





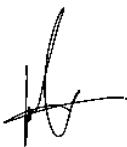

Course Code: UECS 3213 / UECS 3483 / UECS 3453

Course Name: DATA MINING

Lecturer: DR FATIMAH 'AUDAH MD. ZAKI

Academic Session: JANUARY 2023

Title: DATA MINING ASSIGNMENT

| GROUP NO | STUDENT NAME                   | STUDENT ID | PROGRAMME CODE | SIGNATURE   | FINAL MARK  |
|----------|--------------------------------|------------|----------------|---|---|
| 14       | Emily binti Mohd Syazwan Cheah | 2200449    | AM             |  | CLO 2: 40<br>CLO 3: 40<br>CLO 4: 40<br>Total: 120/120 |
|          | Darren Lai Tye Jie             | 2003164    | SE             |  |   |
|          | Lee Chin Yee                   | 2200494    | AM             |  |   |
|          | Chia Inn Zhan                  | 2102456    | SE             |  |   |

**MARKING RUBRIC**

| CLO | Criteria   | Poor | Satisfactory | Excellent        | Marks |
|-----|--|------|--------------|------------------|-------|
| 2   | <b>Data Understanding: (20 marks)</b> <ul style="list-style-type: none"> <li>Identifies the dataset and describes its source (5m)</li> <li>Summarizes the variables in the dataset (5m)</li> <li>Identifies potential issues or limitations with the dataset (5m)</li> <li>Demonstrates understanding of the dataset and its context (5m)</li> </ul>   |      |              | 5<br>5<br>5<br>5 | 20    |
|     | <b>Data Description (20 marks)</b> <ul style="list-style-type: none"> <li>Provides descriptive statistics and summaries of the dataset (5m)</li> <li>Identifies key features, patterns, and trends in the data (5m)</li> <li>Uses appropriate statistical and visual methods to summarize the data (5m)</li> <li>Demonstrates understanding of the data and its properties (5m)</li> </ul>       |      |              | 5<br>5<br>5<br>5 | 20    |
| 3   | <b>Data Preprocessing (20 marks)</b> <ul style="list-style-type: none"> <li>Performs data cleaning, transformation, and normalization (5m)</li> <li>Selects and justifies data sampling techniques (5m)</li> <li>Removes any irrelevant data or outliers (5m)</li> <li>Demonstrates understanding of data preprocessing techniques (5m)</li> </ul>   |      |              | 5<br>5<br>5<br>5 | 20    |
|     | <b>Visualization (20 marks)</b> <ul style="list-style-type: none"> <li>Creates effective and informative visualizations to summarize the data (5m)</li> <li>Uses appropriate chart types and labeling to convey information (5m)</li> <li>Provides clear and accurate data labels, titles, and legends (5m)</li> <li>Demonstrates understanding of data visualization techniques (5m)</li> </ul> |      |              | 5<br>5<br>5<br>5 | 20    |
| 4   | <b>Research Questions (20 marks)</b>   |      |              |                  |       |

|  |   |  |  |   |    |
|--|---|--|--|---|----|
|  | <ul style="list-style-type: none"> <li>Identifies and justifies research questions to address using the dataset (5m)</li> <li>Develops and applies appropriate data mining methods to answer research questions (5m)</li> <li>Evaluates the data mining model to ensure that the model is accurate and reliable (5m)</li> <li>Provides meaningful insights and conclusions from the data analysis (5m)</li> </ul> |  |  | 5 |    |
|  |   |  |  | 5 |    |
|  |   |  |  | 5 |    |
|  |   |  |  | 5 | 20 |
|  | <b>Presentation (20 marks)</b> <ul style="list-style-type: none"> <li>Presents the analysis in a clear, organized, and engaging manner (5m)</li> <li>Uses appropriate language, tone, and style for the intended audience (5m)</li> <li>Follows appropriate formatting and structure for the presentation (5m)</li> <li>Demonstrates good communication skills (5m)</li> </ul>                                    |  |  | 5 |    |
|  |   |  |  | 5 |    |
|  |   |  |  | 5 |    |
|  |   |  |  | 5 | 20 |

# TABLE OF CONTENTS

|   |    |
|---|----|
| TABLE OF CONTENTS .....                                       | IV |
| TABLE OF FIGURES.....   | V  |
| 1. DATA UNDERSTANDING.....                                    | 1  |
| 1.1. IDENTIFY DATASETS SOURCE.....                            | 1  |
| 1.2. SUMMARISE VARIABLES.....                                 | 1  |
| 1.3. POTENTIAL ISSUES AND LIMITATION .....                    | 4  |
| 2. DATA DESCRIPTION .....                                     | 6  |
| 2.1. DESCRIPTIVE STATISTICS AND SUMMARIES OF THE DATASET..... | 6  |
| 2.2 KEY FEATURES, PATTERNS, AND TRENDS IN THE DATA .....      | 8  |
| 3. DATA PREPROCESSING .....                                   | 11 |
| 3.1. DATA CLEANING .....                                      | 11 |
| 3.2. HANDLING ATTRIBUTES .....                                | 12 |
| 3.3. BALANCING DATA.....                                      | 13 |
| 3.4. SCALING .....  | 15 |
| 3.5. FEATURE SELECTION-PCA .....                              | 16 |
| 3.6. TRAINING AND TESTING .....                               | 18 |
| 4. CLASSIFICATIONS.....                                       | 19 |
| 5. PERFORMANCE EVALUATION.....                                | 20 |

## TABLE OF FIGURES

|  |    |
|--|----|
| <b>Table 1</b> Top 5 rows of the dataset .....   | 2  |
| <b>Table 2</b> Descriptive statistic of the data .....   | 7  |
| <b>Figure 2.1</b> Correlations of the data.....  | 8  |
| <b>Figure 2.2</b> Scatterplot of the data .....  | 9  |
| <b>Figure 3.1</b> Summary of number of missing value(s) in the data frame .....                                  | 11 |
| <b>Figure 3.2</b> Summary of number of non-numeric data in the data frame.....                                   | 11 |
| <b>Figure 3.3</b> Summary of number of duplicated data(s) in the data frame .....                                | 11 |
| <b>Figure 3.4</b> Difference between Current Liability to Equity and Current Liabilities/Equity. ..              | 12 |
| <b>Figure 3.5</b> Difference between Current Liability to Liability and Current Liabilities /<br>Liability. .... | 12 |
| <b>Figure 3.6</b> Only a small number of pairs of data that are not similar to each other.....                   | 13 |
| <b>Figure 3.7</b> Average Net Value Per Share is added into the data frame.....                                  | 13 |
| <b>Figure 3.8</b> The imbalanced data between Class 0 and Class 1. ....  | 13 |
| <b>Figure 3.9</b> The balanced data after the process of oversampling.....                                       | 15 |
| <b>Figure 3.10</b> The size of data in Class 1 and Class 0 before and after oversampling. ....                   | 15 |
| <b>Figure 3.11</b> The data set after scaling. ....  | 16 |
| <b>Figure 3.12</b> The PCA curve for ROS and SMOTE Model. ....   | 16 |
| <b>Figure 3.13</b> The influence rate for each attributes. ....  | 17 |
| <b>Figure 3.14</b> The train and test sample size for unbalanced data. ....                                      | 18 |
| <b>Figure 3.15</b> The train and test sample size after oversampling.....  | 18 |
| <b>Figure 5.1</b> Recall rate for survived and bankrupt class in percentage (%) .....                            | 20 |
| <b>Figure 5.2</b> Precision rate for survived and bankrupt class in percentage (%).....                          | 21 |
| <b>Figure 5.3</b> F1-score for survived and bankrupt class in percentage (%).....                                | 21 |
| <b>Figure 5.4</b> Overall performance for all models.....  | 23 |
| <b>Figure 5.5</b> Time taken for all models in milliseconds(ms) .....  | 23 |
| <b>Figure 5.6</b> Overall performance for all models.....  | 24 |

## 1. DATA UNDERSTANDING

### 1.1. Identify datasets source

The dataset that we decided to use is the “Company Bankruptcy Prediction” data from the Taiwan Economic Journal throughout the years 1999-2009. This data is taken from Kaggle which is an online community of data scientists and machine learning. It provides access to a vast collection of publicly available datasets, tools, and computing resources, enabling users to analyse, visualise, and model data to solve real-world problems. Kaggle also hosts a variety of competitions that challenge participants to build the best predictive models for a given problem or dataset. The datasets we took are from a data scientist named Fedesoriano and it was published 2 years ago.

Following are the research questions of this study:

what do these statistic means for each feature?e.g.

1. What is the impact of the unbalanced class distribution on performance?
2. Which machine learning model can give better performance in terms of accuracy and execution time?

