Add Company Name

Business Case Study and Report

PRESENTED BY:

Write name of presenter here.

This event has live translations.

Agenda

Challenges and Objectives

EDA

Correlation and Multicollinearity

Preprocessing

Feature Engineering and model Building

Model Evaluation

Confusion Matrix and Classification Report

SHAP Plot

Conclusion

Problem Statement and Objective

01

Bank (financial) Problem Statement

The prediction of bankruptcy is a phenomenon of increasing interest in firms that stand to lose money because of unpaid debts. Since computers can store huge data sets pertaining to bankruptcy, making accurate predictions from them beforehand is becoming important. Company bankruptcy was defined based on the business regulations of the Netherlands (Financial Institution) in this project you will use various classification algorithms on the bankruptcy dataset to predict bankruptcies with satisfying accuracies long before the actual event.

02

Goal & Objectives

Goal & Objective: This exercise aims to build a model, using historical data that will determine the prediction of bankruptcy.

Challenges and Objectives

Challenges

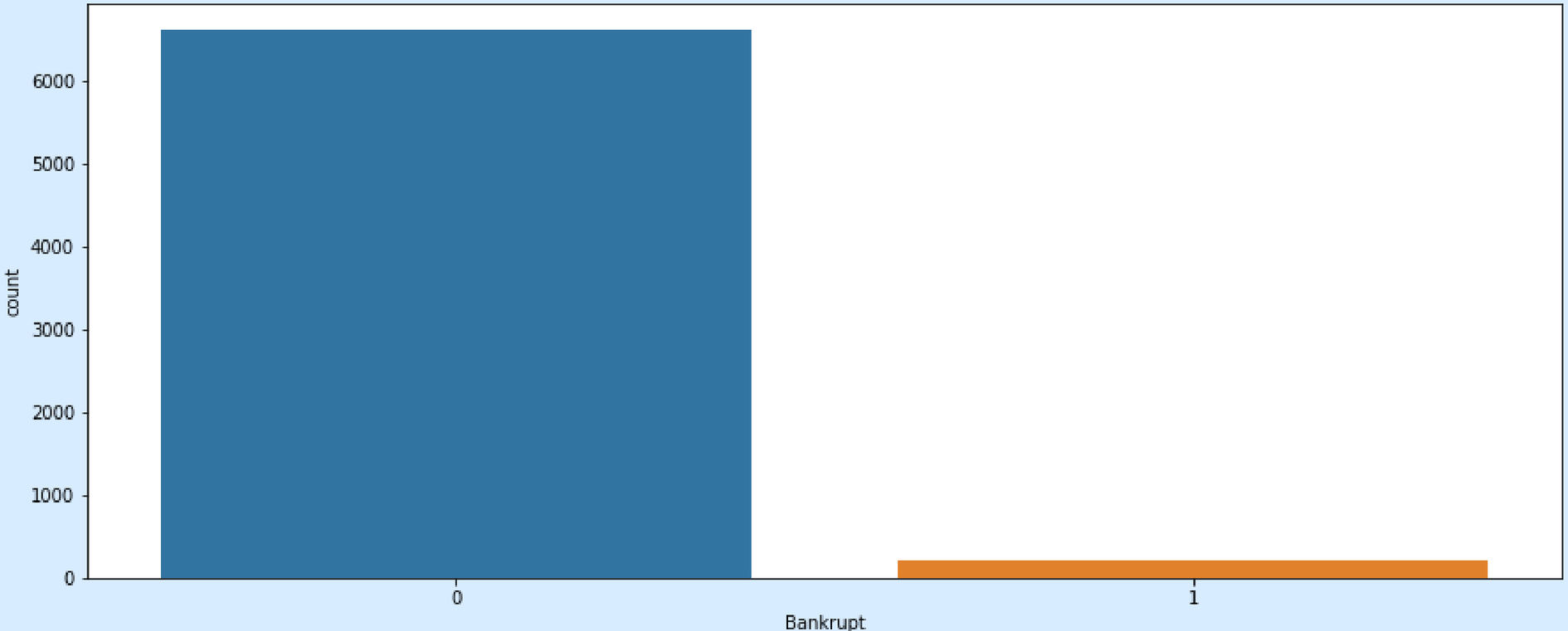
- dealing with high positive and negative correlated values
- Extreme values, skewness
- Class Imbalance
- Low variability
- Curse of Dimensionality

Objectives

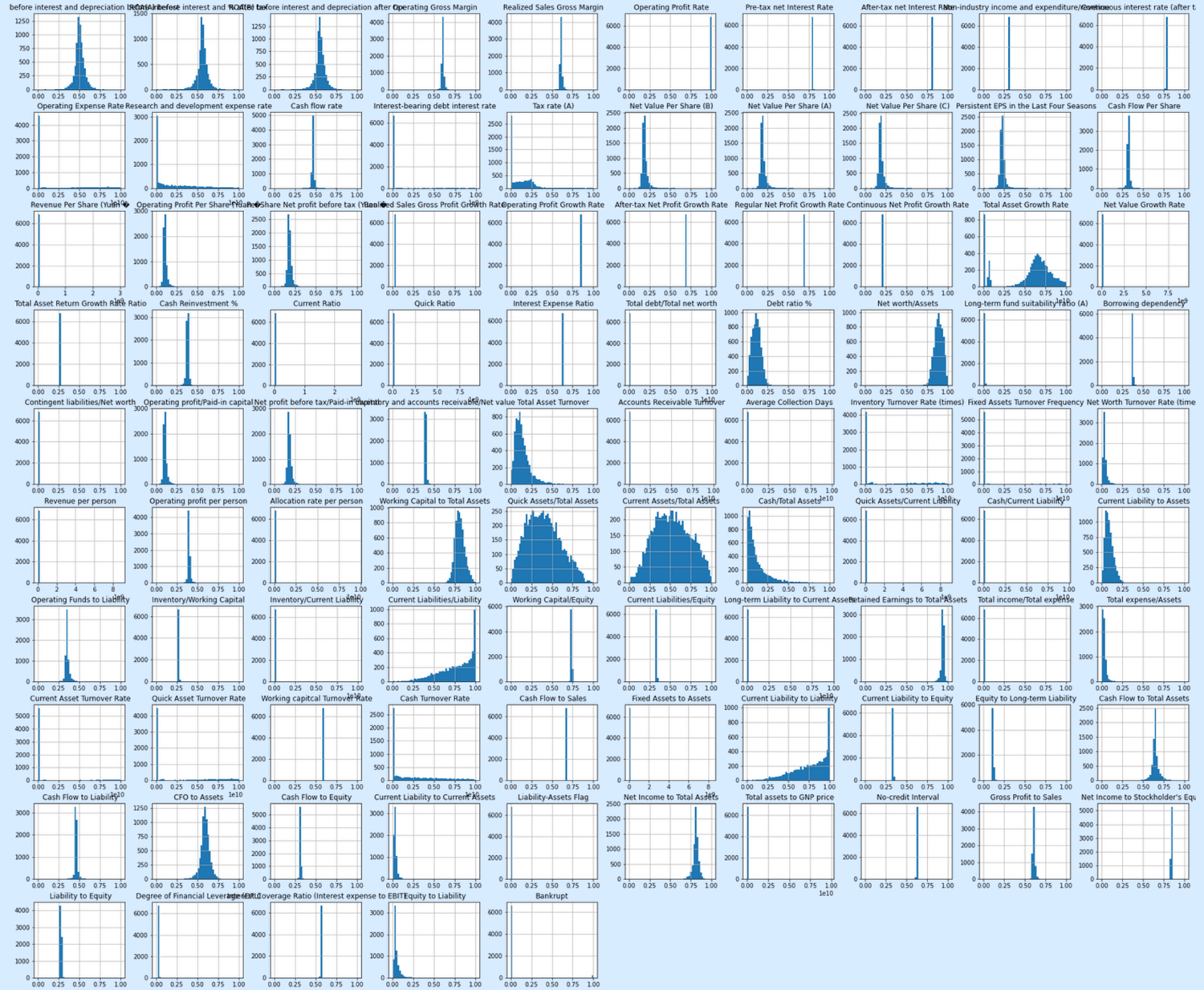
- Check correlation
- Perform multicollinearity test
- Binning, Winsorization CHAID
- ADASYN, SMOTE
- variance threshold
- PCA, Models(tree-based modes, boosting modes)

