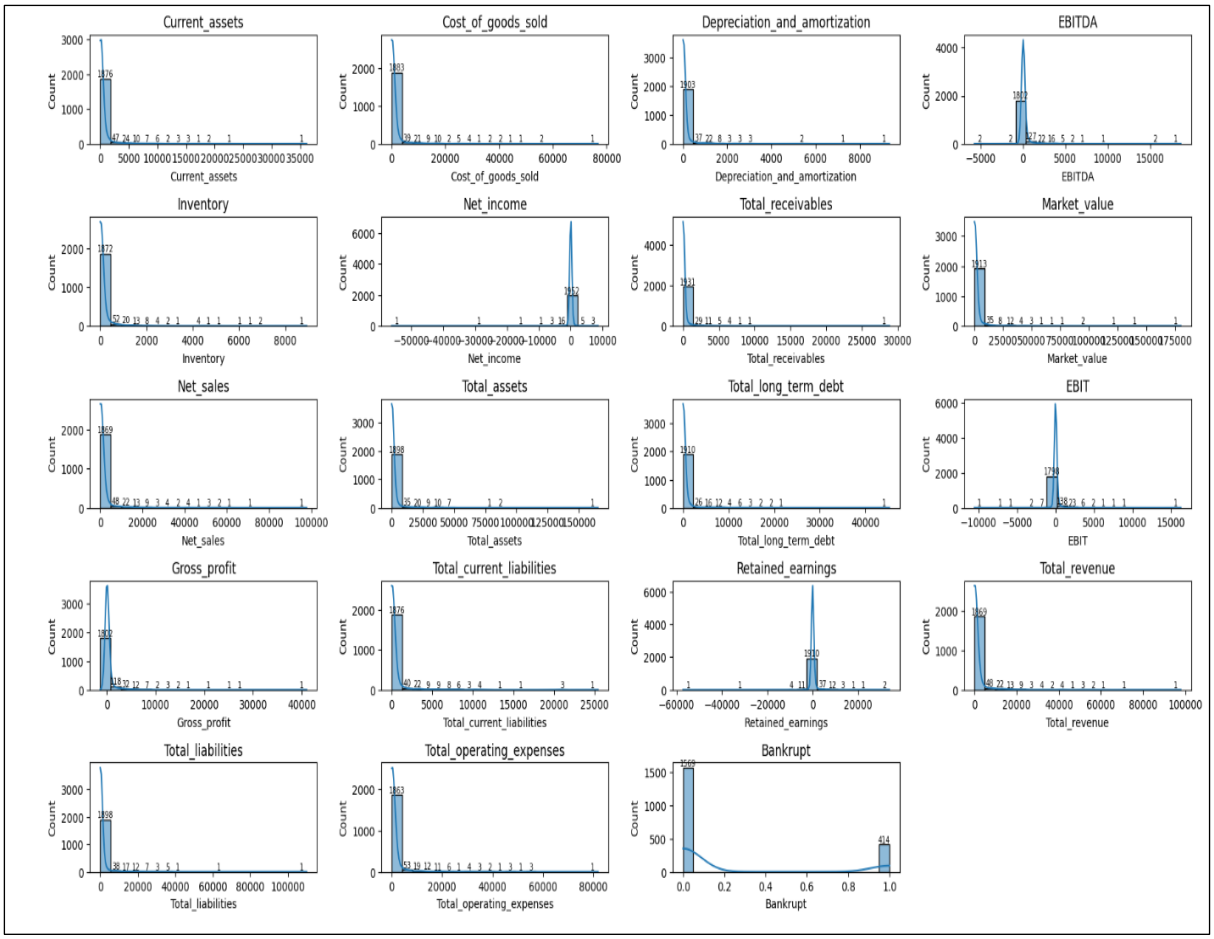

Figure 5 Distribution of Numerical Variables with Count Overlay

Key observations:

Skewed Distributions:

- Most financial variables (e.g., Current_assets, Total_assets, Total_liabilities, Net_sales, Total_revenue) exhibit right-skewed distributions, indicating the presence of a few companies with significantly higher values than the majority.

Presence of Outliers:

- Variables such as Market_value, Cost_of_goods_sold, Net_income, and Total_long_term_debt show long tails, suggesting extreme outlier values.
- These outliers may impact modeling and should be considered for transformation or treatment.

Binary Variable - Bankrupt:

- The Bankrupt variable is bimodal, representing a binary class (0: Not Bankrupt, 1: Bankrupt).
- There is a significant class imbalance, with more companies not going bankrupt.

3.2 Bivariate Analysis

3.2.1 Boxplots for all the numerical Variable against target Variable

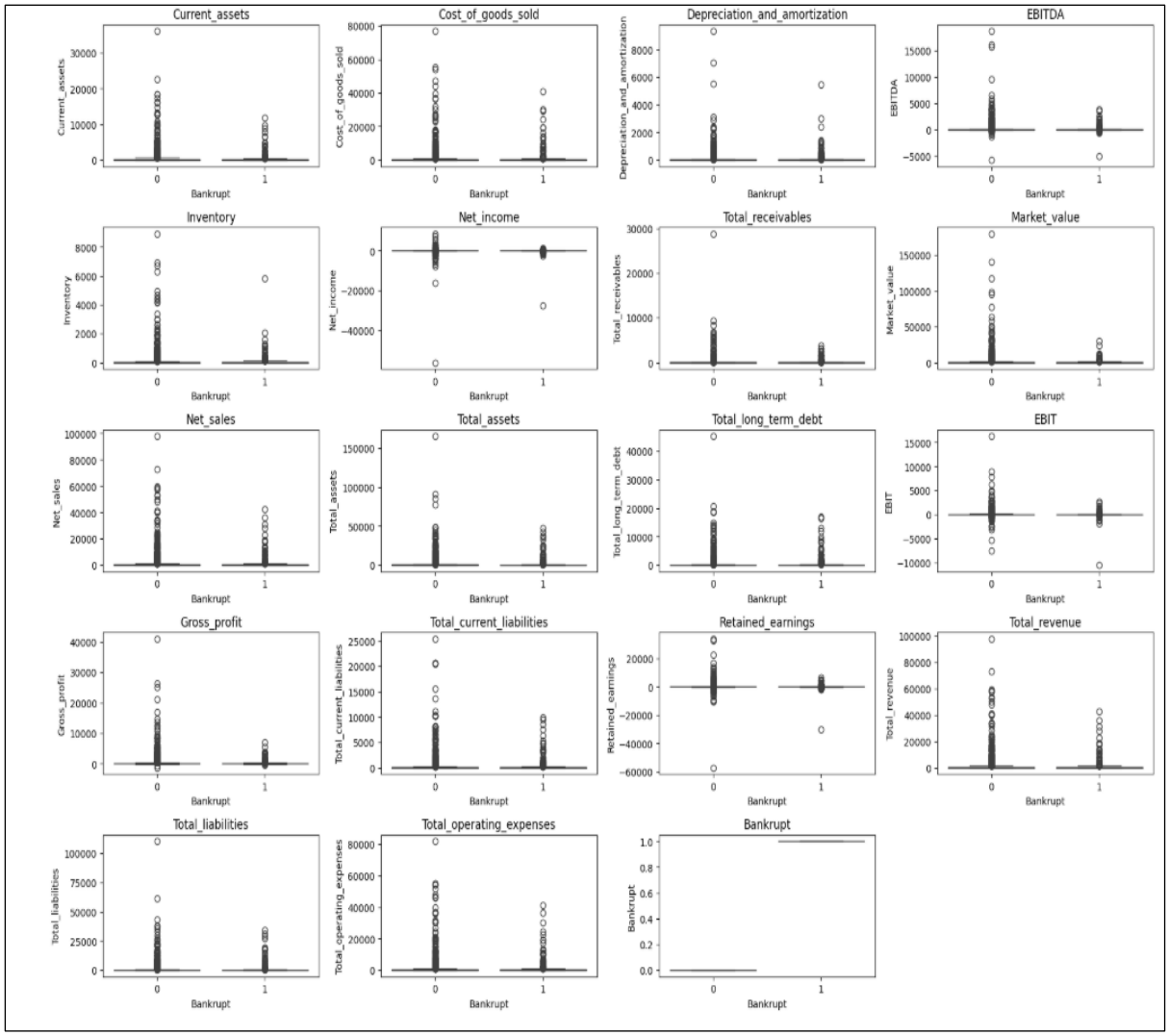

Figure 6 Boxplot of numerical variables against 'Default'

Key Observations:

Distribution Spread:

- Most financial variables have **wide spreads and many outliers**, especially for companies that did **not go bankrupt** (Bankrupt = 0).
- This suggests a **high variance** in financial health among non-bankrupt companies.

Central Tendency Comparison:

- For variables like Net_income, EBIT, EBITDA, and Gross_profit, bankrupt companies (Bankrupt = 1) tend to have **lower medians**, often near or below zero.
- This implies that **lower profitability is associated with higher bankruptcy risk**.

Retained Earnings:

- Companies that went bankrupt generally show **more negative retained earnings**, indicating **accumulated losses over time**.

Total Liabilities and Long-Term Debt:

- Higher values are observed for both bankrupt and non-bankrupt firms, but **bankrupt companies** tend to have **relatively higher long-term debt burdens**, showing the strain of debt over time.

Outliers:

- Significant number of **extreme outliers** in most financial metrics for both groups.
- Suggests the need for **outlier handling** during preprocessing for modeling.

Bankrupt Class (0 vs 1):

- Despite overlap in distributions, **subtle differences in medians and spread** suggest that **financial features do influence bankruptcy**, though likely not linearly separable.