### 1.2. Summarise Variables

The dataset contains information on various financial ratios and indicators for companies that have bankrupted or are still surviving. There are 6819 observations in the dataset, with 96 features to identify the bankruptcy status. The class of this dataset is named “Bankrupt?” which is an asymmetric binary class that consists of 0 and 1. 1 will be assigned to this column if the company is bankrupt, which is more important, whereas 0 represents the company that survived from bankruptcy. The remaining 95 features include various financial ratios such as liquidity ratios, solvency ratios, profitability ratios, cash flows, assets and liabilities etc., as well as the industry sector of the company. Due to the large numbers of attributes which are difficult to visualise, 16 attributes are randomly selected for further explanation.

Table 1

*Top 5 rows of the dataset*

|   | Bankrupt? | Non-industry<br>income and<br>expenditure/<br>revenue | Interest-<br>bearing<br>debt<br>interest<br>rate | Net<br>Value<br>Per<br>Share<br>(A) | Persistent<br>EPS in the<br>Last Four<br>Seasons | Net<br>Value<br>Growth<br>Rate | Interest<br>Expense<br>Ratio | Total<br>debt/Total<br>net worth | Borrowing<br>dependency | Net profit<br>before<br>tax/Paid-<br>in capital | Fixed Assets<br>Turnover<br>Frequency | Cash/Total<br>Assets | Net<br>Income<br>to Total<br>Assets | Net Income to<br>Stockholder's<br>Equity | Degree of<br>Financial<br>Leverage<br>(DFL) | Equity<br>to<br>Liability |
|---|-----------|---|--|-------------------------------------|--|--------------------------------|------------------------------|----------------------------------|-------------------------|---|---------------------------------------|----------------------|-------------------------------------|--|---|---------------------------|
| 0 | 1         | 0.302646  | 0.000725   | 0.147950                            | 0.169141   | 0.000327                       | 0.629951                     | 0.021266                         | 0.390284                | 0.137757  | 1.165010e-04                          | 0.004094             | 0.716845                            | 0.827890                                 | 0.026601                                    | 0.016469                  |
| 1 | 1         | 0.303556  | 0.000647   | 0.182251                            | 0.208944   | 0.000443                       | 0.635172                     | 0.012502                         | 0.376760                | 0.168962  | 7.190000e+08                          | 0.014948             | 0.795297                            | 0.839969                                 | 0.264577                                    | 0.020794                  |
| 2 | 1         | 0.302035  | 0.000790   | 0.177911                            | 0.180581   | 0.000396                       | 0.629631                     | 0.021248                         | 0.379093                | 0.148036  | 2.650000e+09                          | 0.000991             | 0.774670                            | 0.836774                                 | 0.026555                                    | 0.016474                  |
| 3 | 1         | 0.303350  | 0.000449   | 0.154187                            | 0.193722   | 0.000382                       | 0.630228                     | 0.009572                         | 0.379743                | 0.147561  | 9.150000e+09                          | 0.018851             | 0.739555                            | 0.834697                                 | 0.026697                                    | 0.023982                  |
| 4 | 1         | 0.303475  | 0.000686   | 0.167502                            | 0.212537   | 0.000439                       | 0.636055                     | 0.005150                         | 0.375025                | 0.167461  | 2.935210e-04                          | 0.014161             | 0.795016                            | 0.839973                                 | 0.024752                                    | 0.035490                  |

*Note.* Table 1 contains the first five rows with 16 features extracted from the dataset, where 15 is numerical features and a class label.

The features consist of:

**Bankrupt?:** A binary feature that indicates whether a company has been declared bankrupt (1) or not (0), the target variable for classification.

**Non-industry income and expenditure/revenue:** This attribute describes the net income obtained from non-operating activities.

**Interest-bearing debt interest rate:** It describes the amount of the companies' outstanding in debt.

**Net Value Per Share (A):** This indicator calculates the share of net value by dividing the share value to the outstanding shares of a company.

**Persistent earnings per share (EPS) in the last four seasons:** It is a ratio to indicate the earnings obtained for per share (EPS).

**Net Value Growth Rate:** These attributes describe the company's growth potential in quantitative value.

**Interest Expense Ratio:** This is a ratio which examines how much interest is charged for every single dollar earned.

**Total debt/Total net worth:** This ratio examines how much debt is earned by every single net worth value.

**Borrowing dependency:** It provides an indication of a company's dependency on borrowing to finance its operations.

**Net profit before tax/Paid-in capital:** It shows the net profit/ capital before taxation.

**Fixed Assets Turnover Frequency:** Indication of a company's efficiency in using its fixed assets.

**Cash/Total Assets:** It indicates the company's liquidity.

**Net Income to Total Assets:** This feature represents a company's net income divided by its total assets. It provides an indication of a company's profitability relative to its total assets.



**Net Income to Stockholder's Equity:** This feature represents a company's net income divided by its stockholder's equity which is named as 'Return on equity(ROE)'.

**Degree of Financial Leverage (DFL):** It shows the company's sensitivity to changes in its financial structure.

**Equity to Liability:** This feature represents a company's equity divided by its liabilities. ✓

Throughout the investigation, we noticed that some of the attributes might be duplicates and have diminished involvement in the preprocessing of the data. Therefore, these attributes will be removed in the data preprocessing phase, which will be discussed in detail in that section. From the perspective of analysis of data patterns and trends, we know that by analysing the dataset, one could identify patterns and trends that may be relevant to predicting bankruptcy. For example, certain financial ratios, such as the quick ratio or the debt ratio, may be more predictive of bankruptcy than others. Lastly, the awareness of the data context. The dataset is relevant to the larger context of predicting bankruptcy in companies. It can be used to develop models to predict the likelihood of bankruptcy based on financial and non-financial factors.

### 1.3. Potential issues and limitation

This large dataset might contain some issues and limitations that may affect the model's performance. Increasing the number of attributes is often perceived to obtain better accuracy as more information is given. However, a large number of attributes in the dataset can lead to the "curse of dimensionality", where the accuracy may be reduced if the number of features exceeds a certain threshold. Making it difficult to analyse and interpret the data, especially when dealing with dependency variables. Furthermore, processing a large dataset is computationally complex, making the computational cost expensive. As it requires high-performance computing resources, and is time-consuming to analyse and model the data effectively.

Subsequently, this dataset consists of redundant and /or irrelevant values, which will affect the quality of the dataset, leading to noise in the data and affecting the accuracy of the results. Moreover, this dataset has an unbalanced class distribution of 220 bankrupted companies and 6599 surviving companies. This phenomenon may result in model overfitting to the majority class (surviving company), yet the minority class (bankruptcy company) is more

significant for investigation. The model may overfit to the least important class rather than the underlying patterns of the important information, leading to a biased result. Lastly, it can be challenging to visualise and communicate the data effectively with a large number of attributes, making it harder to convey insights and findings to stakeholders.



## **2. DATA DESCRIPTION**

In this section, the descriptive statistics, key features, patterns, and trends of the data will be analysed and summarised using statistical and visual methods.

### **2.1. Descriptive statistics and summaries of the dataset**

Table 2 below summarised the descriptive analysis of the dataset. The mean of the "Bankrupt?" column is 0.032263, which means only about 3.2263% of the company is bankrupt, indicating the unbalanced distribution of the dataset. By analysing the standard deviation of each feature, it is obvious that Interest-bearing debt interest rate, Net Value Growth Rate, Total debt/Total net worth, and Fixed Assets Turnover Frequency have extremely high standard deviation. This means that these 4 features have high variability and volatility in the dataset, causing inconsistency of the data set and difficult to make predictions. While the standard deviation of some other features is less than 0.1, demonstrating a less dispersed and lower error from the mean, making a good prediction data.