EDA

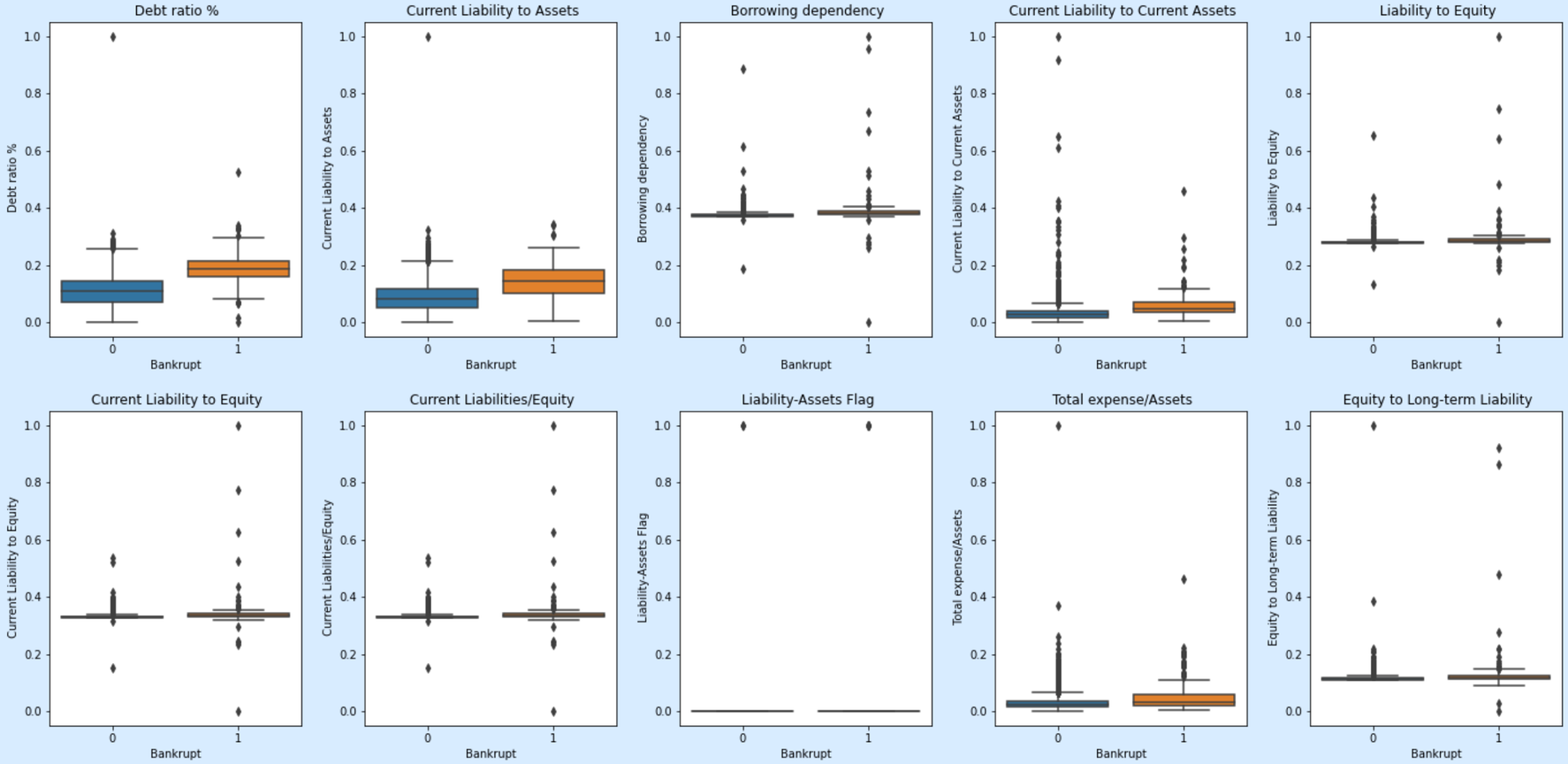
Looking at the plot below, we can see how our labels are strongly unbalanced, which is the main obstacle we need to solve to obtain good performance. Nearly 96.77 % of the companies are financially stable and 3.23 % are bankrupt. This is a case of extreme class imbalance.



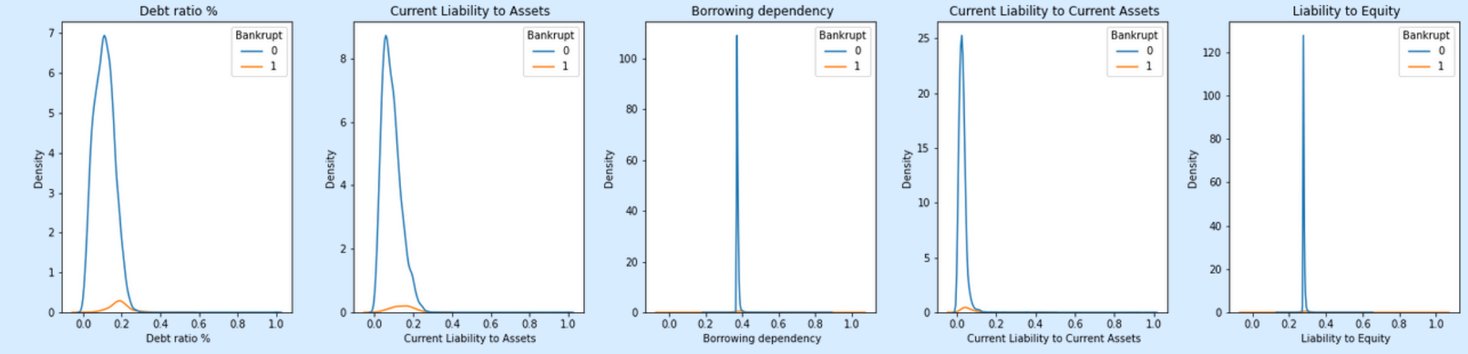
A large proportion of the data contains outliers and is heavily skewed, and in others, the values fall into just one bin. The outliers will be visualized and dealt with further to better understand how they affect our predictions. Additionally, we can observe some extreme values that could lead to high-standard errors.