3.2.2 Calculate the correlation matrix

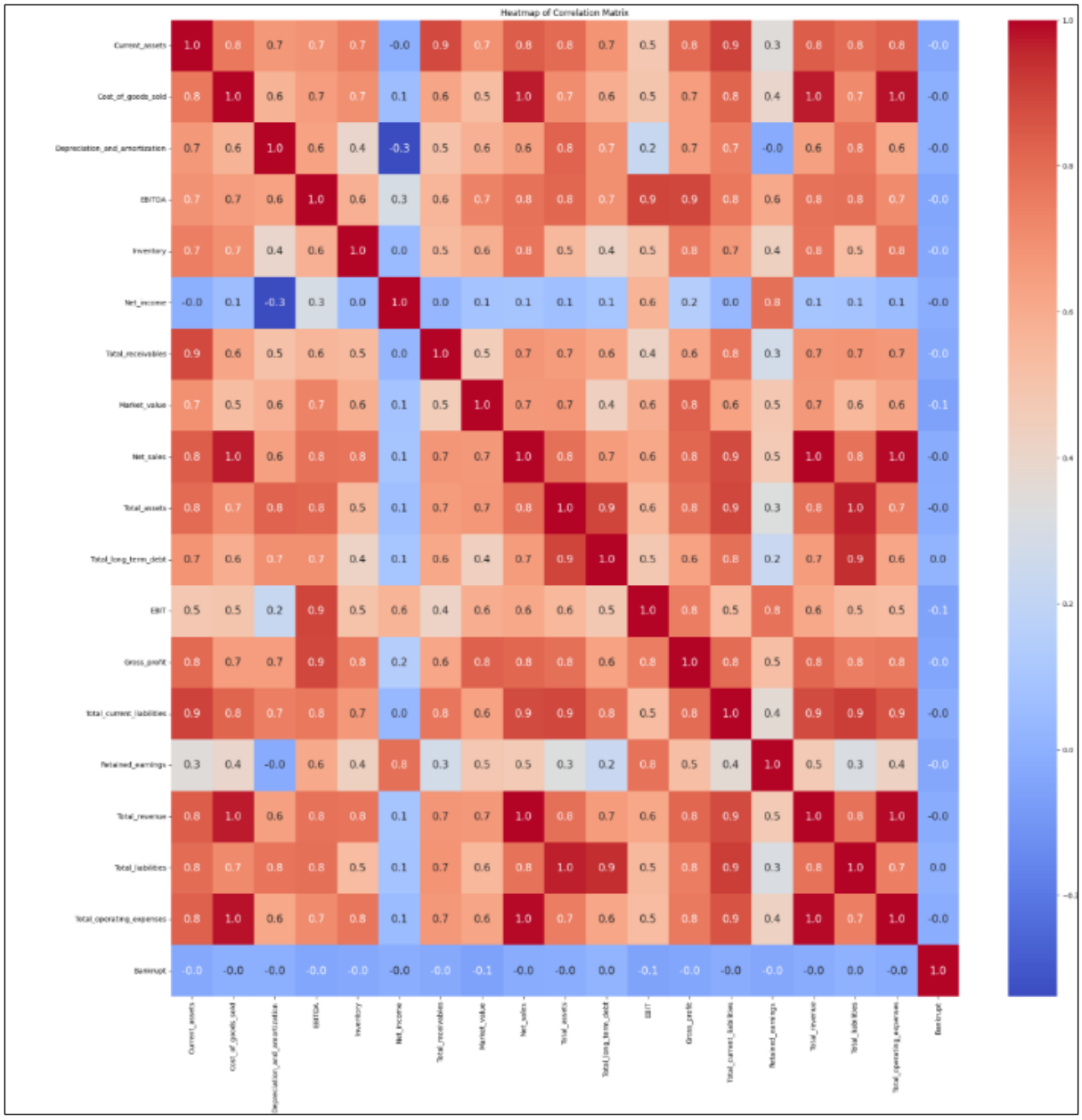

Figure 7 Calculation of correlation matrix

Key Observations:

1. **Bankruptcy Correlation**:
    - All features show **very weak correlation with Bankrupt**, values near **0.0 to -0.1**.
    - This confirms that **no single variable linearly explains bankruptcy**, indicating the **need for non-linear models** (e.g., Random Forest, Gradient Boosting).

2. **Highly Correlated Features** (Potential Multicollinearity):
   Several variables have **strong positive correlations** with each other, suggesting redundancy:
   - Net_sales vs Total_revenue: **1.0** – almost perfectly correlated.
   - Gross_profit vs EBITDA: **0.9**
   - Total_assets vs Total_liabilities: **0.9**
   - Cost_of_goods_sold vs Total_operating_expenses: **1.0**
   - Total_assets, Net_sales, Total_revenue, Total_liabilities, and Total_current_liabilities are **clustered** and strongly interrelated.
3. **Weak/Negative Correlations**:
   - Net_income has **weak correlation** with most other features (often below 0.2).
   - Net_income vs Depreciation_and_amortization: **-0.3**
   - Retained_earnings shows **moderate correlation** with profitability indicators like EBIT, EBITDA, Gross_profit.

3.3 Data Preprocessing

3.3.1 Checking Outliers

S.No	Column	Number of Outliers
1	Current_assets	279
2	Cost_of_goods_sold	311
3	Depreciation_and_amortization	302
4	EBITDA	335
5	Inventory	298
6	Net_income	485
7	Total_receivables	307
8	Market_value	278
9	Net_sales	296
10	Total_assets	310
11	Total_long_term_debt	332
12	EBIT	413
13	Gross_profit	280
14	Total_current_liabilities	314
15	Retained_earnings	408
16	Total_revenue	296
17	Total_liabilities	312
18	Total_operating_expenses	296
19	Bankrupt	414

Table 3 Checking Outluers

Key Observations:

1. High Outlier Count in Key Financial Metrics

- Variables such as Net_income (485), EBIT (413), and Retained_earnings (408) have a high number of outliers.
- These are critical features related to a company's profitability and performance this suggests a wide range of financial stability across companies.

2. Bankruptcy Column (414 Outliers)

- Since this is a binary variable (0 or 1), this outlier count likely reflects its imbalance (i.e., only ~21% are bankrupt) — not actual statistical outliers.

3. Implications for Modeling

- Outliers can bias models, especially distance-based ones (e.g., KNN, clustering) and linear models.
- **Need to handle outliers by:**
    - Using robust scalers (e.g., IQR, RobustScaler),
    - Applying log/sqrt transformations, or
    - Capping/extreme value treatment (Winsorization).

4. Common Across Revenue & Cost Columns

- Total_revenue, Net_sales, Cost_of_goods_sold, and Total_operating_expenses all have 290–310 outliers.
- Indicates high financial disparity among companies, possibly due to size, industry, or life cycle stage.

5. Model Robustness

- Tree-based models like Random Forest and XGBoost are generally more robust to outliers, making them preferable without deep outlier treatment.

3.3.2 Data Preparation for Modeling

- Seperating target variable from the rest of the data
- Splitting the data into train (75%) and test (25%) sets

3.3.3 Missing Values Detection and Treatment

Missing Values Detection on Training Dataset

S.No	Column	Missing Values
1	Current_assets	0
2	Cost_of_goods_sold	0
3	Depreciation_and_amortization	0
4	EBITDA	0
5	Inventory	0
6	Net_income	0
7	Total_receivables	0
8	Market_value	0
9	Net_sales	0
10	Total_assets	0
11	Total_long_term_debt	0
12	EBIT	0
13	Gross_profit	0
14	Total_current_liabilities	0
15	Retained_earnings	0
16	Total_revenue	0
17	Total_liabilities	0
18	Total_operating_expenses	0

Table 4 Missing Values Detection on Training Dataset

Key Observations:

- **No missing values** are present in the training set across all features.
- This indicates the dataset is **clean and ready for modeling** without any imputation required.

Missing Values Detection on Test Dataset

S.No	Column	Missing Values
1	Current_assets	0
2	Cost_of_goods_sold	0
3	Depreciation_and_amortization	0
4	EBITDA	0
5	Inventory	0
6	Net_income	0
7	Total_receivables	0
8	Market_value	0
9	Net_sales	0
10	Total_assets	0
11	Total_long_term_debt	0
12	EBIT	0
13	Gross_profit	0
14	Total_current_liabilities	0
15	Retained_earnings	0
16	Total_revenue	0
17	Total_liabilities	0
18	Total_operating_expenses	0

Table 5 Missing Values Detection on Test Dataset

Key Observations:

- **All 18 features** have **zero missing values**.
- This indicates a **clean dataset** ideal for statistical modelling and machine learning.
- We can proceed without any imputation or missing data treatment.

3.3.4 Scaling the Data

Scaling of features is done to bring all the features to the same scale.