Table 2

Descriptive statistic of the data

what do these statistic means for each feature?  
e.g. net profit before tax, you can see the mean based on bankrupt?

|       | Bankrupt? | Non-industry<br>income and<br>expenditure/<br>revenue | Interest-bearing<br>debt interest<br>rate | Net Value<br>Per Share<br>(A) | Persistent<br>EPS in the<br>Last Four<br>Seasons | Net Value<br>Growth Rate | Interest<br>Expense<br>Ratio | Total debt/<br>Total net<br>worth | Borrowing<br>dependency | Net profit<br>before tax/<br>Paid-in<br>capital | Fixed<br>Assets<br>Turnover<br>Frequency | Cash/ Total<br>Assets | Net Income to<br>Total Assets | Net Income to<br><del>Stockholder's</del><br>Equity | Degree of<br>Financial<br>Leverage<br>(DFL) | Equity to<br>Liability |
|-------|-----------|---|---|-------------------------------|--|--------------------------|------------------------------|-----------------------------------|-------------------------|---|--|-----------------------|-------------------------------|---|---|------------------------|
| count | 6819.00   | 6819.000000   | 6.819000e+03                              | 6819.000000                   | 6819.00000                                       | 6.819000e+03             | 6819.0000                    | 6.819000e+03                      | 6819.000000             | 6819.000000                                     | 6.819000e+03                             | 6819.000000           | 6819.000000                   | 6819.000000   | 6819.0000                                   | 6819.00                |
| mean  | 0.032263  | 0.303623  | 1.644801e+07                              | 0.190633                      | 0.228813   | 1.566212e+06             | 0.630991                     | 4.416337e+06                      | 0.374654                | 0.182715  | 1.008596e+09                             | 0.124095              | 0.807760                      | 0.840402  | 0.027541                                    | 0.047578               |
| std   | 0.176710  | 0.011163  | 1.082750e+08                              | 0.033474                      | 0.033263   | 1.141594e+08             | 0.011238                     | 1.684069e+08                      | 0.016286                | 0.030785  | 2.477557e+09                             | 0.139251              | 0.040332                      | 0.014523  | 0.015668                                    | 0.050014               |
| min   | 0.000000  | 0.000000  | 0.000000e+00                              | 0.000000                      | 0.000000   | 0.000000e+00             | 0.000000                     | 0.000000e+00                      | 0.000000                | 0.000000  | 0.000000e+00                             | 0.000000              | 0.000000                      | 0.000000  | 0.000000                                    | 0.000000               |
| 25%   | 0.000000  | 0.303466  | 2.030200e-04                              | 0.173613                      | 0.214711   | 4.409690e-04             | 0.630612                     | 3.007049e-03                      | 0.370168                | 0.169376  | 2.330010e-04                             | 0.033543              | 0.796750                      | 0.840115  | 0.026791                                    | 0.024477               |
| 50%   | 0.000000  | 0.303525  | 3.210320e-04                              | 0.184400                      | 0.224544   | 4.619560e-04             | 0.630698                     | 5.546284e-03                      | 0.372624                | 0.178456  | 5.930940e-04                             | 0.074887              | 0.810619                      | 0.841179  | 0.026808                                    | 0.033798               |
| 75%   | 0.000000  | 0.303585  | 5.325530e-04                              | 0.199570                      | 0.238820   | 4.993620e-04             | 0.631125                     | 9.273292e-03                      | 0.376271                | 0.191607  | 3.652371e-03                             | 0.161073              | 0.826455                      | 0.842357  | 0.026913                                    | 0.052838               |
| max   | 1.000000  | 1.000000  | 9.900000e+08                              | 1.000000                      | 1.000000   | 9.330000e+09             | 1.000000                     | 9.940000e+09                      | 1.000000                | 1.000000  | 9.990000e+09                             | 1.000000              | 1.000000                      | 1.000000  | 1.000000                                    | 1.000000               |

features

Note. Descriptive analysis on all 16 of the dataset

## 2.2 Key features, patterns, and trends in the data

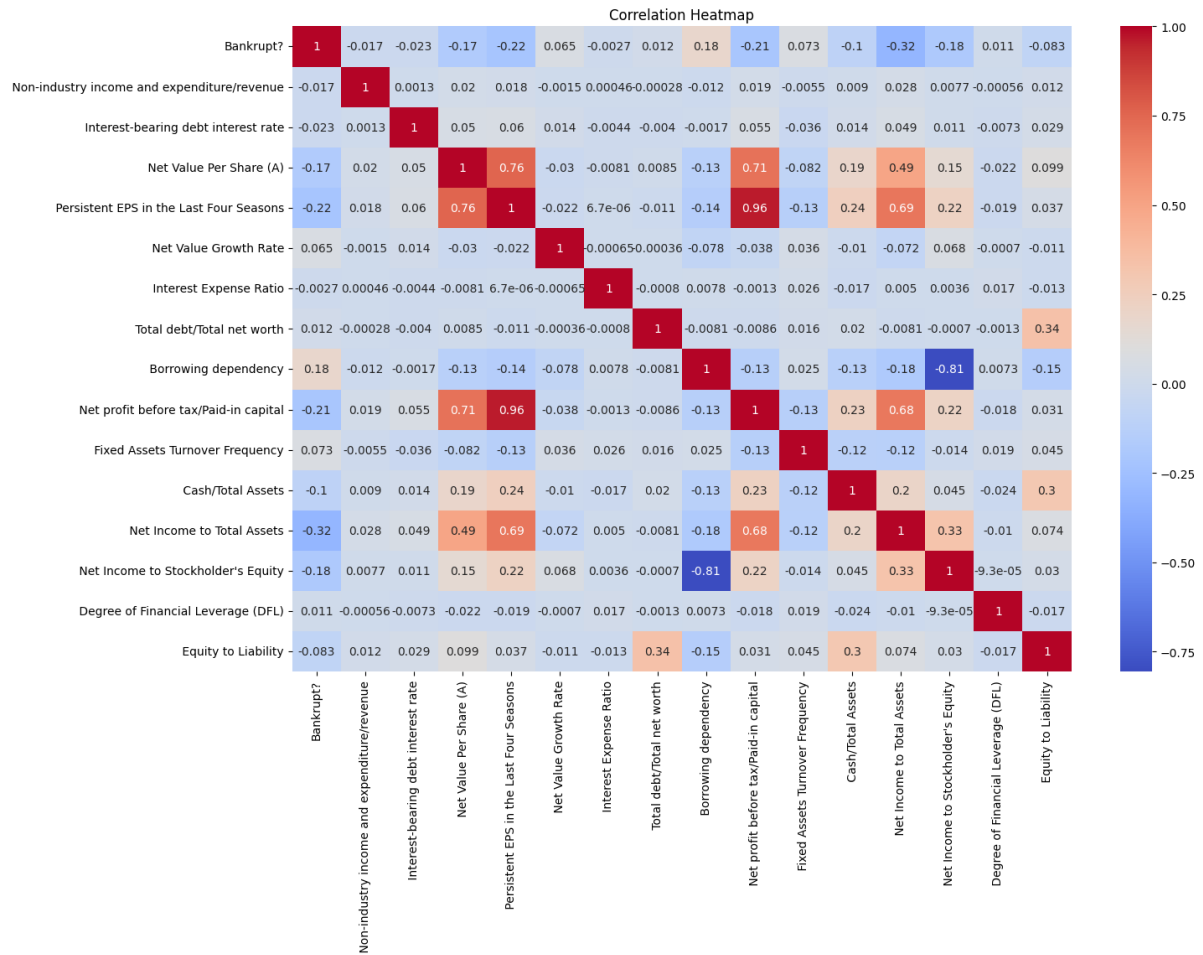


Figure 2.1 Correlations of the data

The "Bankrupt?" column has a negative correlation with most randomly selected features, showing that bankrupt companies tend to have lower correlation when measured to these features. Among these 15 features, "Net income to total assets" has the highest negative correlation to the "Bankrupt?" column. This indicates a significant influence on the results where this feature could significantly affect a company's bankruptcy. Since companies facing financial difficulties are usually experiencing lower revenue, more debt, and less profitability, it is totally understandable from this perspective.

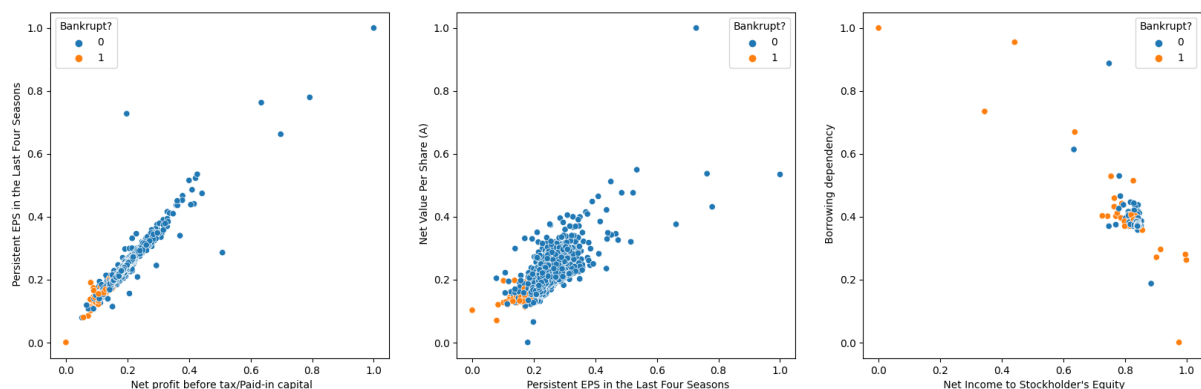
Meanwhile, "Borrowing dependency" and "Fixed assets turnover frequency" are both highly positively correlated to the "Bankrupt?" column as well as to each other. Because of the negative correlation of -0.13 between borrowing dependency and cash/total assets can explain this situation. If the borrowing dependency is high, it might because of a lack of cash flow. As a result, some companies may sell their fixed assets in order to maintain or increase their

liquidity. However, the 0.073 correlation to bankruptcy shows that this might not be the right practice as some of them were bankrupted eventually.