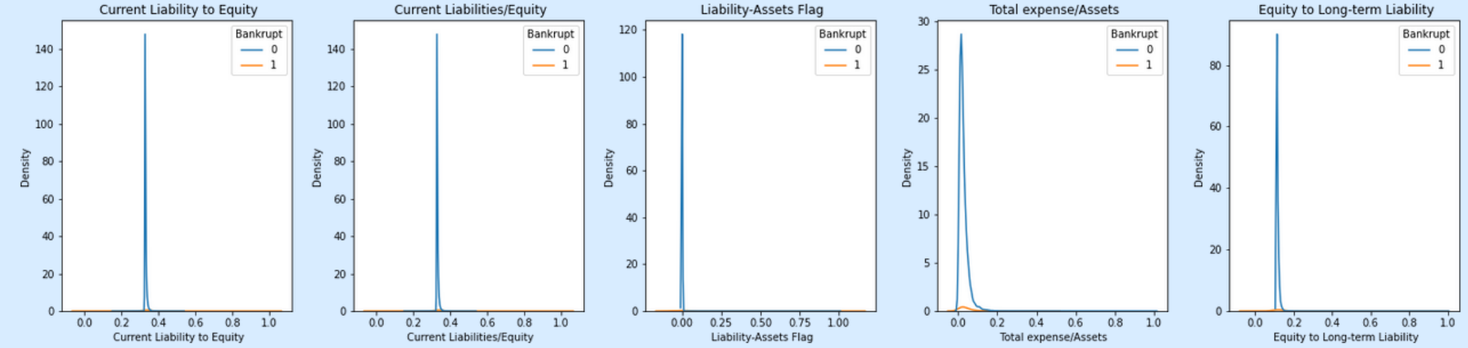


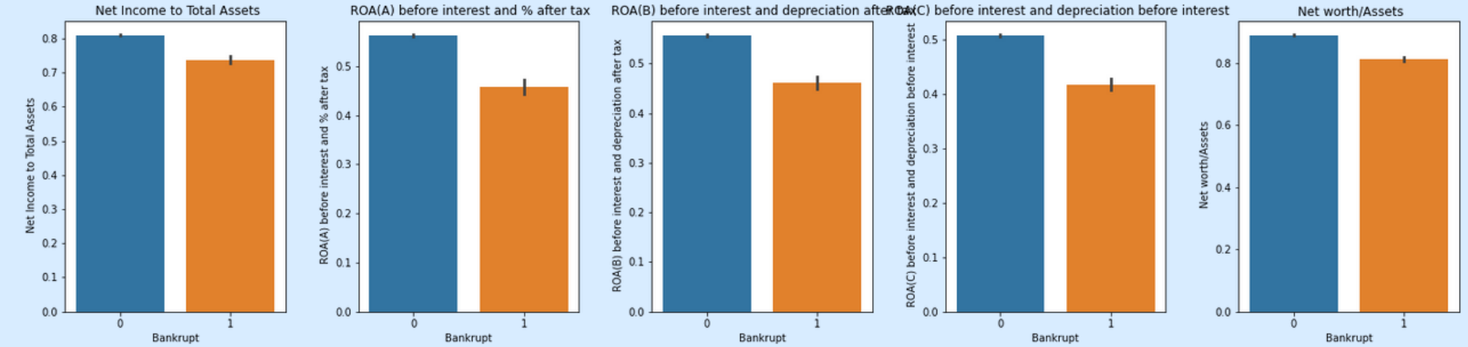
From the boxplot, it is clear that variables Net Income to Total Assets, Total debt/Total net worth, Debt ratio %, Net worth/Assets, Working Capital to Total Assets, Cash/Total Assets, Current Liability to Assets, Retained Earnings to Total Assets shows sign of a healthy company. It is observed that the “Debt Ratio %, Current Liability To Assets, Current Liability To Current Assets" attributes are a few of the attributes that have a high correlation with the target attribute. An increase in the values of the attributes “Debt Ratio %, Current Liability To Assets, Current Liability To Current Assets” causes an organization to suffer heavy losses, thus resulting in bankruptcy. An increase in the values of the attributes that have a negative correlation with the target attribute helps an organization avoid bankruptcy.