First 5 Rows of Train Dataset

	Current_assets	Cost_of_goods_sold	Depreciation_and_amortization	EBITDA	Inventory	Net_income	Total_receivables	Market_value	Net_sales	Total_assets	Total_long_term_debt
0	-0.159768	-0.142125	-0.234780	-0.063513	-0.113882	0.099548	-0.072098	-0.091104	-0.132506	-0.167604	-0.217799
1	-0.274514	-0.236003	-0.261220	-0.199194	-0.237175	0.000488	-0.191910	-0.198944	-0.253588	-0.267951	-0.263127
2	-0.175941	-0.160988	-0.136617	-0.075109	-0.078514	0.054720	-0.122962	-0.138230	-0.155378	-0.172370	-0.173698
3	-0.274668	-0.236195	-0.261416	-0.198389	-0.237910	0.001816	-0.192215	-0.199350	-0.253852	-0.267943	-0.263127
4	-0.272938	-0.235566	-0.252166	-0.220011	-0.234932	-0.043588	-0.192093	-0.190211	-0.253589	-0.264987	-0.257469

Figure 8 First 5 Rows of Train Dataset

First 5 Rows of Test Dataset

	Current_assets	Cost_of_goods_sold	Depreciation_and_amortization	EBITDA	Inventory	Net_income	Total_receivables	Market_value	Net_sales	Total_assets	Total_long_term_debt
0	0.961504	0.178218	1.739125	0.534310	0.435299	-0.094970	0.491741	0.398906	0.381419	0.604897	0.140906
1	-0.263573	-0.229975	-0.189535	-0.198939	-0.237910	-0.093674	-0.177780	-0.198400	-0.242869	-0.262008	-0.262274
2	-0.272713	-0.234976	-0.258923	-0.199232	-0.237679	-0.003717	-0.190784	-0.198345	-0.252523	-0.266531	-0.261575
3	-0.265103	-0.230545	-0.251250	-0.177672	-0.225244	0.016272	-0.182056	-0.185388	-0.243520	-0.255296	-0.253015
4	0.759964	0.384240	0.441011	0.824896	1.277753	0.613444	0.699953	0.449776	0.614304	0.551349	0.307429

Figure 9 First 5 Rows of Test Dataset

4. Model Building

4.1 Metric of Choice

To assess the performance of the predictive model for identifying defaulter companies based on financial metrics, selecting the appropriate evaluation metrics is crucial. Since the problem involves classification, where the objective is to predict whether a company will default (binary classification: defaulter vs. non-defaulter), the following evaluation metrics are considered.

Key Metrics and Justification:

1. Accuracy

- **Definition:** Accuracy measures the proportion of correctly classified instances (both defaulters and non-defaulters) to the total number of instances.
- **Justification:** Accuracy alone is insufficient in cases of class imbalance, where non-defaulters may significantly outnumber defaulters. Additional metrics are required for a balanced evaluation.

2.Precision & Recall

- Precision (Positive Predictive Value): Measures the proportion of correctly predicted defaulters to the total instances predicted as defaulters.
- Formula: Precision = TP / (TP + FP)
- Recall (Sensitivity or True Positive Rate): Measures the proportion of actual defaulters correctly identified.
- Formula: Recall = TP / (TP + FN)
- Justification: These metrics are essential for evaluating false positives (incorrectly tagging a company as a defaulter) and false negatives (failing to identify an actual defaulter), both of which have significant financial implications.

3.F1-Score

- Definition: The harmonic mean of precision and recall, providing a balance between the two metrics.
- Formula: F1 = 2 * (Precision * Recall) / (Precision + Recall)
- Justification: Since both false positives and false negatives impact financial decision-making, the F1-score ensures a balance between precision and recall.

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

- Definition: Evaluates the model's ability to distinguish between defaulters and non-defaulters by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold values.
- Justification: A higher AUC value indicates better model performance in classification.

5.Confusion Matrix

- Definition: Provides a summary of classification results, displaying the number of true positives, true negatives, false positives, and false negatives.
- Justification: It helps in understanding the distribution of predictions and assessing model performance beyond a single metric.

Conclusion

Given the potential imbalance in the dataset, a combination of F1-score, ROC-AUC will be the primary evaluation metrics, ensuring robustness in detecting defaulters while minimizing false positives and false negatives. Accuracy, precision, recall, confusion matrix, and Log Loss will supplement the analysis for a holistic model assessment.

4.2 Model Selection and Justification

1.Logistic Regression

Why?

- A simple and interpretable model
- Suitable for problems with linear relationships
- Outputs probability scores, making threshold adjustments easier

2.Random Forest

Why?

- Captures complex non-linear relationships
- Handles missing values well
- Reduces overfitting by averaging multiple decision trees
- Model Training Steps
- Train Logistic Regression and Random Forest models using training data.
- Tune hyperparameters (e.g., regularization for Logistic Regression, number of trees for Random Forest).
- Predict outcomes on the test dataset.

4.2.1 Logistic Regression

```
         Current function value: 0.497065
         Iterations: 35
         Function evaluations: 37
         Gradient evaluations: 37
                         Logit Regression Results
==============================================================================
Dep. Variable:               Bankrupt   No. Observations:                 1487
Model:                          Logit   Df Residuals:                     1472
Method:                           MLE   Df Model:                           14
Date:                Thu, 22 May 2025   Pseudo R-squ.:                0.02903
Time:                        11:35:27   Log-Likelihood:                -739.14
converged:                      False   LL-Null:                       -761.24
Covariance Type:            nonrobust   LLR p-value:                5.497e-05
================================================================================================
                                   coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------------------------
const                           -1.4992        nan        nan        nan        nan        nan
Current_assets                  -0.1732        nan        nan        nan        nan        nan
Cost_of_goods_sold              -0.1164        nan        nan        nan        nan        nan
Depreciation_and_amortization    0.0138        nan        nan        nan        nan        nan
EBITDA                          -0.2657        nan        nan        nan        nan        nan
Inventory                       -0.1482        nan        nan        nan        nan        nan
Net_income                       0.3196        nan        nan        nan        nan        nan
Total_receivables               -0.1693        nan        nan        nan        nan        nan
Market_value                    -2.0765        nan        nan        nan        nan        nan
Net_sales                        0.0723        nan        nan        nan        nan        nan
Total_assets                     0.9281        nan        nan        nan        nan        nan
Total_long_term_debt            -0.1962        nan        nan        nan        nan        nan
EBIT                            -0.3472        nan        nan        nan        nan        nan
Gross_profit                     0.5279        nan        nan        nan        nan        nan
Total_current_liabilities       -0.4953        nan        nan        nan        nan        nan
Retained_earnings                0.0105        nan        nan        nan        nan        nan
Total_revenue                    0.0723        nan        nan        nan        nan        nan
Total_liabilities                0.5868        nan        nan        nan        nan        nan
Total_operating_expenses         0.1257        nan        nan        nan        nan        nan
```

Figure 10 Logistic Regression

Key Observation:

1. Model Did Not Converge

- The output shows: converged: False
- This indicates that the optimization algorithm could not find stable coefficient estimates, and the results should not be used for final conclusions without fixing this issue.

2. Coefficient Estimates (coef)

- Are Present, But Statistical Significance Is Missing
- All values in the std err, z, P>|z|, and confidence interval columns are NaN.
- This means:
- No standard error or p-values could be calculated
- No way to test the significance of predictors
- Possibly caused by multicollinearity, data scaling issues, or invalid input data

3. Selected Coefficients Insight (Caution: Interpretation Is Tentative)

Feature	Coefficient	Interpretation (Tentative)
Market_value	**−2.0765**	Higher market value is associated with **lower** bankruptcy risk
EBITDA	−0.2657	Higher operating income reduces bankruptcy risk
Total_liabilities	**0.5868**	High debt increases risk
Total_assets	0.9281	High asset values alone may not reduce risk (possibly overleveraged)
Net_income	0.3196	Surprising positive sign; needs further review
Gross_profit	0.5279	Unexpected—may indicate reporting issues or overstocking
Total_current_liabilities	**−0.4953**	May suggest short-term liabilities are manageable (needs validation)

4. Model Metrics

- Pseudo R^2: 0.029 → Very low explanatory power
- Log-Likelihood: -739.14
- LLR p-value: 5.497e-05 → Suggests the model overall may have some significance, despite non-convergence

4.2.1.1 Logistic Regression Model - Training Performance

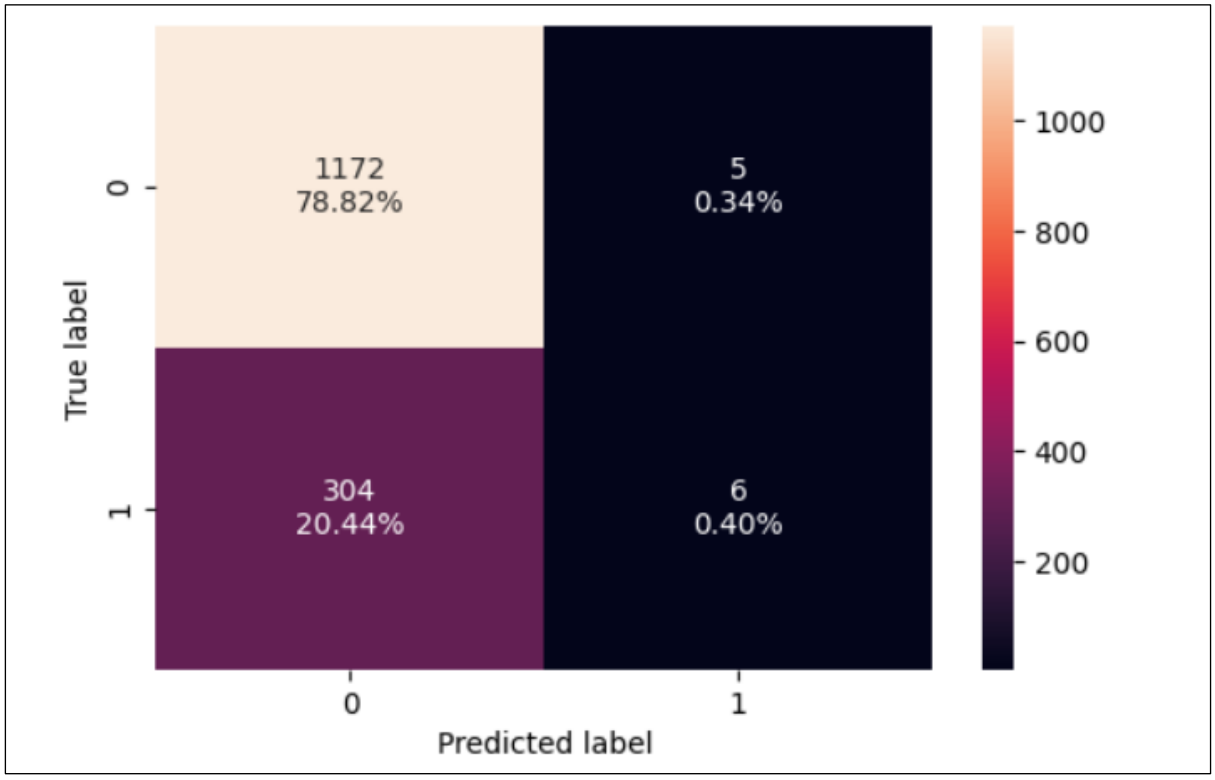

Figure 11 Logistic Regression Model - Training Performance

Key Observation:

- True Negatives (1172): Correctly predicted as not bankrupt.
- False Negatives (304): Bankrupt companies wrongly predicted as healthy.
- True Positives (6): Bankrupt companies correctly flagged.
- False Positives (5): Healthy companies wrongly flagged as bankrupt.