Furthermore, the "Non-industry income and expenditures and revenue" column has a positive correlation with the "Interest Expense Ratio" and "Total debt/Total net worth" columns, which means that companies with more non-industry revenues and incomes are more likely to have higher debt and interest expenses.

Besides, the "Net Value Per Share (A)" column has a positive correlation with the "Net Income to Total Assets" and "Net Income to Stockholder's Equity" columns. This proves that stakeholders are more confident of a company when it has a higher net value per share because of its higher profitability. Also, the "Interest-bearing debt interest rate" column has a positive correlation with the "Total debt/Total net worth" column, which indicates that debt levels are higher for companies with higher interest-bearing debt rates.

With the correlations above, we can identify some patterns and trends of the data.



**Figure 2.2** Scatterplot of the data

The first scatterplot shows the relationship between "Persistent EPS in the Last Four Seasons" and "Net Value Per Share (A)", and the second scatterplot shows the relationship between "Net profit before tax/Paid-in capital" and "Persistent EPS in the Last Four Seasons". As indicated in the legend, the hue parameter colored in orange and blue dots represents bankruptcy (Class 1) and surviving (Class 0) respectively. These two plots illustrate how bankrupted companies have lower values for the three attributes mentioned above. Earnings per share of a company depend on the net profit. Financial difficulties may arise for a low-net-income company in the future, resulting in insolvency, cost cutting, lower. In such a situation, they should issue additional shares rather than paying dividends to their stakeholder. Since the

✓ good

extra shares liquidate the share value, this reduces the earning per share and share value as well as the interest of shareholders and investors. Eventually, the company may find it hard to get funding, which decreases its liquidity and profitability. Due to this, most companies experience financial difficulties until they declare bankruptcy without a proper resolution measure. Even companies with lower values of these attributes can still survive. However, the reversed statement is not necessarily true. These companies might have implemented different executive standards according to their situation to avoid bankruptcy. Thus, non-bankrupt companies have a wider range of values for these variables.

The third scatterplot shows a negative correlation between "Net Income to Stockholders Equity" (ROE) and "Borrowing dependency". Class 0 is mostly gathered around the coordinate (x-axis=0.8, y-axis=0.4). However, there are many Class 1 surroundings the coordinate as well. Therefore, we cannot prove that this is the optimal combination. Despite that, we can still spot some patterns here where all Class 0's ROE fall between 0.6-0.9. Theoretically, if the borrowing dependency is too high or too low, a company might have a hidden issue or underinvesting matter which might cause them to increase their risk to bankruptcy. However, a company's ability to repay its debts is also important. Using equity to finance a company's growth, ROE measures the effectiveness of that company's growth. In order to make more profit from this, the company needs to have an increasing or high ROE, which indicates that they are able to reinvest their earnings wisely. While a decreasing or low ROE means that company is making poor strategies in their financing and investing strategies. This explained that if the company's ROE below 0.6 and are all from Class 1, they might not able to pay back their debt due to their poor management.

In conclusion, these scatterplots provide a visual representation of the correlations, patterns, and trends between specific pairs of variables in the dataset, and how these correlations relate to bankruptcy status. The scatterplots suggest that bankrupt companies tend to have lower values for certain financial metrics, such as "Persistent EPS in the Last Four Seasons", "Net Value Per Share (A)", and "Net profit before tax/Paid-in capital", and higher values for others, such as "Borrowing dependency" and "Net Income to Stockholder's Equity". These visualizations can provide useful insights for further analysis and modelling.

✓ very good insights!

### 3. DATA PREPROCESSING

#### 3.1. Data Cleaning

```
Check for missing value(s):
Bankrupt? 0
ROA(C) before interest and depreciation before interest 0
ROA(A) before interest and % after tax 0
ROA(B) before interest and depreciation after tax 0
Operating Gross Margin 0
..
Liability to Equity 0
Degree of Financial Leverage (DFL) 0
Interest Coverage Ratio (Interest expense to EBIT) 0
Net Income Flag 0
Equity to Liability 0
Length: 96, dtype: int64
```

*Figure 3.1 Summary of number of missing value(s) in the data frame*

```
Check for non-numeric data:
[]
```

*Figure 3.2 Summary of number of non-numeric data in the data frame*

In this stage, we will check if there is any missing, duplicate or noise in the dataset. Figure 3.1 and 3.2 shows that this dataset has no missing values or non-numeric values (NaN), so we do not need to handle missing values. When we check for data duplication, there is no data redundancy in this dataset as shown in Figure 3.3. Eventually, we also found that every data is reasonable and relevant to their features, so there is no noisy data in the dataset.

```
#check for any duplicate data
print("Number of duplicated data:",file.duplicated().sum())
Number of duplicated data: 0
```

*Figure 3.3 Summary of number of duplicated data(s) in the data frame*



### 3.2. Handling Attributes

```
Difference between Current Liabilities/Equity and Current Liability to Equity:
0      0.0
1      0.0
2      0.0
3      0.0
4      0.0
...
6814   0.0
6815   0.0
6816   0.0
6817   0.0
6818   0.0
Length: 6819, dtype: float64
```

The total of difference between Current Liabilities/Equity and Current Liability to Equity is: 0.0

**Figure 3.4** Difference between Current Liability to Equity and Current Liabilities/Equity.

```
Difference between Current Liabilities/Liability and Current Liability to Liability:
0      0.0
1      0.0
2      0.0
3      0.0
4      0.0
...
6814   0.0
6815   0.0
6816   0.0
6817   0.0
6818   0.0
Length: 6819, dtype: float64
```

The total of difference between Current Liabilities/Liability and Current Liability to Liability is: 0.0

**Figure 3.5** Difference between Current Liability to Liability and Current Liabilities / Liability.

When the data frame is checked, we found that some attributes are redundant and proven that they are similar, as shown in Figures 3.4 and 3.5. For example, *Current Liability to Liability* is identical to *Current Liability/Liability*. Thus, it is reasonable to remove these two attributes from the data frame. So, one of these attributes was removed, the same was for other duplicated attributes.



The attributes *Net Value Per Share (A)*, *Net Value Per Share (B)*, and *Net Value Per Share (C)* are removed because the values are almost similar to each other. This can be seen in Figure 3.6 below. However, since this attribute is important to the prediction of bankruptcy, we add a new attribute which is named as *Average of Net Value Per Share*. This new attribute reflects the average of three different net values per share. Figure 3.7 shows the new features in the data frame.



There are 60 pair(s) from A and B that are different.  
 There are 0 pair(s) from B and C that are different.  
 There are 78 pair(s) from A and C that are different.

**Figure 3.6** Only a small number of pairs of data that are not similar to each other.

```

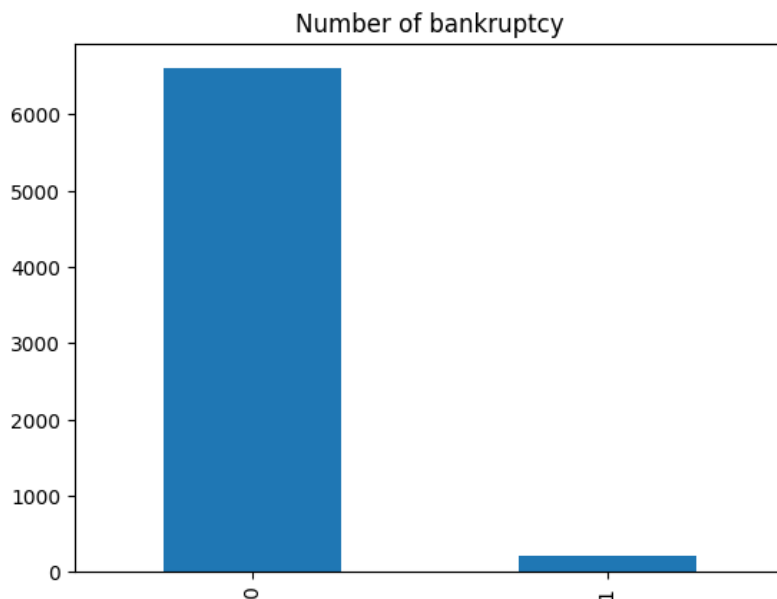
85 Liability to Equity 6819 non-null float64
86 Degree of Financial Leverage (DFL) 6819 non-null float64
87 Interest Coverage Ratio (Interest expense to EBIT) 6819 non-null float64
88 Net Income Flag 6819 non-null int64
89 Equity to Liability 6819 non-null float64
90 Average Net Value Per Share 6819 non-null float64
dtypes: float64(89), int64(2)
memory usage: 4.7 MB

```

**Figure 3.7** Average Net Value Per Share is added into the data frame.

### 3.3. Balancing Data

In the data frame, there are 6819 sets of data, where the number of bankruptcies is 220, accounting for only 3.22% of the overall distributions. This number is very low if we compare it with the number of non-bankruptcies, which is 6599. Figure 3.8 below is a bar graph that shows the uneven distribution between these two classes. The presence of this imbalanced data can cause the model to be biased towards non-bankruptcy. As a result, if the user wanted to use this model to predict bankruptcy, the tendency for the user to obtain 0, which represents non-bankruptcy. The performance of this model will be bad if this problem is not resolved.