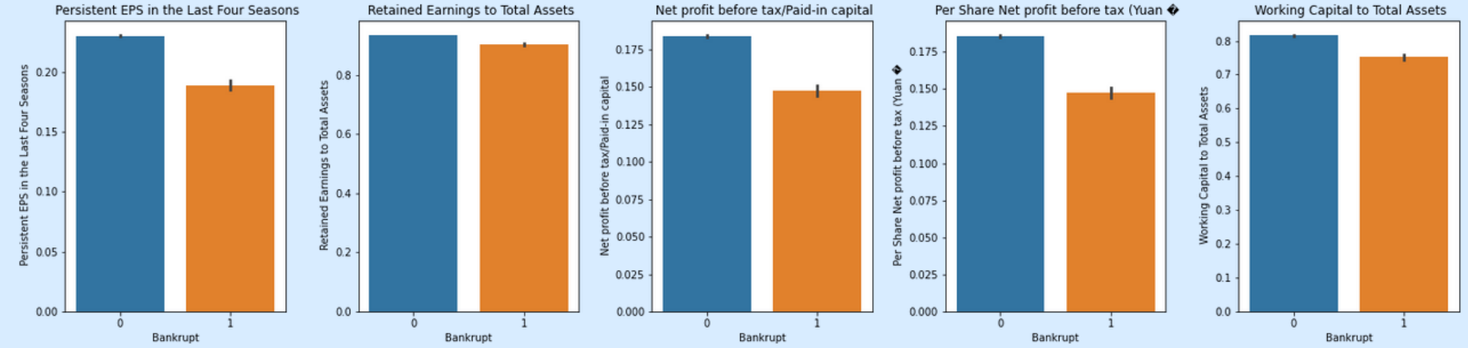


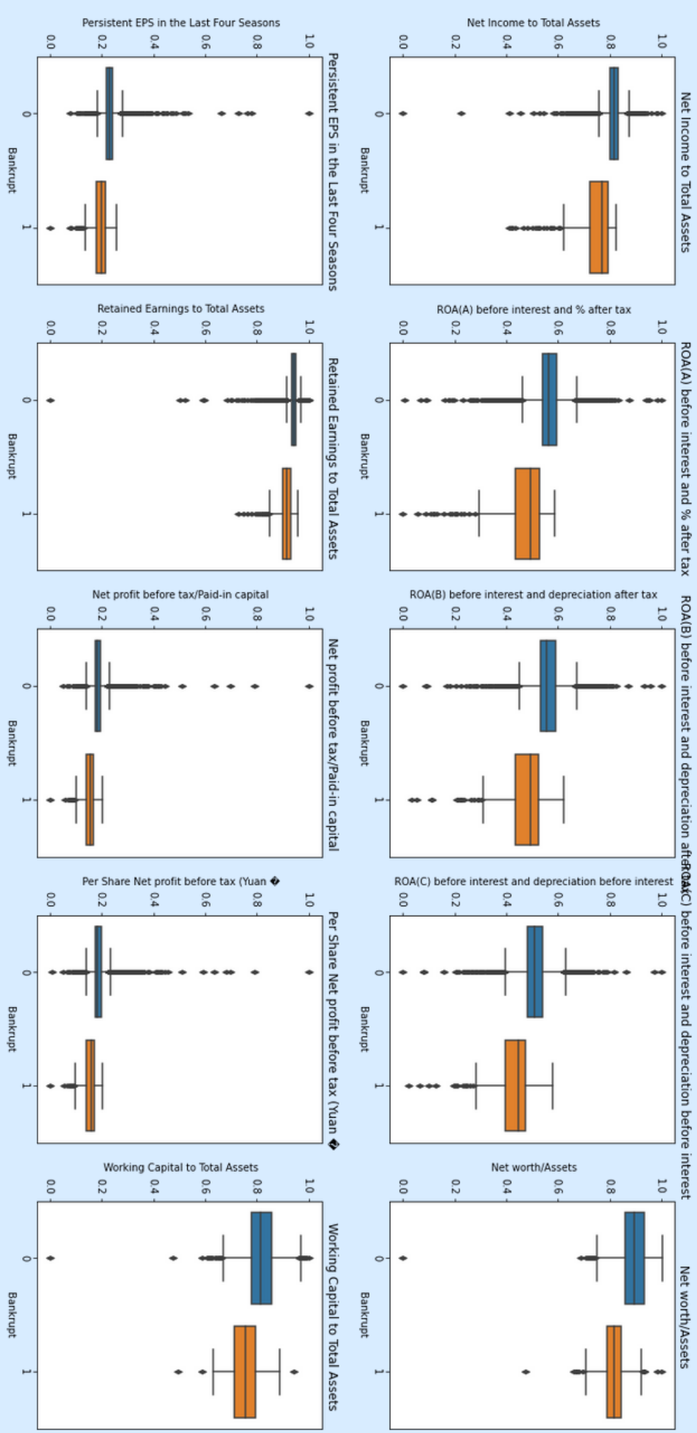
Next, we will visualize how positive and negative correlated variables impact our target variable, The purpose of this visualization is to help understand the impact of each feature on the likelihood of a company going bankrupt. By examining the patterns in the bar plots, kdeplots, and boxplots, business stakeholders can identify which factors are most closely linked to bankruptcy risk, and use this information to inform their decision-making. The 'positive_corr_lst' is a list of features that have been identified as having a positive correlation with the dependent variable 'Bankrupt'. The for loop iterates through each feature in the list, and generates a separate bar plot for each one. The x-axis of each bar plot represents the binary dependent variable 'Bankrupt' (0 = not bankrupt, 1 = bankrupt), while the y-axis represents the values of the corresponding feature. The bars in each plot represent the mean value of the feature for each value of the dependent variable. Overall, this visualization can help business stakeholders to gain insights into the factors that are most closely associated with bankruptcy risk and can be used to guide decision-making in areas such as risk assessment, lending, and investment.

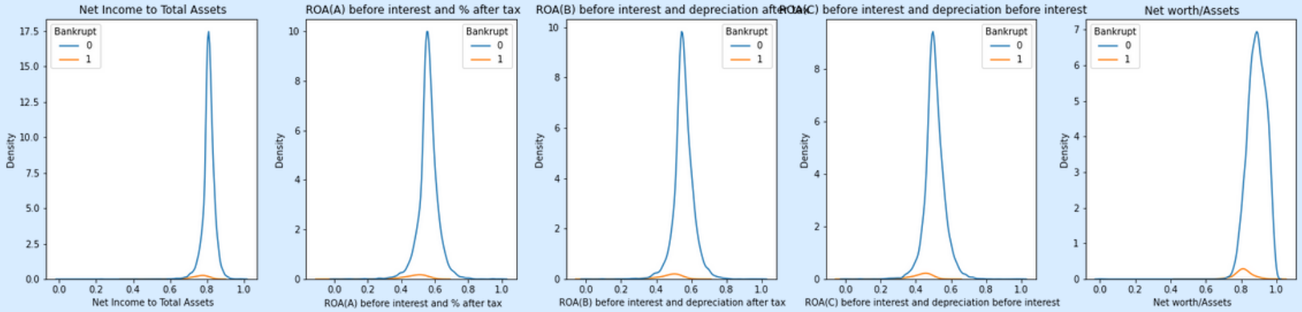

Debt ratio %
Current Liability to Assets
Borrowing dependency
Current Liability to Current Assets
Liability to Equity
Bankrupt
0
1
Density

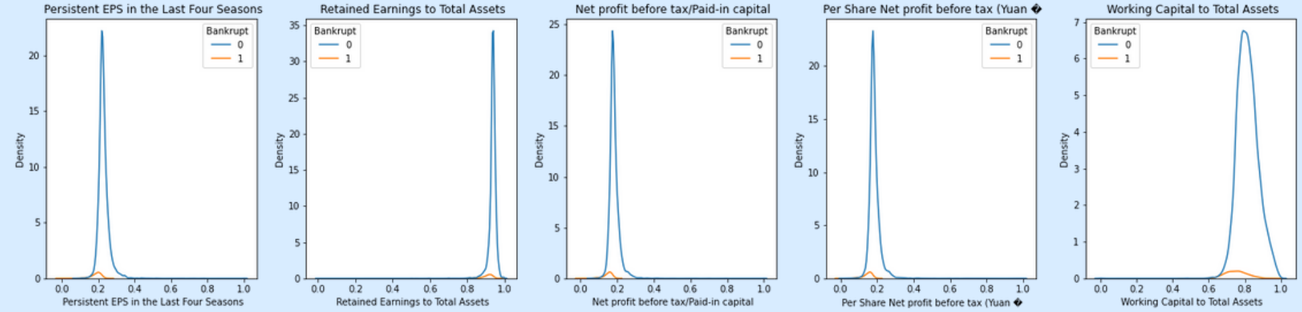

Current Liability to Equity
Current Liabilities/Equity
Liability-Assets Flag
Total expense/Assets
Equity to Long-term Liability
Bankrupt
0
1
Density

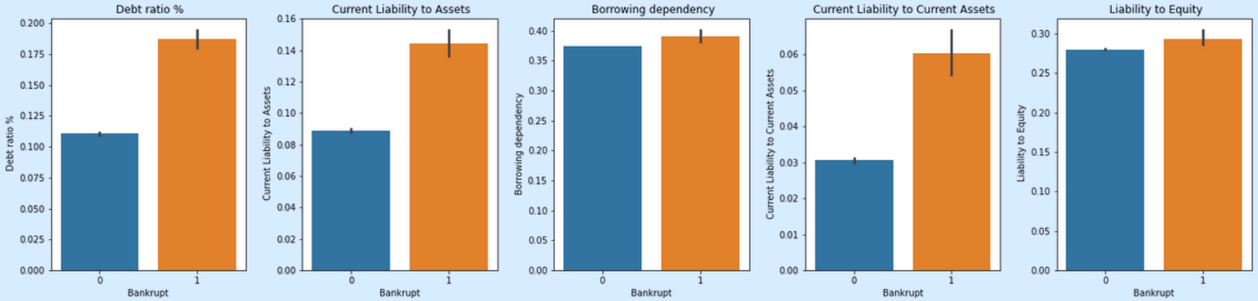

Net Income to Total Assets
ROA(A) before interest and % after tax
ROA(B) before interest and depreciation after tax
ROA(C) before interest and depreciation before interest
Net worth/Assets
Bankrupt