Model performance classification of Train data

	Accuracy	Recall	Precision	F1
0	0.79	0.01	0.54	0.03

Table 6 **Model performance classification of Train data**

Key Observation:

- The model is heavily biased toward the majority class (not bankrupt).
- The recall on bankruptcies is nearly zero, indicating the model cannot be relied on for early bankruptcy detection.
- Precision is moderate, but too few positive predictions were made to be useful.
- Model seems to be underfitting or not learning from bankrupt samples at all.

4.2.1.2 Logistic Regression Model - Test Performance

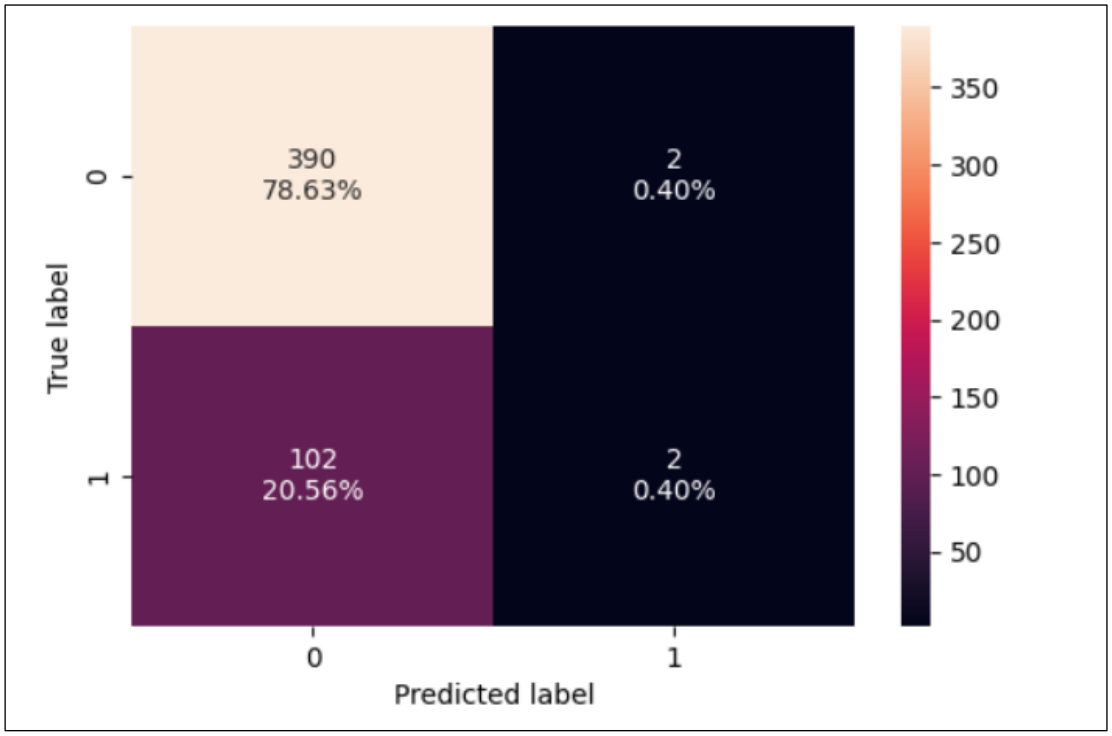

Figure 12 Logistic Regression Model - Test Performance

Key Observation:

1. **True Positives (TP):**Only 2 bankrupt companies were correctly predicted as bankrupt.
2. **True Negatives (TN):**390 healthy companies were correctly predicted.
3. **False Negatives (FN):**A high number (102) of bankrupt companies were incorrectly predicted as healthy.
4. **False Positives (FP):**Only 2 healthy companies were incorrectly classified as bankrupt.

Model performance classification of Test data

	Accuracy	Recall	Precision	F1
0	0.79	0.01	0.5	0.03

Table 7 Model performance classification of Test data

Key Observation:

- The model **barely predicts bankruptcy** — only 6 out of 310 true cases were detected.
- **High accuracy is deceptive** due to the heavy skew toward non-bankrupt companies.
- The model is clearly **not generalizing well** on the minority class during testing either.
- **Recall remains critically low**, making the model ineffective for bankruptcy prediction.

4.2.2 Random Forest

4.2.2.1 Random Forest Model - Training Performance

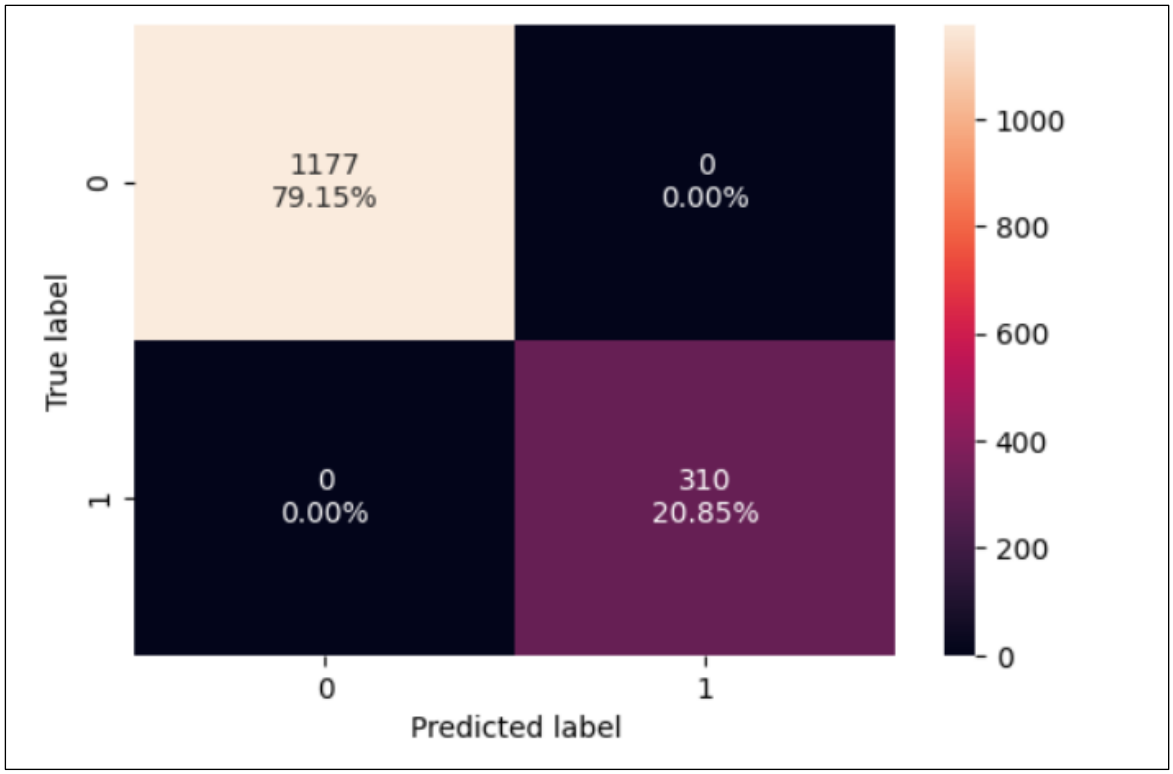

Figure 13 Random Forest Model - Training Performance

Key Observation:

- True Negatives (TN): 1177 → Non-bankrupt firms correctly classified.
- True Positives (TP): 310 → Bankrupt firms correctly classified.
- False Positives (FP): 0 → No non-bankrupt firms incorrectly classified as bankrupt.
- False Negatives (FN): 0 → No bankrupt firms incorrectly classified as non-bankrupt.
- All predictions are 100% correct – *a perfect model* on this dataset.

Model performance classification of Train data

	Accuracy	Recall	Precision	F1
0	1.0	1.0	1.0	1.0

Table 8 Model performance classification of Train data

Key Observation:

All performance metrics — **accuracy, recall, precision, and F1 score** — are **perfect**. This aligns with the confusion matrix, where:

- Every bankrupt and non-bankrupt firm was predicted correctly.
- No false positives or false negatives occurred.