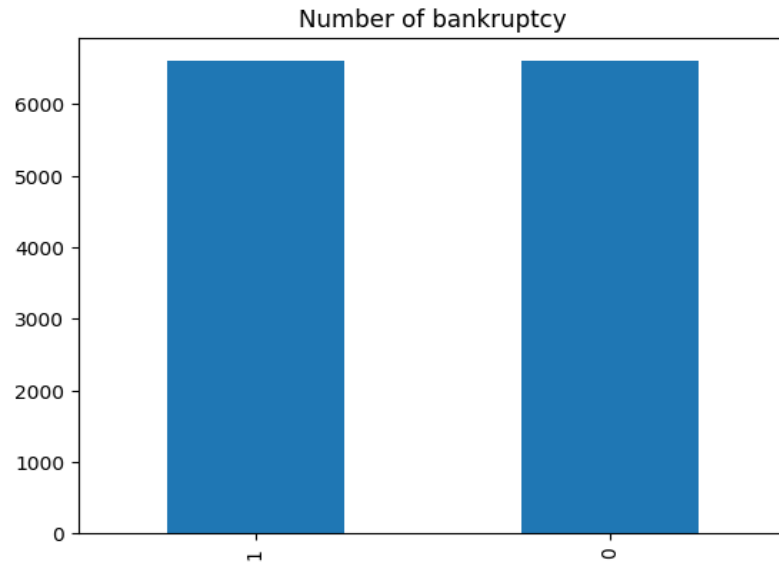
**Figure 3.8** The imbalanced data between Class 0 and Class 1.

In order to balance out the classes, we can remove outliers and then either undersample Class 0 or oversample Class 1, or both. However, there are some concerns about deleting the

outliers. The model will not be affected much if and only if the number of outliers is small and the outliers are not important. As a consequence, the amount of data in Class 1 will decrease if some data under the class are outliers, even though the number is critical. As a result, the tendency of this model becomes more biased towards Class 0. Apart from that, the outliers in this data frame are important because we want to know the cause of a company's bankruptcy. For example, a high debt ratio can lead to a company's bankruptcy. If this outlier is removed, we will lose important data that has a large effect on this model. Thus, we decide to keep the outliers in the dataset.

Undersampling is a technique to balance the class distribution by reducing data from Class 0. However, the flaw in this technique is that the deleted data may be useful to the model. In order to overcome this limitation, heuristics must be used together during the undersampling process. Thus, the prediction process will be less accurate if we use non-heuristic decisions. Not only that, if the undersampling process is executed, the number of class 0 will drop to the critical level.

Hence, we do a shift of perspective towards the method of oversampling Class 1. Random oversampling (ROS) is widely used for oversampling data due to its simplicity. This technique selects data from the minority class (Class 1) and replicates it randomly until the data is balanced as shown in Figure 3.9. However, classes are difficult to generalize as most data is duplicated, and thus, data may overfit. Therefore, the Synthetic Minority Oversampling technique (SMOTE) is suggested to solve the overfitting issue by generating synthetic data. This technique uses KNN to find the k-nearest data from the same classes and generate points on the linear distance between them. Despite that, we decided to use both methods in our study to find out which technique performs better. Eventually, both datasets have 6599 values for each class, making a total of 13198 rows, as shown in Figure 3.10.



**Figure 3.9** The balanced data after the process of oversampling.

```
BEFORE OVERSAMPLING:
Number of bankruptcy: 220
Number of non-bankruptcy: 6599

AFTER OVERSAMPLING:
Number of bankruptcy: 6599
Number of non-bankruptcy: 6599
```

**Figure 3.10** The size of data in Class 1 and Class 0 before and after oversampling.

### 3.4. Scaling

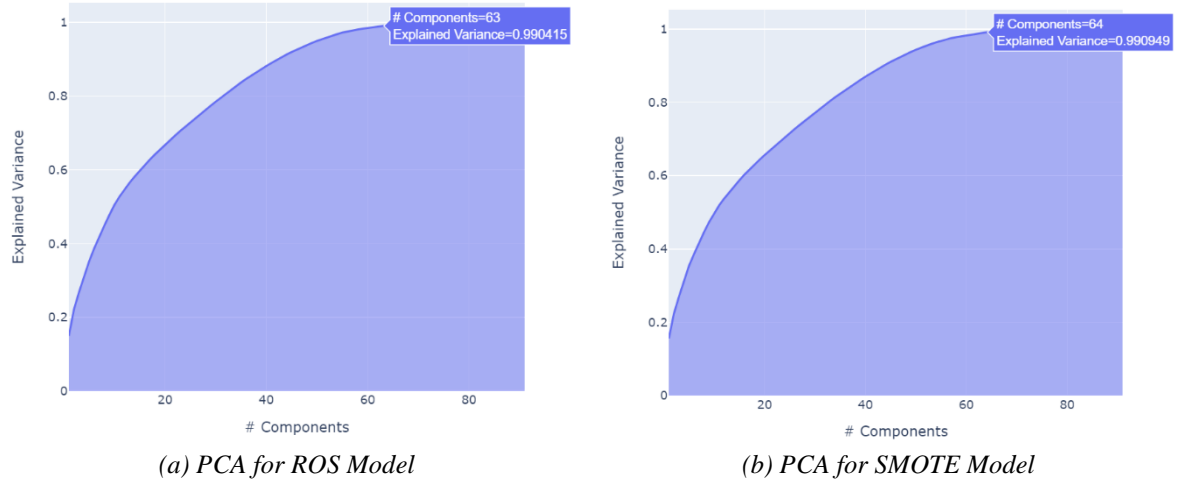
Following that, we standardize the data by scaling them using Standard Scaler. The purpose of this is to ensure that all data can be explained in a comparable ratio, and treated equally. The large-scale value may hinder the visualization when compared to the small-scale value. Therefore, the underlying patterns might be difficult to capture. Moreover, the large value may indicate that the attribute is more important than others with a smaller value. Hence, scaling the data is one of the important steps during preprocessing before the data is modelled. Instead of Min Max Scaler, Standard Scaler is used for this data frame. This type of data frame is unsuitable for us to use Min Max Scaler because we would want all values between the minimum and maximum to be scaled. Furthermore, this data frame follows Gaussian distribution. Thus, Standard Scaler is more suitable to be used to scale this data set. The result can be seen in Figure 3.11 below.

|       | ROA(C)<br>before<br>interest and<br>depreciation<br>before<br>interest | ROA(A)<br>before<br>interest<br>and %<br>after tax | ROA(B)<br>before<br>interest and<br>depreciation<br>after tax | Operating<br>Gross<br>Margin | Realized<br>Sales<br>Gross<br>Margin | Operating<br>Profit<br>Rate | Pre-tax<br>net<br>Interest<br>Rate | After-tax<br>net<br>Interest<br>Rate |
|-------|--|--|---|------------------------------|--------------------------------------|-----------------------------|------------------------------------|--------------------------------------|
| 0     | 0.370594   | 0.424389   | 0.405750  | 0.601457                     | 0.601457                             | 0.998969                    | 0.796887                           | 0.808809                             |
| 1     | 0.464291   | 0.538214   | 0.516730  | 0.610235                     | 0.610235                             | 0.998946                    | 0.797380                           | 0.809301                             |
| 2     | 0.426071   | 0.499019   | 0.472295  | 0.601450                     | 0.601364                             | 0.998857                    | 0.796403                           | 0.808388                             |
| 3     | 0.399844   | 0.451265   | 0.457733  | 0.583541                     | 0.583541                             | 0.998700                    | 0.796967                           | 0.808966                             |
| 4     | 0.465022   | 0.538432   | 0.522298  | 0.598783                     | 0.598783                             | 0.998973                    | 0.797366                           | 0.809304                             |
| ...   | ...  | ...  | ...   | ...                          | ...                                  | ...                         | ...                                | ...                                  |
| 13193 | 0.487886   | 0.545410   | 0.540554  | 0.613190                     | 0.613190                             | 0.999089                    | 0.797442                           | 0.809352                             |
| 13194 | 0.430556   | 0.470508   | 0.472027  | 0.595267                     | 0.595267                             | 0.998923                    | 0.797263                           | 0.809196                             |
| 13195 | 0.495539   | 0.545901   | 0.547674  | 0.619770                     | 0.619770                             | 0.999041                    | 0.797473                           | 0.809367                             |
| 13196 | 0.476820   | 0.522078   | 0.530489  | 0.606927                     | 0.606927                             | 0.999011                    | 0.797354                           | 0.809282                             |
| 13197 | 0.356018   | 0.403184   | 0.385674  | 0.601364                     | 0.601364                             | 0.998859                    | 0.797020                           | 0.808933                             |

13198 rows × 91 columns

*Figure 3.11 The data set after scaling.*

### 3.5. Feature selection-PCA



*Figure 3.12 The PCA curve for ROS and SMOTE Model.*

As mentioned in section 1.3, large numbers of attributes might cause a curse of dimensionality. Thus, principal component analysis (PCA) feature selection is used to reduce the dimensionality of the dataset and find the optimal number of input attributes. We applied this method on oversampling models only but not for the unbalanced dataset to investigate how the curse of dimensionality affects performance. From Figure 3.12, we know the optimal number of attributes is around 63. The number above them did not have a significant impact on the data and the numbers below them still have room for improvement. Thus, we choose the first 63

attributes for the ROS model and 64 attributes for the SMOTE model as they explained 99% of the data.