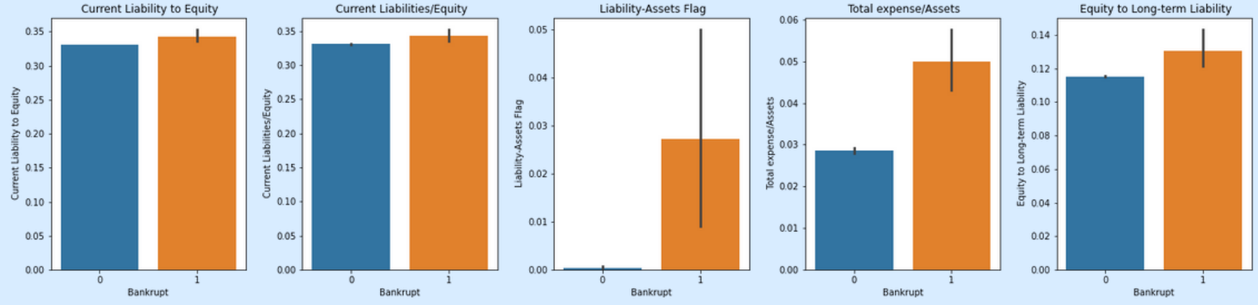

Persistent EPS in the Last Four Seasons
Retained Earnings to Total Assets
Net profit before tax/Paid-in capital
Per Share Net profit before tax (Yuan
Working Capital to Total Assets
Bankrupt

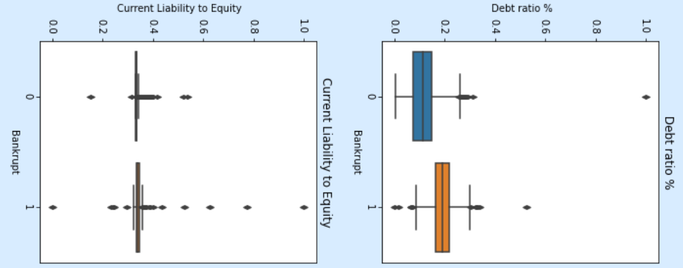

Persistent EPS in the Last Four Seasons
Net Income to Total Assets
Retained Earnings to Total Assets
ROA(A) before interest and % after tax
Net profit before tax/Paid-in capital
ROA(B) before interest and depreciation after tax
Per Share Net profit before tax (Yuan
ROA(C) before interest and depreciation before interest
Working Capital to Total Assets
Net worth/Assets
Bankrupt

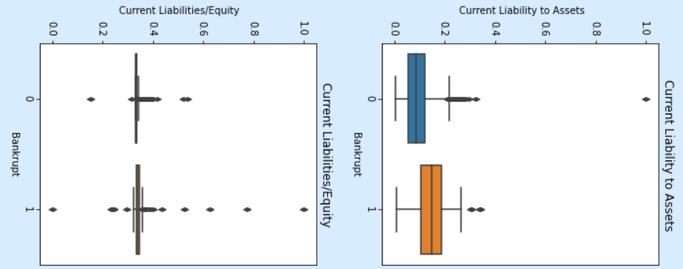

Net Income to Total Assets
ROA(A) before interest and % after tax
ROA(B) before interest and depreciation after tax
ROA(C) before interest and depreciation before interest
Net worth/Assets
Bankrupt
0
1
Density

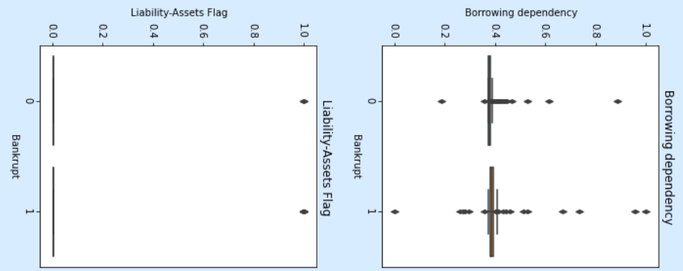

Persistent EPS in the Last Four Seasons
Retained Earnings to Total Assets
Net profit before tax/Paid-in capital
Per Share Net profit before tax (Yuan
Working Capital to Total Assets
Bankrupt
0
1
Density

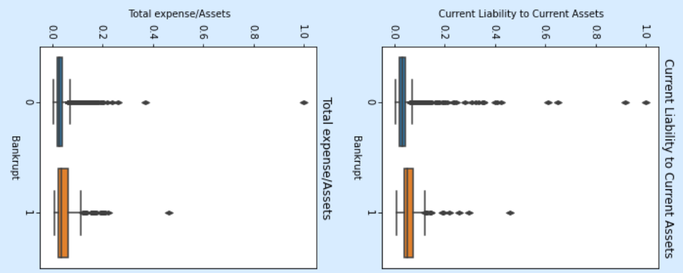

Debt ratio %
Current Liability to Assets
Borrowing dependency
Current Liability to Current Assets
Liability to Equity
Bankrupt

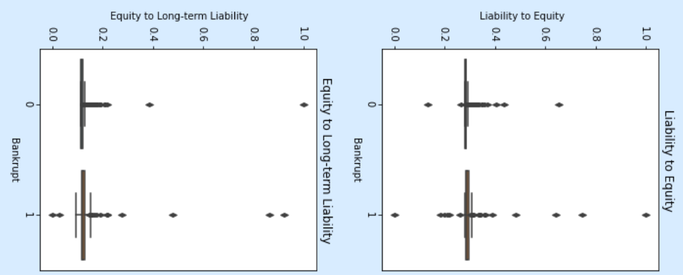

Current Liability to Equity
Current Liabilities/Equity
Liability-Assets Flag
Total expense/Assets
Equity to Long-term Liability
Bankrupt


Current Liability to Equity
Debt ratio %
Bankrupt


Current Liabilities/Equity
Current Liability to Assets
Bankrupt


Liability-Assets Flag
Borrowing dependency
Bankrupt


Total expense/Assets
Current Liability to Current Assets
Bankrupt


Equity to Long-term Liability
Liability to Equity
Bankrupt

Correlation and Multicollinearity

Introduction: In this analysis, we examined the relationships between various financial metrics of a company to identify potential patterns and insights.

Main Findings:

1. Perfect correlation: We found that there were some columns with a perfect correlation of 1, indicating that they are essentially measuring the same thing. This may suggest redundancy in the data or a need to explore the variables further to differentiate them.
2. High correlation: Most of the columns showed a correlation coefficient greater than 0.8 and less than 1. This suggests a strong positive relationship between the variables, indicating that changes in one metric are likely to be reflected in the other. Some notable examples include the Current Liabilities/Equity and Current Liability to Equity, Operating Gross Margin and Gross Profit to Sales, and Net Value Per Share (A) and Net Value Per Share (C).
3. Negative correlation: We also found negative correlations between some metrics. For example, the Debt ratio % showed a perfect negative correlation with Net worth/Assets, and there were negative correlations between Borrowing dependency and Net Income to Stockholder's Equity, and between Contingent liabilities/Net worth and Working Capital/Equity.

Conclusion: Overall, our correlation analysis provided valuable insights into the relationships between financial metrics, highlighting potential redundancies and key areas of strength and weakness. Further analysis and exploration of these metrics could provide additional insights for decision-making and strategic planning.