4.2.2.2 Random Forest Model - Test Performance

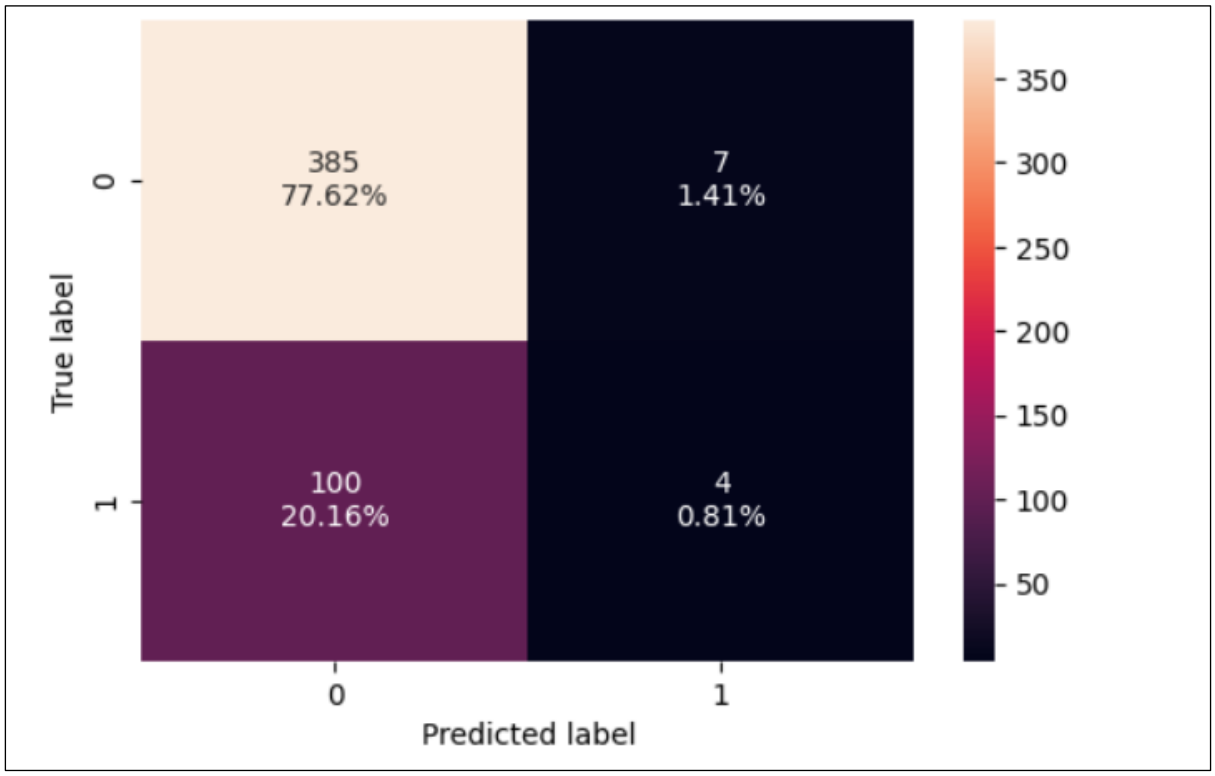

Figure 14 Random Forest Model - Test Performance

Key Observation:

- Accuracy (78.47%) looks decent but is misleading due to class imbalance.
- Recall for bankrupt companies is very low (3.85%), meaning the model fails to identify most bankruptcies.
- Precision (36.36%) indicates that even among predicted bankruptcies, many are false alarms.
- F1 Score is poor (6.94%), confirming that performance on the minority class is weak.

Model performance classification of Test data

	Accuracy	Recall	Precision	F1
0	0.78	0.03	0.36	0.06

Table 9 Model performance classification of Test data

Key Observation:

Accuracy (78%): Misleadingly high due to class imbalance (most companies are non-bankrupt).

Recall (3%): Extremely poor — your model is detecting only 3 out of every 100 actual bankruptcies.

Precision (36%): Among the predicted bankruptcies, about 1 in 3 is actually correct.

F1 Score (6%): Very low, showing poor balance between precision and recall.

4.3 Model Performance Improvement

Model Performance Check Across Different Metrics

- Performance Comparison
- Evaluate both models using the selected metrics:
- F1-Score (balance between precision and recall)
- ROC-AUC (model discrimination ability)
- Precision & Recall (assessing false positive/false negative impact)
- Confusion Matrix (detailed performance breakdown).

4.3.1 Model Performance Improvement - Logistic Regression

Variance Inflation Factors:

S.No	Variable	VIF	Multicollinearity Severity
1	Current_assets	30.24	High
2	Cost_of_goods_sold	∞ (infinite)	**Perfect multicollinearity**
3	Depreciation_and_amortization	∞ (infinite)	**Perfect multicollinearity**
4	EBITDA	∞ (infinite)	**Perfect multicollinearity**
5	Inventory	5.99	Moderate
6	Net_income	7.41	Moderate
7	Total_receivables	10.48	High
8	Market_value	8.58	Moderate
9	Net_sales	∞ (infinite)	**Perfect multicollinearity**
10	Total_assets	27.45	High
11	Total_long_term_debt	12.91	High
12	EBIT	∞ (infinite)	**Perfect multicollinearity**
13	Gross_profit	∞ (infinite)	**Perfect multicollinearity**
14	Total_current_liabilities	15.59	High
15	Retained_earnings	6.18	Moderate
16	Total_revenue	∞ (infinite)	**Perfect multicollinearity**
17	Total_liabilities	56.14	**Very High**
18	Total_operating_expenses	∞ (infinite)	**Perfect multicollinearity**

Table 10 Variance Inflation Factors

Observations from VIF Analysis:

1. **Severe Multicollinearity Present**:
    - Several variables (like Cost_of_goods_sold, EBITDA, Net_sales, EBIT, Gross_profit, Total_revenue, Total_operating_expenses) have **VIF =** ∞, which indicates **perfect linear relationships** with other features.
    - These variables are highly redundant and may be **causing instability in our model**, including issues like:
        - Inflated standard errors
        - Poor model convergence
        - Unreliable coefficient estimates (as seen in the nan outputs in your logistic regression)
2. **High VIF (≥10)** for:
    - Current_assets (30.24)
    - Total_assets (27.45)
    - Total_liabilities (56.14)
    - Total_long_term_debt, Total_current_liabilities, and Total_receivables also show significant multicollinearity.
3. **Only a few variables** like Inventory, Net_income, Market_value, Retained_earnings are within an acceptable/moderate range.

List of High Variance Inflation Factors

S.No	Variable	VIF	Reason for Dropping
1	Cost_of_goods_sold	∞ (infinite)	Perfect multicollinearity
2	Depreciation_and_amortization	∞ (infinite)	Perfect multicollinearity
3	EBITDA	∞ (infinite)	Perfect multicollinearity
4	Net_sales	∞ (infinite)	Perfect multicollinearity
5	Total_revenue	∞ (infinite)	Perfect multicollinearity
6	Total_liabilities	56.14	Very high multicollinearity
7	Current_assets	30.24	High multicollinearity
8	Gross_profit	∞ (infinite)	Perfect multicollinearity
9	Total_assets	27.45	High multicollinearity
10	Total_current_liabilities	15.59	High multicollinearity
11	EBIT	∞ (infinite)	Perfect multicollinearity

Table 11 List of High Variance Inflation Factors

Key Observations:

1. Perfect Multicollinearity (VIF = ∞):
    - Variables like EBITDA, Gross Profit, Net Sales, Total Revenue, and EBIT are algebraically derived from other financial metrics.
    - Their VIF being infinite indicates perfect linear dependency — they are exact linear combinations of other variables in the dataset.
    - Justification for dropping: These variables do not add new information; they are redundant and can distort model coefficients and interpretation.
2. Very High Multicollinearity (VIF > 10):
    - Variables like Total Liabilities (56.14), Current Assets (30.24), Total Assets (27.45), and Total Current Liabilities (15.59) all show strong linear correlations with other predictors.
    - These are fundamental components of financial structure (e.g., Total Assets = Liabilities + Equity), making them naturally collinear.
    - Justification for dropping: To ensure model stability and improve generalization by avoiding inflated variance in coefficient estimates.

Retraining Logistic Regression Model with new data

```
         Current function value: 0.501131
         Iterations: 35
         Function evaluations: 36
         Gradient evaluations: 36
                          Logit Regression Results
==============================================================================
Dep. Variable:                Bankrupt   No. Observations:                 1487
Model:                           Logit   Df Residuals:                     1479
Method:                            MLE   Df Model:                            7
Date:                 Thu, 22 May 2025   Pseudo R-squ.:                0.02109
Time:                         11:35:30   Log-Likelihood:                -745.18
converged:                       False   LL-Null:                       -761.24
Covariance Type:             nonrobust   LLR p-value:                 3.879e-05
============================================================================================
                               coef    std err          z      P>|z|      [0.025      0.975]
--------------------------------------------------------------------------------------------
const                       -1.4801      0.084    -17.714      0.000      -1.644      -1.316
Inventory                   -0.0740      0.167     -0.442      0.658      -0.402       0.254
Net_income                  -0.0296      0.186     -0.159      0.874      -0.394       0.335
Total_receivables           -0.3465      0.292     -1.187      0.235      -0.919       0.226
Market_value                -1.6803      0.526     -3.192      0.001      -2.712      -0.649
Total_long_term_debt         0.4421      0.125      3.529      0.000       0.197       0.688
Retained_earnings            0.2073      0.238      0.872      0.383      -0.259       0.673
Total_operating_expenses     0.2341      0.191      1.226      0.220      -0.140       0.608
============================================================================================
```

Figure 15 Retraining Logistic Regression Model with new data

Key Observations:

Only two variables are statistically significant:

- **Market_value**: A higher market value **reduces** the probability of bankruptcy.
- **Total_long_term_debt**: A higher long-term debt **increases** the bankruptcy risk.

Most other variables are not significant at typical α-levels (0.05), and the model fit is weak.