On the other hand, figure 3.13 below shows which are the exact influential attributes. As we can see, the “ROA(C) before interest and depreciation before tax” on the left can explain around 15% of the data, which is the highest. For the oversampling data set, we retain a 63/64 attribute count from the left for modelling.

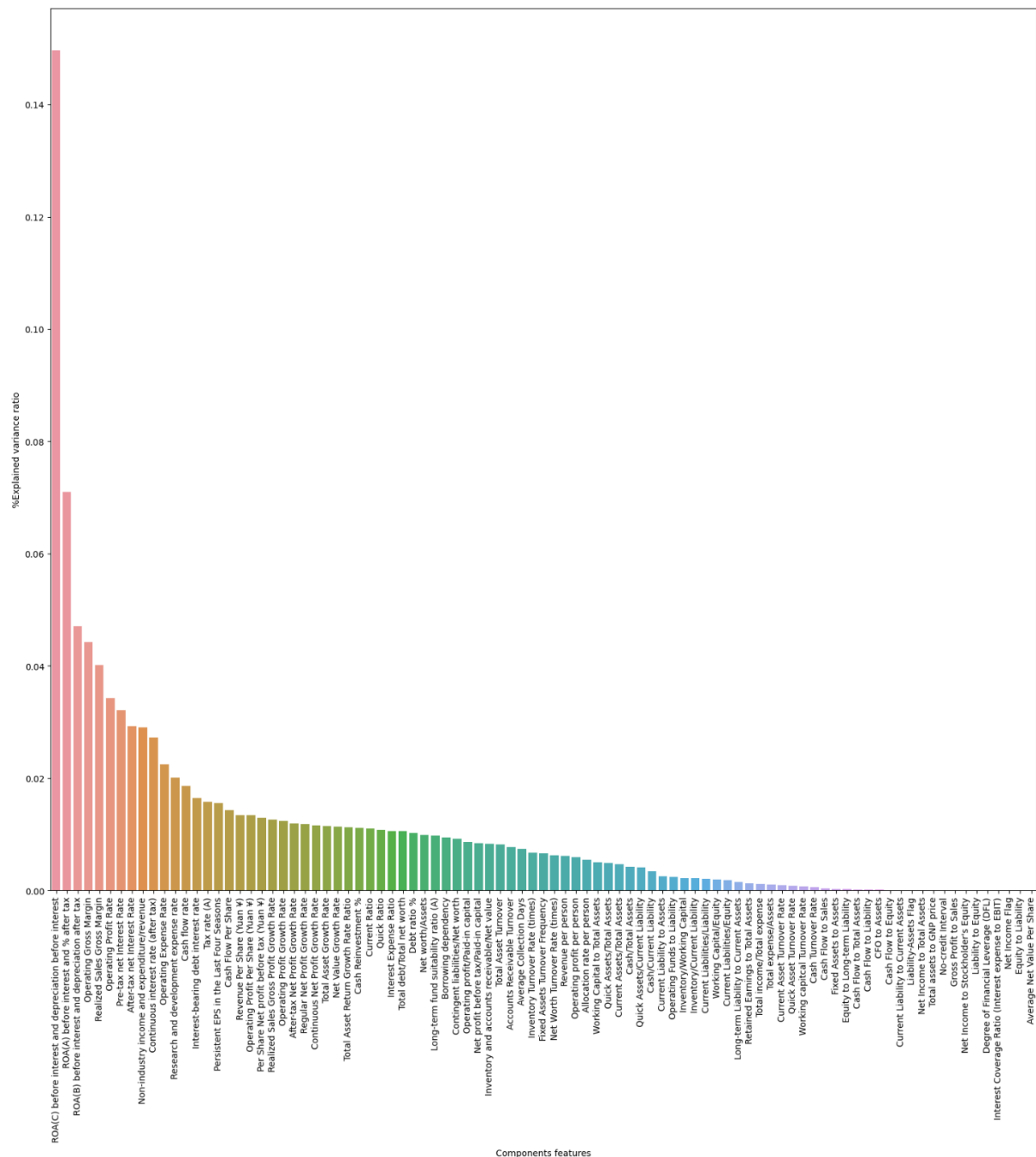


Figure 3.13 The influence rate for each attributes.

good

### 3.6. Training and Testing

Dataset is split into a training and testing set. “train\_test\_split” function is used from the import of the sci-kit-learn library. This process is used to evaluate the performance of the model with a different data instance and helps to avoid overfitting issues. In order to ensure that our model has enough data, we set the percentage of the testing set to 70%. The random state is set to 20 to control the randomness. As the performance of the model will slightly differ with each training, we try to repeat each evaluation with the same sequence of random numbers. Figure 3.15 shows the sample size of the training set and testing set after oversampling.

```
Train sample size (4773, 91)
Test sample size (2046, 91)
```

*Figure 3.14 The train and test sample size for unbalanced data.*

```
Train sample size (9238, 64)
Test sample size (3960, 64)
```

*Figure 3.15 The train and test sample size after oversampling.*

#### 4. CLASSIFICATIONS

In this research, we decided to implement three algorithms for modelling and evaluate their performances. Those are Gaussian Naïve Bayes, K-Nearest Neighbours (KNN), and Decision Tree imported from “scikit.learn” library. Decision trees are more resistant to the noise of the dataset as it is a tree-based algorithm. It requires less effort for data preprocessing, yet it is more time-consuming. While Gaussian Naïve Bayes classifier is because it is highly predictive and less time-consuming. However, it assumes the features are independent which is not practical in the real-world application. KNN makes no assumptions on the attributes by predicting the data in the features space similarities. This instance-based learning method can evolve constantly whenever there is new training data. Nevertheless, the large numbers of attributes and unbalanced distribution data will have a greater impact on the performance. To optimize the model, we perform hyperparameter tuning using “GridSearchCV” in five-cross-fold validation for the model. After the tuning, the best estimator will be used to train the model, and performance evaluation will be done using a test set. The following are the hyperparameters:

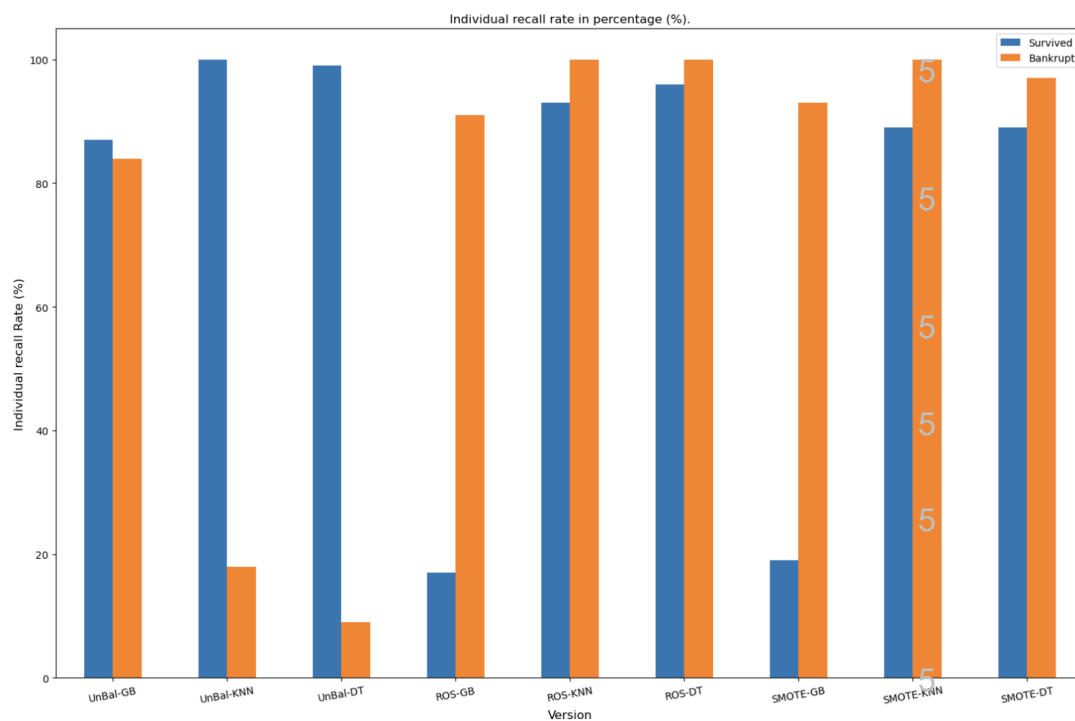
| Classifier                    | Hyperparameters  | Value                               |
|-------------------------------|------------------|-------------------------------------|
| Decision Tree                 | Max_depth        | 2,3,5,10,20                         |
|                               | Min_samples_leaf | 5,10,20,50,100                      |
|                               | criterion        | Gini, entropy                       |
| K-Nearest Neighbours<br>(KNN) | N_neighbors      | 5,7,9,11,13,15                      |
|                               | Weights          | Uniform, distance                   |
|                               | metric           | Minkowski, Euclidean,<br>Manhattan. |