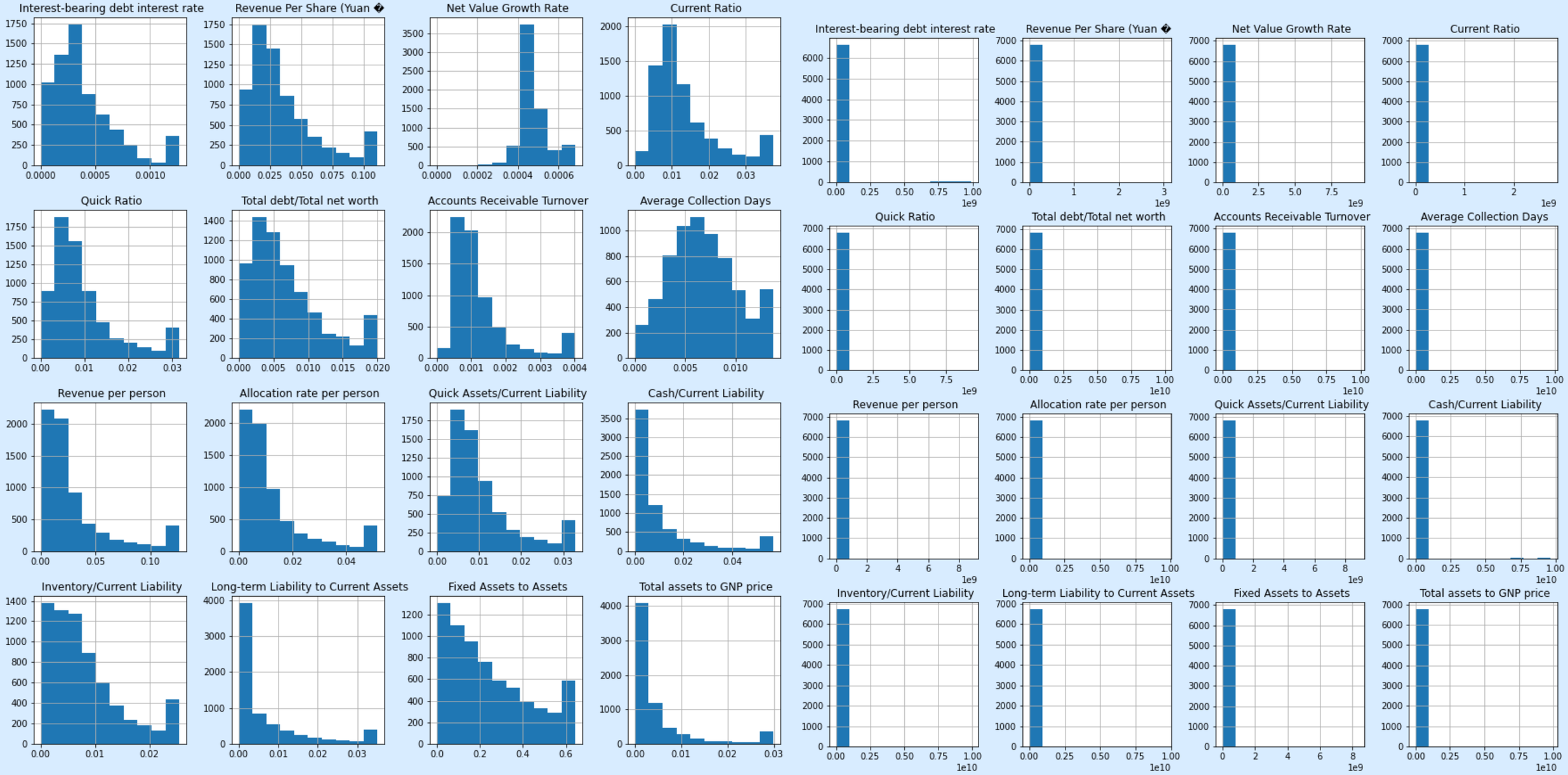
Correlation and Multicollinearity

n order to develop an accurate and reliable predictive model for our business, it is crucial to ensure that the independent variables we use are not heavily correlated with each other. Correlated variables can cause the model's coefficients to become unstable and lead to increased errors in our predictions. To address this issue, we will first manually identify and remove columns that have a high correlation coefficient (i.e., between 0.9 and 1) with other columns. We will then use VIF (Variance Inflation Factor) to detect and remove any remaining columns with high correlation. Furthermore, we will also check for variance in our dataset, as we have noticed that some columns have values ranging from 0 to 1 on a percentile basis of 0 to 75, but with some extreme values lying in the 75th-99th percentile. These outliers can skew our results and lead to inaccurate predictions. To address this issue, we will check each column's percentage of values greater than 1, and if it exceeds 30 percent, we will use winsorization to clip the upper values. This ensures that we maintain the integrity of the data while also removing the effects of outliers on our model's accuracy. By following these steps, we can ensure that our predictive model is reliable, accurate, and useful in making informed business decisions.

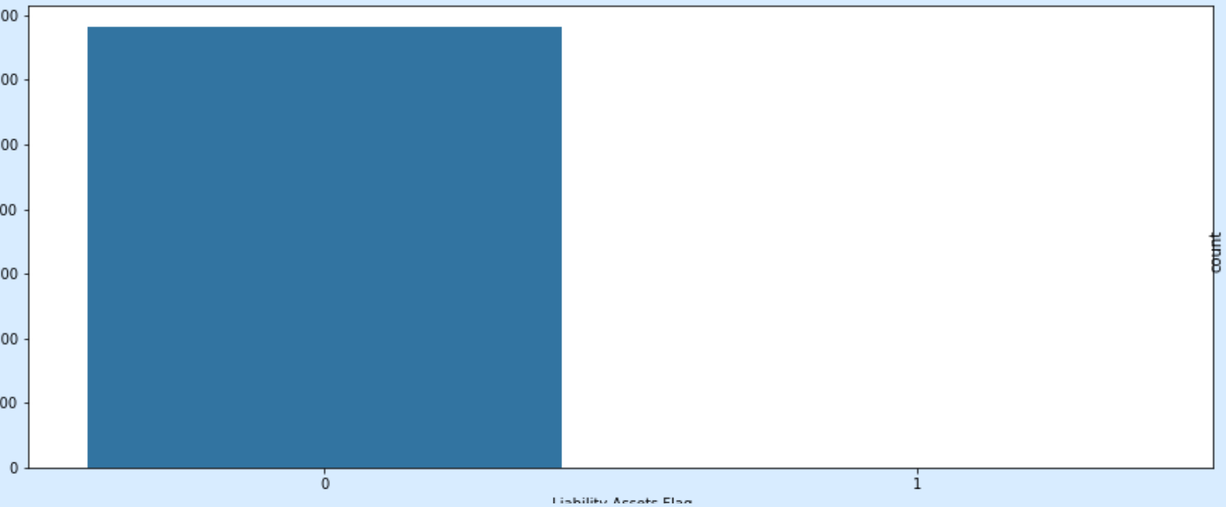
The chart shows the effect of Winsorization on a set of variables. Winsorization is a data transformation technique that replaces extreme values (outliers) with less extreme values to reduce their impact on the analysis. The chart displays the distribution of the variables before and after applying Winsorization.