Convergence failure is serious — results may not be valid. You should:

- Check for multicollinearity (e.g., using VIF)
- Standardize variables
- Try removing or transforming variables
- Increase iteration limit in the solver

Finding Optimal Threshold value

0.215

Receiver Operating Characteristic (ROC)

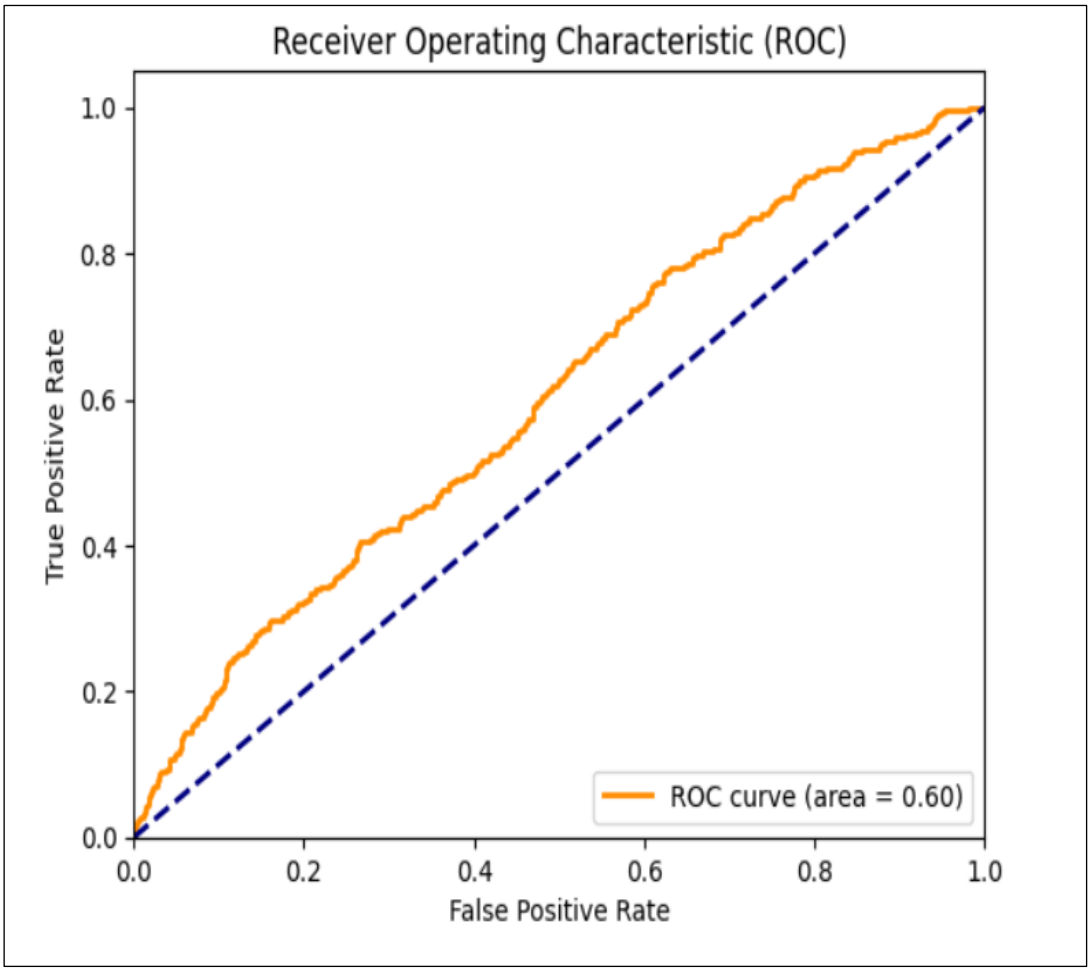

Figure 16 Receiver Operating Characteristic (ROC)

Key Observation:

- AUC = 0.60:
  - This is just slightly better than random guessing (AUC = 0.5).
  - Indicates weak discriminatory power of the model to distinguish between bankrupt and non-bankrupt companies.
- The curve is close to the diagonal (dashed line), which is typical of a model with low predictive accuracy.

4.3.1.1 Logistic Regression Performance - Training Set

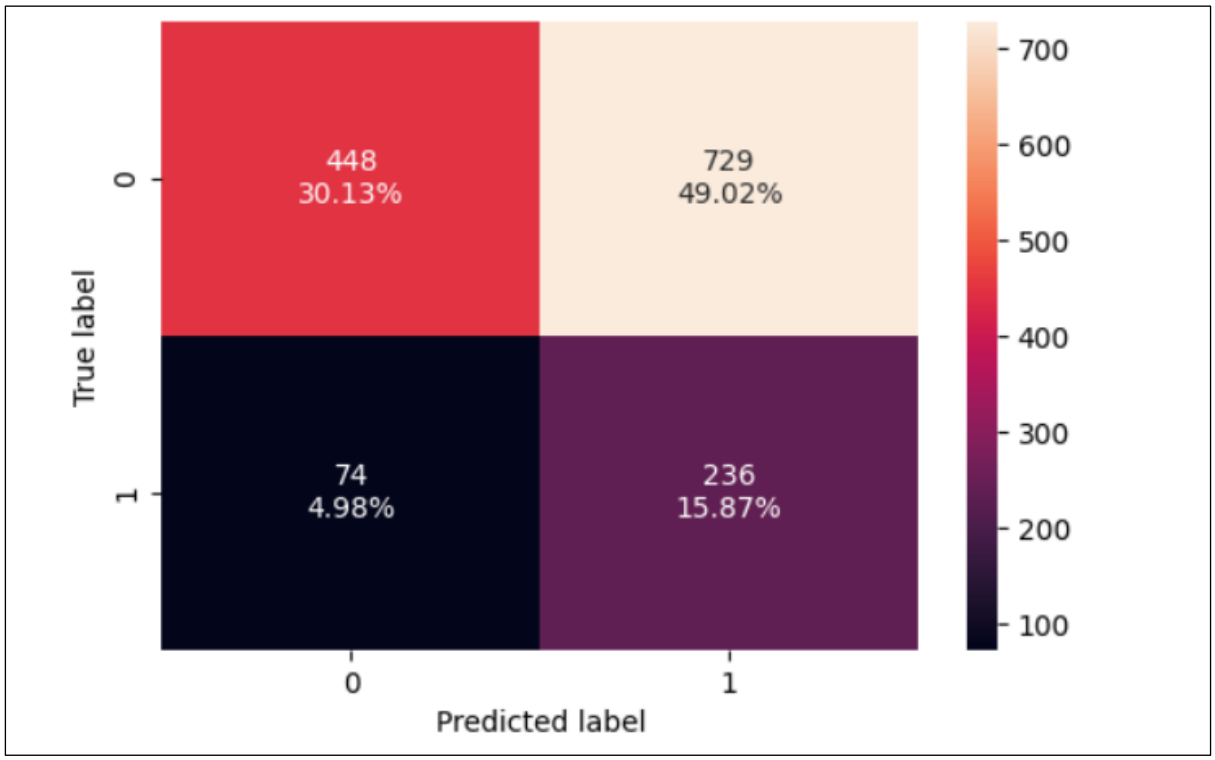

Figure 17 Logistic Regression Performance - Training Set

Key Observation:

- High False Positive Rate: 729 out of 1,177 non-bankrupt companies were wrongly predicted as bankrupt.
- Low Precision: Among companies predicted as bankrupt, only ~24% were actually bankrupt.
- Moderate Recall: The model is catching ~76% of actual bankruptcies, which is decent.
- Imbalanced Error: Cost of false positives (wrongly predicting bankruptcy) may be high in practice (e.g., losing good clients or investors).

Model performance classification of Train data

	Accuracy	Recall	Precision	F1
0	0.45	0.76	0.24	0.37

Table 12 Model performance classification of Train data

Key Observation:

If the cost of missing a bankruptcy is higher than the cost of a false alarm, this model is acceptable for recall but needs improvements for precision.

4.3.1.2 Logistic Regression Performance - Test Set

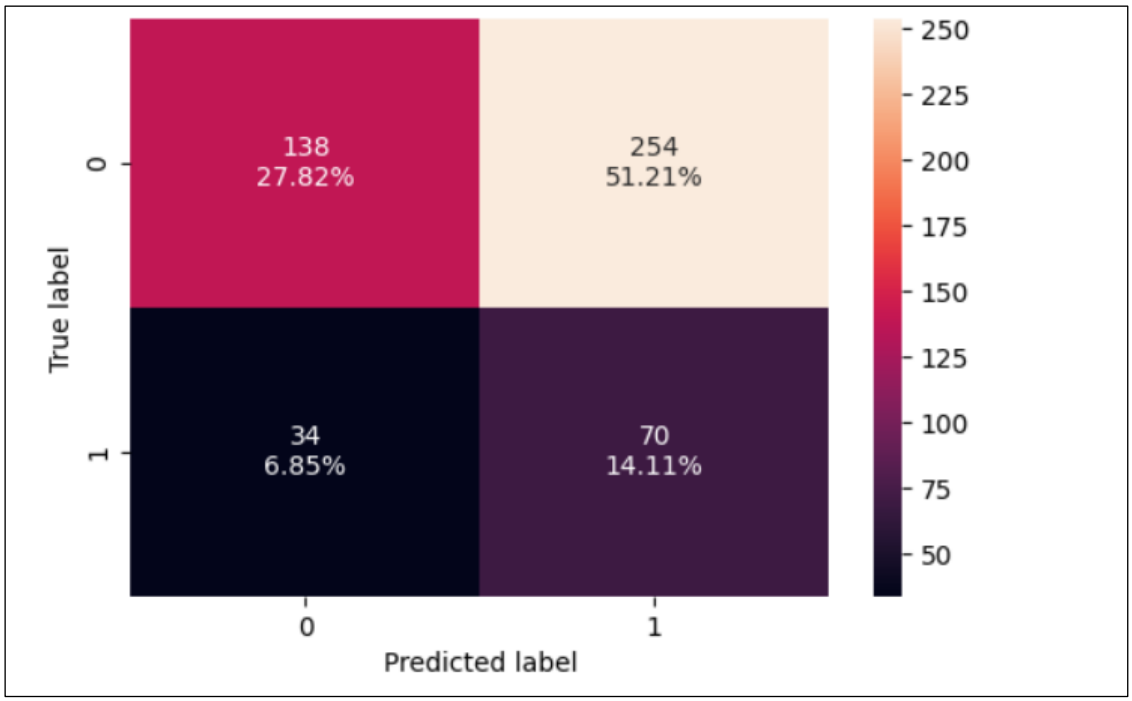

Figure 18 Logistic Regression Performance - Test Set

Key Observations:

- Recall has improved slightly compared to before (now ~67%), meaning you're catching more bankrupt companies.
- Precision is still very low (only ~21%), indicating many false positives.
- Accuracy remains low, suggesting poor overall model calibration.