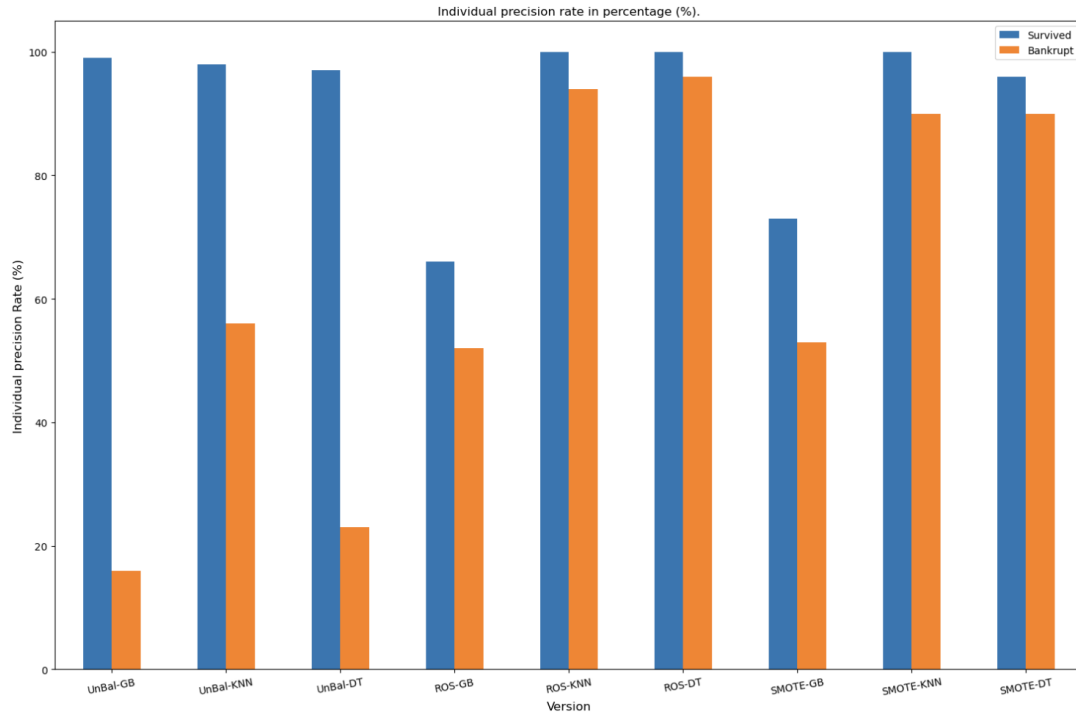


## 5. PERFORMANCE EVALUATION

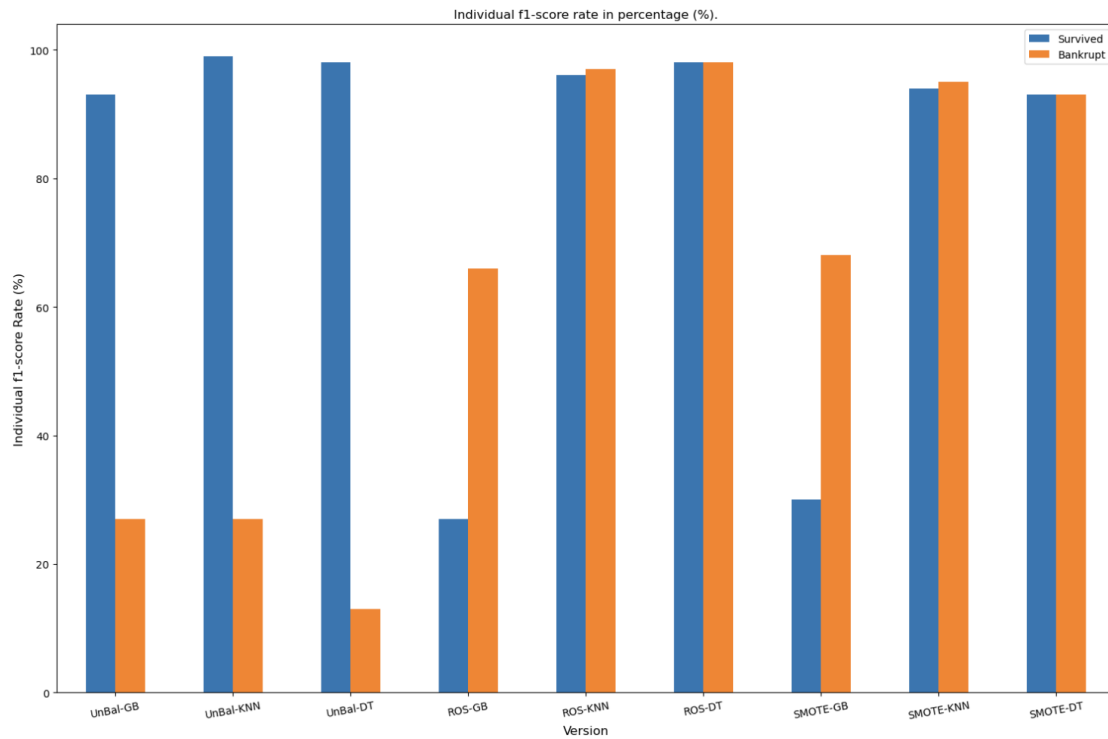
For all figures in section 5, the first three columns on the left represent unbalanced data modelled with three different machine learning models while the rest is the oversampling model. Figure 5.1 shows the recall rate of each model describing how well the model may detect the positive value correctly and therefore, it is a fundamental measure for the asymmetric binary class. While the precision rate in Figure 5.2 indicates the possibility of the predicted positive value is true.



**Figure 5.1** Recall rate for survived and bankrupt class in percentage (%)



**Figure 5.2** Precision rate for survived and bankrupt class in percentage (%)



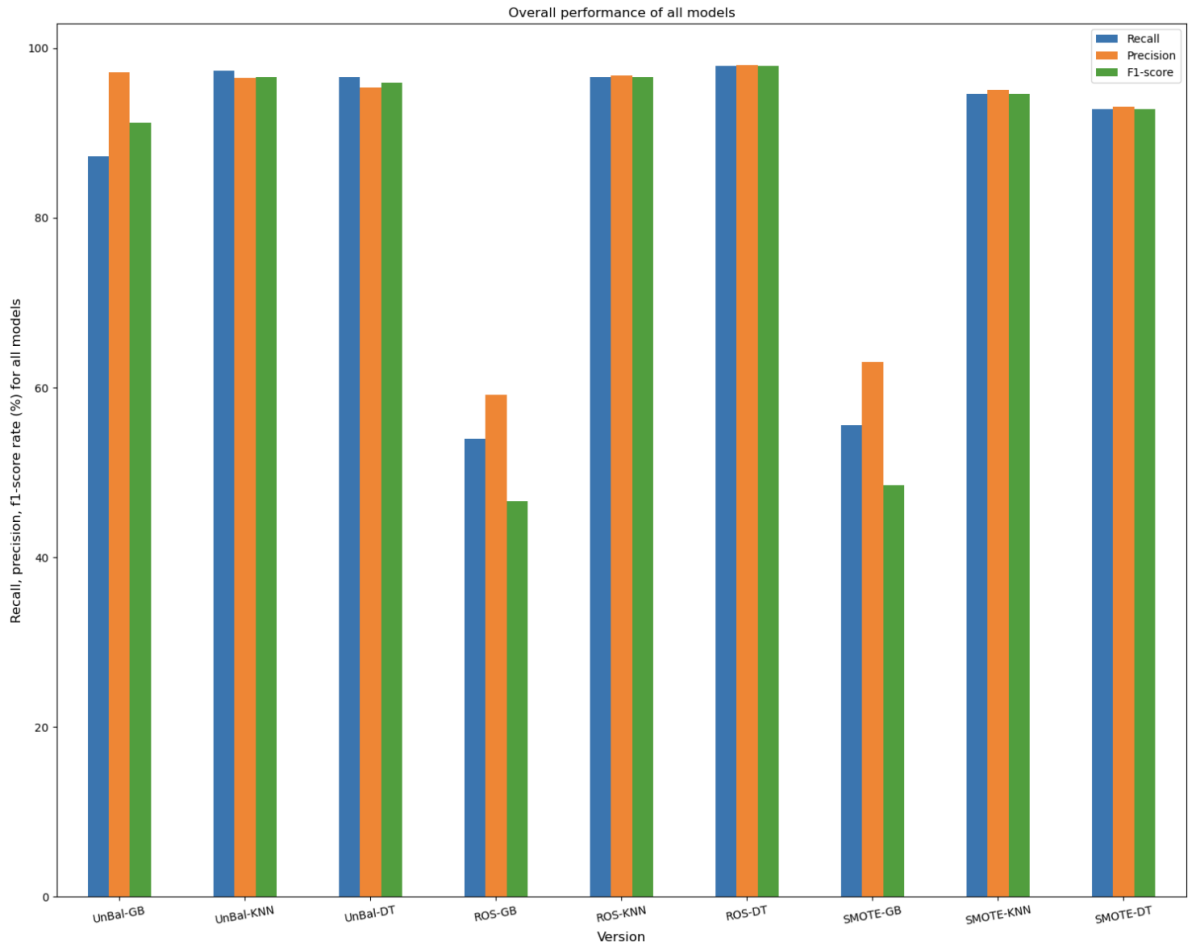
**Figure 5.3** F1-score for survived and bankrupt class in percentage (%)