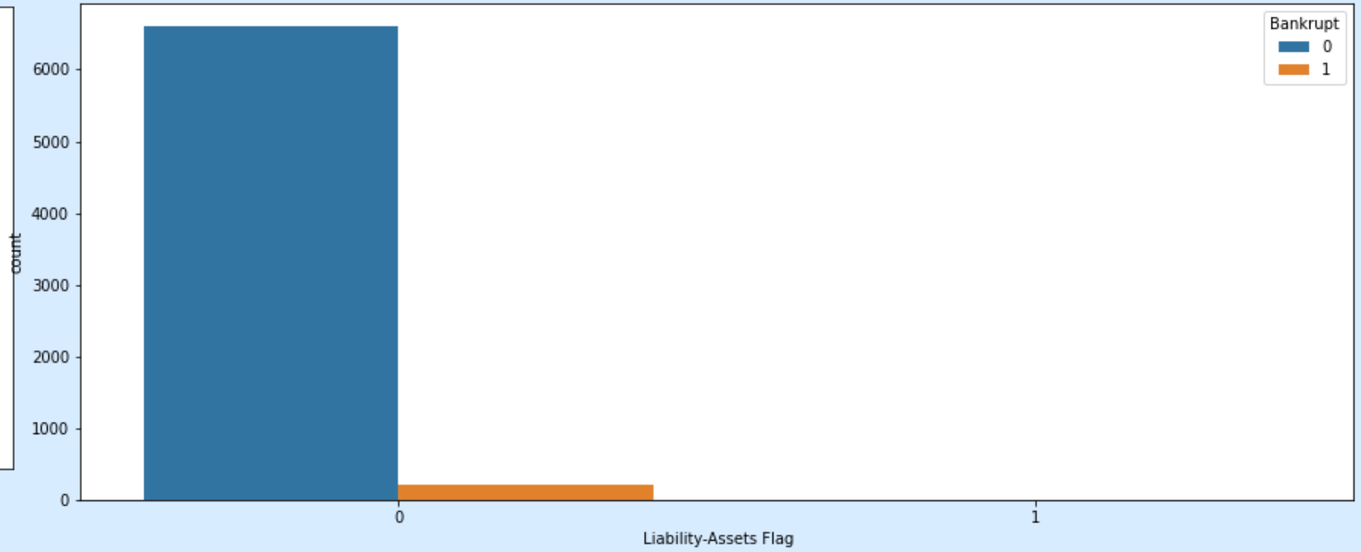
The left set of histograms shows the original distributions of the variables, while the right set of histograms shows the same variables after applying Winsorization. As we can see, the extreme values in the original distributions have been replaced by less extreme values, resulting in a more balanced distribution.

Winsorization can be a useful technique to improve the accuracy of statistical analysis by reducing the impact of outliers. However, it is important to use it with caution, as it can also distort the underlying distribution of the data and potentially introduce bias. Therefore, it is recommended to carefully evaluate the effects of Winsorization on the data before using it for analysis.


Interest-bearing debt interest rate
Revenue Per Share (Yuan ₹
Net Value Growth Rate
Current Ratio
Quick Ratio
Total debt/Total net worth
Accounts Receivable Turnover
Average Collection Days
Revenue per person
Allocation rate per person
Quick Assets/Current Liability
Cash/Current Liability
Inventory/Current Liability
Long-term Liability to Current Assets
Fixed Assets to Assets
Total assets to GNP price
Interest-bearing debt interest rate
Revenue Per Share (Yuan ₹
Net Value Growth Rate
Current Ratio
Quick Ratio
Total debt/Total net worth
Accounts Receivable Turnover
Average Collection Days
Revenue per person
Allocation rate per person
Quick Assets/Current Liability
Cash/Current Liability
Inventory/Current Liability
Long-term Liability to Current Assets
Fixed Assets to Assets
Total assets to GNP price

After winsorization and removing redundant columns using VIF and manually removing highly correlated columns, we notice that there is a column with a high variance and that is the 'Liability-Assets Flag', there were two unique values 1 and 0 The "Liability-Assets" flag denotes the status of an organization, where if the total liability exceeds total assets, the flagged value will be 1, else the value is 0. A majority number of times, organizations/companies' assets are more than their liabilities. A small portion of organizations suffers bankruptcy, although possessing more assets than their liabilities





Preprocessing

In order to improve the accuracy of our predictive model, we took steps to ensure that our data was normally distributed. We found that some of our variables were skewed, which can lead to biased results and reduce the effectiveness of the model. To address this issue, we applied feature transformation techniques to normalize these variables.

For variables with negative skewness, we used the yeo-jhonson power transformation method, while for variables with positive skewness, we used the log transformation method. This helped bring the variables closer to a normal distribution, which is necessary for accurate modeling. Additionally, we binned the values with low variance using the Optimal Binning library in Python. This process involves grouping similar values together to reduce the amount of noise in the data and improve the accuracy of the model. Standard scaling was not sufficient for our data, given its skewed nature. By applying these specific transformation techniques, we were able to ensure that our data was more reliable and could be used to build a more accurate model. This is crucial for our business as accurate predictions can help us make informed decisions and improve our overall performance. In summary, by using data transformation techniques and optimal binning, we were able to improve the quality of our data and build a more accurate model. This can have a significant impact on our business operations and help us make better decisions based on the insights derived from our data.

Feature Engineering and model Building

I used various feature selection techniques to identify the most important predictors for my predictive model. Firstly, I employed Recursive Feature Elimination (RFE) to select the top n features and cross-validated them using XGBoost as the base model. This helped me identify 40 important features that had a significant impact on predicting the dependent variable. To further refine the feature selection process, I then used the inbuilt feature importance of Random Forest and selected the best features using Sklearn's SelectKBest, SelectFdr, and SelectFpr methods. This helped me identify the most important predictors, which I used to build multiple models. After evaluating the performance of these models, I found that XGBoost provided better results than Random Forest, especially after oversampling using the ADASYN technique. This means that the selected features, when used with XGBoost, can accurately predict the dependent variable.

Overall, these feature selection techniques helped me narrow down the most important predictors, resulting in a more accurate and efficient predictive model.

Model Evaluation

Model evaluation is an important step in machine learning as it helps us determine the performance of our model and make informed decisions about its effectiveness. In my project, I used several methods for model evaluation to ensure that the model is performing well and can be trusted.

First, I used the confusion matrix to analyze the number of true positives, false positives, true negatives, and false negatives in our model predictions. This helped me understand how well the model was predicting the positive and negative classes. From the confusion matrix, I calculated the accuracy, precision, recall, and F1 score to evaluate the overall performance of the model.

Next, I used the classification report to further analyze the precision, recall, and F1 score of each class in our data. This allowed me to identify which class was being predicted well by the model and which one needed more improvement.

To evaluate the model's performance in terms of the receiver operating characteristic (ROC) curve, I used the roc_auc_curve method. This helped me understand how well the model was able to distinguish between the positive and negative classes.

Finally, I used the SHAP library to gain insights into how each feature contributed to the model's predictions. This allowed me to identify which features were most important in predicting the dependent variable.

Overall, using a combination of these methods for model evaluation helped me to gain a comprehensive understanding of the model's performance and to make informed decisions about its effectiveness. This ensured that the model was reliable and could be used for making predictions with confidence.

Confusion Matrix and Classification Report

The confusion matrix is a table that shows the number of true positives, false positives, true negatives, and false negatives of a classification model. In this case, the model predicted a total of 6575 samples with 3194 true negatives (TN) and 3160 true positives (TP). The model made 115 false positives (FP), which means it predicted the company would go bankrupt, but it actually did not. Similarly, the model made 106 false negatives (FN), which means it predicted that the company would not go bankrupt, but it actually did.