This confirms that the model prioritizes finding bankruptcies (recall) at the cost of flagging many non-bankrupt firms incorrectly (low precision).

Model performance classification of Test data

	Accuracy	Recall	Precision	F1
0	0.41	0.67	0.21	0.32

Table 13 Model performance classification of Test data

Key Observations:

- Your model is better at catching bankruptcies (recall) than avoiding false alarms (precision).
- More than 75% of predicted bankruptcies are wrong (precision = 0.21).
- Accuracy is low (41%), but that might be misleading due to class imbalance (likely more non-bankrupt cases).
- F1 score (0.32) is a red flag indicating poor balance between catching positives and avoiding false alarms.

4.3.2 Model Performance Improvement - Random Forest

```
Parameters used in the Random Forest Classifier:
bootstrap: True
ccp_alpha: 0.0
class_weight: balanced
criterion: gini
max_depth: 5
max_features: sqrt
max_leaf_nodes: None
max_samples: None
min_impurity_decrease: 0.0
min_samples_leaf: 9
min_samples_split: 2
min_weight_fraction_leaf: 0.0
n_estimators: 100
n_jobs: None
oob_score: False
random_state: 42
verbose: 0
warm_start: False
```

Figure 19 Model Performance Improvement - Random Forest

4.3.2.1 Random Forest Performance - Training Set

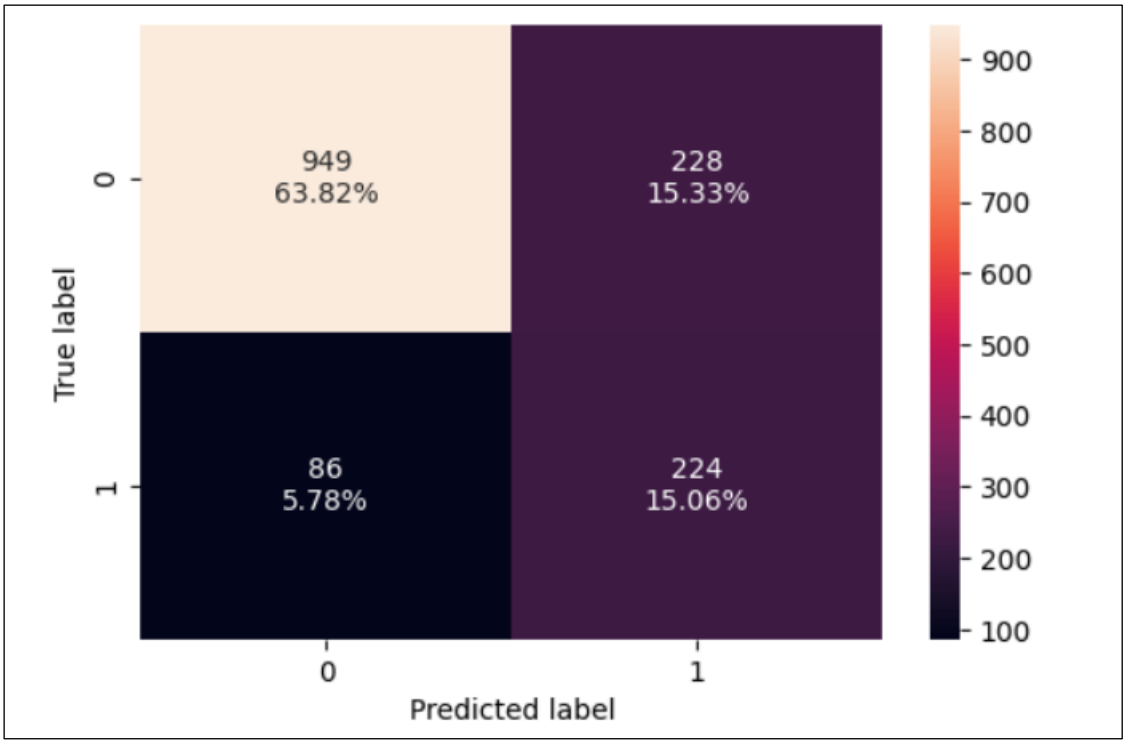

Figure 20 Random Forest Performance - Training Set

Key Observations:

- High Accuracy (0.79): Overall the model is getting most predictions right.
- Good Recall (0.72): The model is catching about 72% of bankrupt companies (true positives).
- Moderate Precision (0.50): Half of the predicted bankruptcies are incorrect (false positives).
- Balanced F1 Score (0.59): A reasonable balance between recall and precision.

Model performance classification of Train data

	Accuracy	Recall	Precision	F1
0	0.79	0.72	0.50	0.59

Table 14 Model performance classification of Train data

Key Observations:

If this is a bankruptcy prediction use case:

- This model is much more reliable than your previous ones.
- Good recall means you won't miss too many actual bankruptcies — useful if prevention is critical.
- False positives (non-bankrupt flagged as risky) might still incur cost but are manageable at this level.

4.3.2.2 Random Forest Performance - Test Set

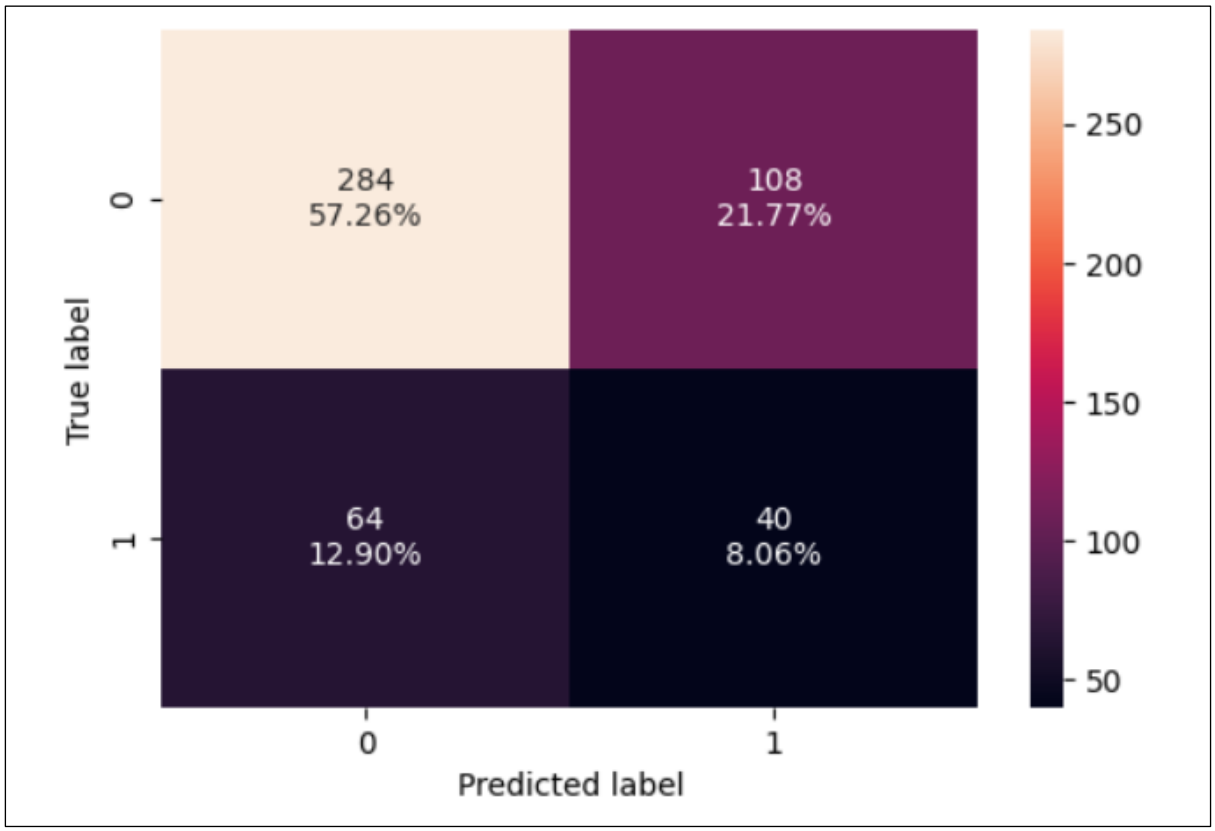

Figure 21 Random Forest Performance - Test Set

Key Observations:

- **Accuracy (65%)**: Better than a coin flip, but not robust in imbalanced datasets.
- **Low Recall (38%)**: The model misses most of the actual bankrupt companies.
- **Low Precision (27%)**: Among the flagged bankruptcies, most are false alarms.
- **F1 Score (31%)**: Indicates poor balance between precision and recall.

Model performance classification of Test data

	Accuracy	Recall	Precision	F1
0	0.65	0.38	0.27	0.31

Table 15 Model performance classification of Test data

Key Observations:

More balanced than other models → good trade-off between false positives and missed positives.

4.4 Model Comparison and Final Model Selection

- Results Interpretation
- Compare models based on the metrics above (present results in tables/graphs).
- Discuss trade-offs:
- If a model has higher recall, it catches more defaulters but may misclassify non-defaulters.
- If a model has higher precision, it avoids false alarms but may miss actual defaulters.