In the unbalanced model, we can see that Gaussian Naïve Bayes (GB) scores higher recall rate (>80%) in detecting “Bankrupt” class (Class 1) comparing to other models, showing a higher resistance to the unbalanced class distribution issue (UnBal). However, the lowest

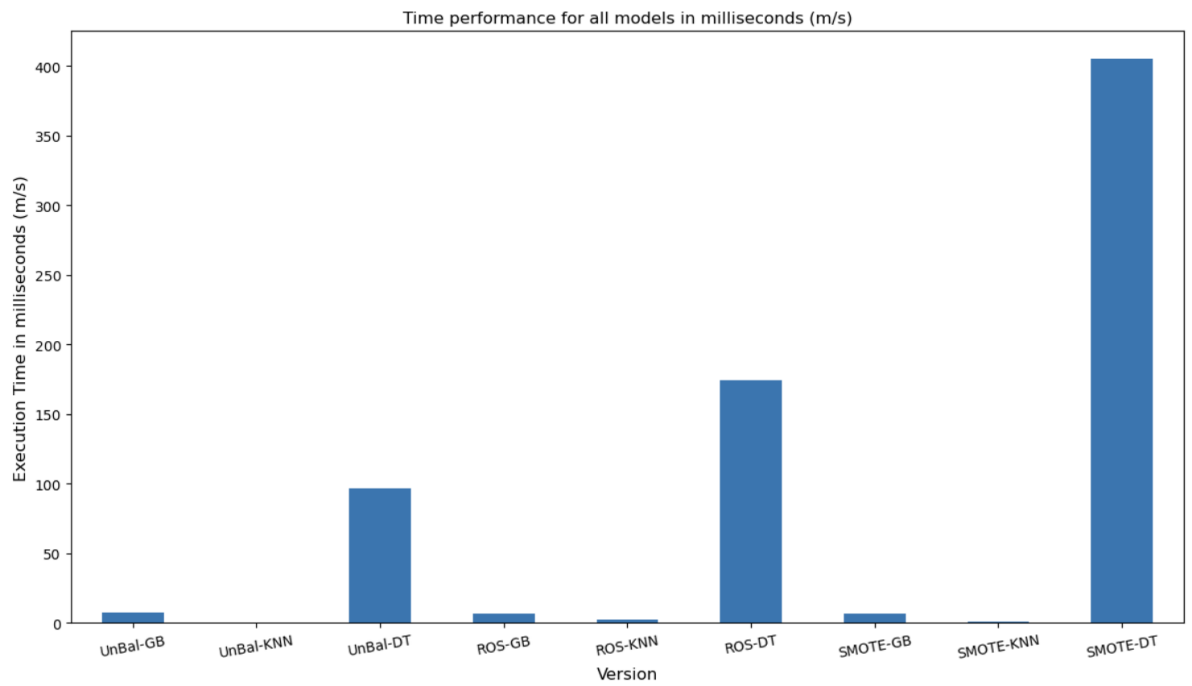
precision rate (16%) shows that many “survived” companies (Class 0) are predicted as “Bankrupt”, leading to high misjudgement of Class 1. Same for the oversampled model, GB has the lowest precision rate (53% for both ROS and SMOTE) though it has a good recall rate (91%-93%).

On the other hand, KNN and DT have very high and consistent scores for both classes in the oversampled models. ROS-DT performed the best as it scored over 90% for both classes. However, though the recall rates are nearly 100% for Class 0, they did not perform well for unbalanced data as it is critically low for Class 1. UnBal-DT has the lowest recall rate (9%) for the “Bankrupt” class which is significant. The lack of a feature selection process and unbalanced distributed data might be able to explain these phenomena. As there are only 220 rows of Class 1 in unbalanced data, the model might not be familiar to the minority class. Besides that, the curse of dimensionality might have a vital impact on the KNN model’s performance as aforementioned in section 4.

F1- score evaluates the class performance by combining the recall and precision rate into one metric, providing an aggregate performance. ROS-DT scored highest (98%) for both classes followed by ROS-KNN, SMOTE-KNN, and SMOTE-DT, indicating these oversampled models are performing well. In contrast, the “Bankrupt” class in the unoptimized model (UnBal) has a low F1-score, demonstrating that all model performance is actually affected by the unbalanced class issue. In short, KNN and DT performed very well after oversampling, while GB had a better recall rate for the unbalanced data set without feature selections.



**Figure 5.4** Overall performance for all models.



**Figure 5.5** Time taken for all models in milliseconds(ms)

|                  | Recall | Precision | F1-score | Time(ms) |
|------------------|--------|-----------|----------|----------|
| Model            |        |           |          |          |
| <b>UnBal-GB</b>  | 87.29  | 97.16     | 91.20    | 4.982    |
| <b>UnBal-KNN</b> | 97.31  | 96.51     | 96.63    | 1.029    |
| <b>UnBal-DT</b>  | 96.63  | 95.35     | 95.90    | 93.602   |
| <b>ROS-GB</b>    | 55.56  | 61.44     | 49.26    | 7.146    |
| <b>ROS-KNN</b>   | 96.44  | 96.68     | 96.44    | 0.998    |
| <b>ROS-DT</b>    | 98.46  | 98.51     | 98.46    | 194.464  |
| <b>SMOTE-GB</b>  | 55.33  | 62.53     | 48.10    | 7.173    |
| <b>SMOTE-KNN</b> | 94.44  | 95.00     | 94.43    | 0.867    |
| <b>SMOTE-DT</b>  | 93.41  | 93.43     | 93.41    | 460.618  |

*Figure 5.6 Overall performance for all models*

From Figure 5.4, the overall performance for the unbalanced model is good despite the model having a poor performance in predicting the minority class (Bankrupt) as shown in Figure 5.1 to 5.3. This demonstrated that the model performance is biased toward the majority class (survived), causing performance misleading. Besides that, we also discovered that DT performs better on the ROS model while KNN performs better for SMOTE. This may be because the ROS method does not affect the decision tree constructions, as the Class 1 data (220 instances) are replicated with the same criterion until reaching 6599 data instances. Thus, decisions can be made easily and accurately due to data repetition. However, SMOTE generates synthetic data in a similar but different value, which might change the criteria of the decision tree. It affects the tree construction and is the most time-consuming model (460.62 ms) as shown in Figures 5.5 and 5.6.

Contrarily, KNN performs better than DT in SMOTE might be due to the same underlying algorithm of SMOTE, which is KNN. It allows the KNN easier to identify its neighbour from those synthetic data, generating a better decision boundary. To conclude, despite ROS seeming to perform better than SMOTE in terms of time and accuracy, we are not able to conclude that ROS is the best oversampling method for our dataset. As the high accuracy of ROS possibly caused by overfitting. Although SMOTE has lower accuracy, yet the result is more reliable than ROS in real-life applications, where most of the data is non-identical.

In conclusion, we recommended the KNN model over the DT model if time is the main concern in the model selection. For the oversampling method, ROS has a higher accuracy score while SMOTE avoids the overfitting issues. If time and accuracy are equally important, the SMOTE-KNN model is recommended as the F1-score is acceptably high (94.43%) and consumes the shortest time (0.867 ms).