The classification report provides additional metrics such as precision, recall, and F1-score, which help us evaluate the performance of the model. Precision is the number of true positives divided by the sum of true positives and false positives, while recall is the number of true positives divided by the sum of true positives and false negatives. F1-score is the harmonic mean of precision and recall. In this case, the model has a precision of 0.97 and recall of 0.97 for both classes, which indicates that the model is good at identifying both bankrupt and non-bankrupt companies. The weighted average F1-score is also 0.97, which is a good overall measure of the model's performance.

In summary, the confusion matrix and classification report show that the model has a high accuracy in predicting bankruptcy with a good balance of precision and recall for both classes.

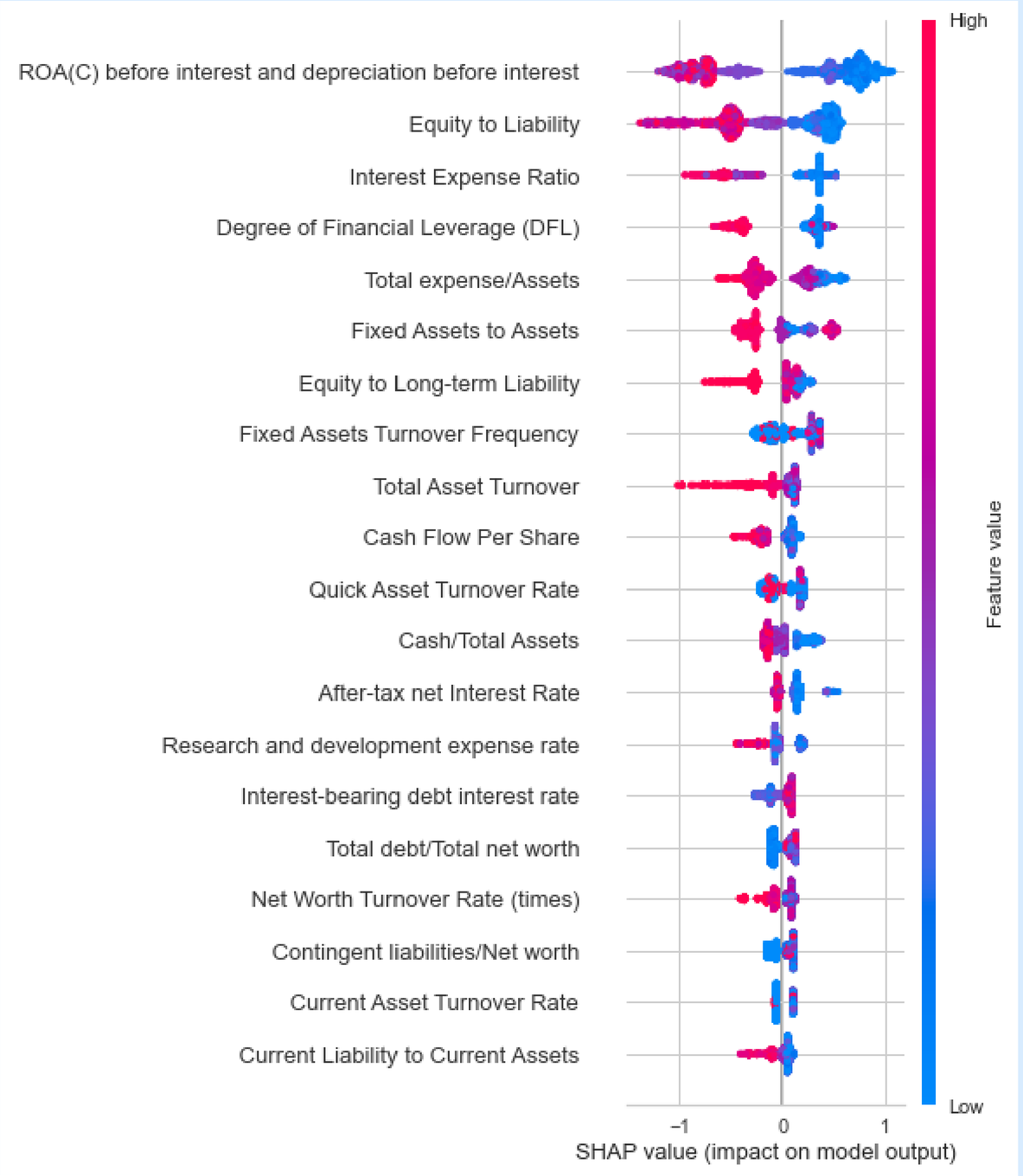
3194	115
106	3160

	precision	recall	f1-score	support
0	0.97	0.97	0.97	3309
1	0.96	0.97	0.97	3266
precision			0.97	6575
precision	0.97	0.97	0.97	6575
precision	0.97	0.97	0.97	6575

SHAP Plot

The SHAP (SHapley Additive exPlanations) plot is a type of graph that helps to interpret the output of a machine learning model. In this plot, the features of the data are ranked according to their importance in contributing to the final prediction of the model. The higher the feature on the Y-axis, the more important it is in predicting the output. Each dot on the plot represents a single instance of the data. The horizontal position of the dot shows the impact of the feature on the prediction, while the color of the dot indicates the actual value of the feature for that instance. This SHAP plot shows the most important features in predicting the output of the model. The red dots indicate that higher values of the feature contribute to a higher predicted value of the output, while blue dots indicate that lower values of the feature contribute to a higher predicted value of the output.

By analyzing this plot, we can identify which features are most influential in determining the output of the model. This can be useful in identifying areas where improvements can be made to the model, as well as providing insights into the underlying factors that are driving the model's predictions.


ROA(C) before interest and depreciation before interest
Equity to Liability
Interest Expense Ratio
Degree of Financial Leverage (DFL)
Total expense/Assets
Fixed Assets to Assets
Equity to Long-term Liability
Fixed Assets Turnover Frequency
Total Asset Turnover
Cash Flow Per Share
Quick Asset Turnover Rate
Cash/Total Assets
After-tax net Interest Rate
Research and development expense rate
Interest-bearing debt interest rate
Total debt/Total net worth
Net Worth Turnover Rate (times)
Contingent liabilities/Net worth
Current Asset Turnover Rate
Current Liability to Current Assets
High
Feature value
Low
−1
0
1
SHAP value (impact on model output)

Conclusion

In conclusion, we can say that building a predictive model is a complex process that requires a combination of data cleaning, feature engineering, and model selection. We used various techniques such as feature selection, feature transformation, oversampling, and model evaluation to build an accurate and reliable model. We also used different libraries and tools such as scikit-learn, XGBoost, OptimalBinning, SHAP, and matplotlib to implement these techniques and evaluate the model's performance. By using these techniques and tools, we were able to improve the accuracy and reliability of our model, making it more effective in predicting the dependent variable. The model's performance was evaluated using various metrics such as confusion matrix, classification report, ROC-AUC curve, and SHAP plots. Overall, the process of building a predictive model requires careful attention to detail, an understanding of the data, and the ability to choose the right tools and techniques. By following a systematic approach and leveraging the latest tools and techniques, we can build models that are accurate, reliable, and effective in solving real-world business problems.