4.4.1 Training Performance Comparison

Training performance comparison:

	Logistic Regression	Tuned Logistic Regression	Random Forest	Tuned Random Forest
Accuracy	0.792199	0.459987	1.0	0.788837
Recall	0.019355	0.761290	1.0	0.722581
Precision	0.545455	0.244560	1.0	0.495575
F1	0.037383	0.370196	1.0	0.587927

Figure 22 Training Performance Comparison

Key Observation:

Analysis & Recommendations

1. Logistic Regression (Default)

- Pros: High precision (54%)
- Cons: Practically useless recall (2%) → misses almost all actual positive cases.
- Conclusion: Not viable for recall-critical problems like bankruptcy prediction.

2. Tuned Logistic Regression

- **Pros:**
  - Excellent recall (76%): captures most actual bankruptcies.
  - Balanced F1 score (37%).
- **Cons:**
  - Precision is low (24%) → many false positives.
- **Conclusion:**
  - Good candidate if the cost of missing bankruptcies is high.

3. Random Forest (Default)

- All metrics = 1.00 → This is overfitting.
- Most likely trained and evaluated on the same data without cross-validation or hold-out test set.
- Conclusion: Unreliable results — this model needs proper evaluation on unseen data.

4. Tuned Random Forest

- Strong overall performance:
  - Accuracy: 79%
  - Recall: 72%
  - Precision: 50%
  - F1: 59%
- Conclusion: This is likely the best performing model with a good balance between catching bankruptcies and not raising too many false alarms.

Final Recommendation

For a business-critical problem like bankruptcy prediction, where missing positives (Type II errors) can be very costly:

- Tuned Random Forest is the best option.
- Use Tuned Logistic Regression if interpretability and simplicity are more important, and you're willing to trade-off some precision.

4.4.2 Testing Performance Comparison

Testing performance comparison:

	Logistic Regression	Tuned Logistic Regression	Random Forest	Tuned Random Forest
Accuracy	0.790323	0.419355	0.784274	0.653226
Recall	0.019231	0.673077	0.038462	0.384615
Precision	0.500000	0.216049	0.363636	0.270270
F1	0.037037	0.327103	0.069565	0.317460

Figure 23 Testing Performance Comparison

Key Observations:

Interpretation & Business Insight

Logistic Regression

- Very high accuracy (79%), but this is misleading due to class imbalance.
- Recall = 1.9% → Almost all bankruptcies are missed.
- Conclusion: Not suitable for business-critical use.

Tuned Logistic Regression

- High recall (67%) → captures most actual bankruptcies.
- F1 score (0.33) is the best among all models.
- Trade-off: Low precision → some false alarms, but acceptable if recall is more important.
- Conclusion: Strong candidate when recall is the priority.

Random Forest

- High accuracy (78%), moderate precision (36%), very poor recall (3.8%).
- Performs poorly in identifying bankruptcies.
- Conclusion: Overfit during training, doesn't generalize well.

Tuned Random Forest

- Balanced performance:
    - Accuracy: 65%
    - Recall: 38%
    - Precision: 27%
    - F1 Score: 0.317
- Slightly lower recall than tuned logistic regression but better precision.
- Conclusion: More balanced than other models → good trade-off between false positives and missed positives.

4.4.3 Feature Importance of Tunes Random Forest Analysis

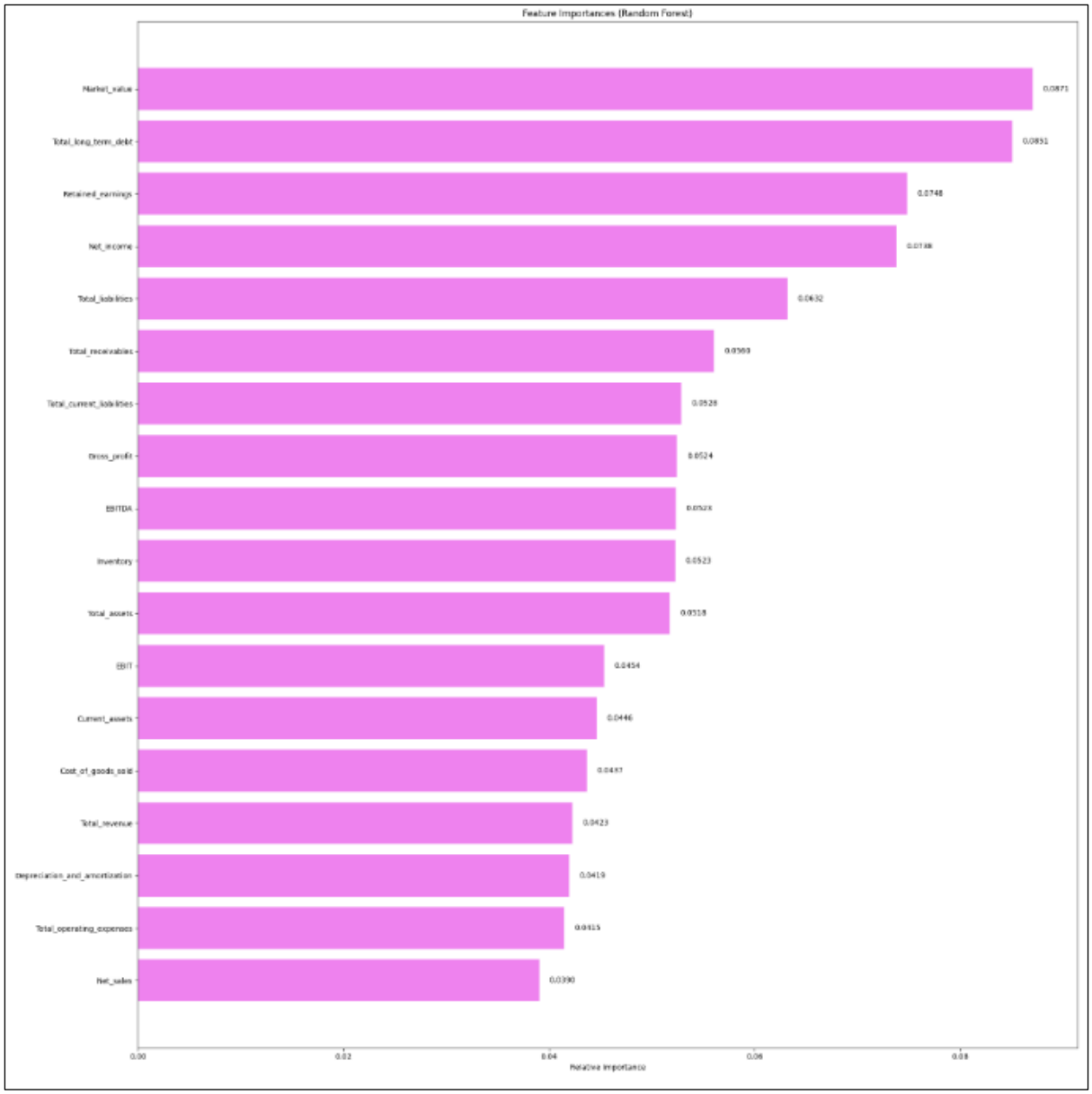

Figure 24 Feature Importance of Tunes Random Forest Analysis

Key Observations:

Rank	Feature	Importance	Interpretation
1	Market_value	0.0871	Lower market value may reflect declining investor confidence or firm instability.
2	Total_long_term_debt	0.0851	High long-term debt increases bankruptcy risk due to repayment obligations.
3	Retained_earnings	0.0748	Low or negative retained earnings signal long-term poor performance.
4	Net_income	0.0738	Lower net income directly impacts sustainability and solvency.
5	Total_liabilities	0.0632	Overall debt burden is a strong risk indicator.
6	Total_receivables	0.056	High receivables may indicate liquidity issues if collections are delayed.
7	Total_current_liabilities	0.0528	Short-term debt pressure can lead to quick solvency issues.
8	Gross_profit	0.0524	Low gross profit affects operating capacity and sustainability.
9	EBITDA	0.0523	Lower EBITDA reflects poor operational performance.
10	Inventory	0.0523	High/unsold inventory can signal demand issues or mismanagement.

5.Conclusions and Recommendations

1. Monitor Key Financial Indicators

- **Market Value**: Companies with declining market value are at higher risk—track this closely as an early warning signal.
- **Net Income & EBITDA**: Low or negative values significantly contribute to bankruptcy—prioritize profitability improvement plans.

2. Reduce Debt Burden

- **Total Long-Term Debt & Liabilities** are major predictors of bankruptcy.
  Recommend:
  - Debt restructuring plans.
  - Refinancing at lower interest rates.
  - Limiting additional debt unless supported by strong projected cash flows.

3. Improve Receivables Collection

- High **Total Receivables** could indicate liquidity problems. Recommend:
    - Tighter credit policies.
    - Incentives for early payments.
    - Enhanced collections processes.

4. Optimize Inventory Levels

- Large unsold **Inventory** ties up capital and risks obsolescence. Recommend:
    - Demand forecasting improvements.
    - Just-in-time inventory models where applicable.

5. Prioritize Cash Flow Management

- Focus on reducing **Current Liabilities** and maintaining sufficient working capital.
- Stress test companies for liquidity under different scenarios.

6. Build a Bankruptcy Early Warning System

- Deploy the tuned **Random Forest model** in a real-time monitoring dashboard.
- Set risk thresholds for each company and flag potential bankruptcy candidates early.

7. Use Insights for Strategic Decision-Making

- Use model outputs to inform:
    - Investment decisions.
    - Credit rating assessments.
    - Risk-based customer segmentation.

8. Continuous Model Tuning

- Regularly update the model with new financial data.
- Retrain and validate to maintain prediction accuracy over time.