2023-06

DESIGNING BANK DISTRESS PREDICTION MODEL USING MACHINE LEARNING ALGORITHMS

TILAHUN, TADESSE TEKLU

BAHIR DAR UNIVERSITY

BAHIR DAR INSTITUTE OF TECHNOLOGY

FACULTY OF COMPUTING

MSc Thesis:

DESIGNING BANK DISTRESS PREDICTION MODEL USING MACHINE LEARNING ALGORITHMS

By:

TILAHUN TADESSE TEKLU

June 2023

Bahir Dar, Ethiopia

BAHIR DAR UNIVERSITY

BAHIR DAR INSTITUTE OF TECHNOLOGY

FACULTY OF COMPUTING

DESIGNING BANK DISTRESS PREDICTION MODEL USING MACHINE LEARNING ALGORITHMS

By:

TILAHUN TADESSE TEKLU

A Thesis submitted

In partial fulfillment of the requirements for the Degree of

Master of Science in Computer Science

Advisor: Mekonnen Wagaw (PhD)

June 2023

Bahir Dar, Ethiopia

DECLARATION

This is to certify that the thesis entitled "DESIGNING BANK DISTRESS PREDICTION MODEL USING MACHINE LEARNING ALGORITHMS", submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Science under Computing Faculty, Bahir Dar Institute of Technology, is a record of original work carried out by me and has never been submitted to this or any other institution to get any other degree or certificates. The assistance and help I received during the course of this investigation have been duly acknowledged.

Name of Student	Signature	Date
Tilahun Tadesse Teklu		06/07/2023

This thesis has been submitted for examination with my approval as a university advisor.

Advisor Name	Signature
Mekonnen Wagaw (Phd)	

BAHIR DAR UNIVERSITY
BAHIR DAR INSTITUTE OF TECHNOLOGY
SCHOOL OF GRADUATE STUDIES
FACULTY OF COMPUTING

Approval of thesis for defense result

I hereby confirm that the changes required by the examiners have been carried out and incorporated in the final thesis.

Name of Student: - **Tilahun Tadesse** Signature: - Date: - **06/07/2023**

As members of the board of examiners, we examined this thesis entitled "**Designing Bank Distress Prediction Model Using Machine Learning Algorithms**" by **Tilahun Tadesse**. We hereby certify that the thesis is accepted for fulfilling the requirements for the award of the degree of Masters of science in "**Computer Science**".

Board of Examiners

Name of Advisor Signature Date

Mekonnen Wagaw (Dr.) July 29, 2023

Name of External examiner Signature Date

Mohammed Abebe (Dr.) August 03, 2023

Name of Internal Examiner Signature Date

Alemu Kumilachew 03/01/2016

Name of Chairperson Signature Date

Dagnachew Melesew 03/01/2016

Name of Chair Holder Signature Date

Kidest M. 03/01/2016

Name of Faculty Dean Signature Date

03/01/2016

Faculty Stamp

ACKNOWLEDGMENT

First of all, I would like to thank GOD as nothing would be possible without his divine support. I would like to say thank you very much to my advisor Dr. Mekonnen Wagaw- since he gives me advice related to this research starting from the beginning.

Next, I would like to thank my Managers Guluma Abdisa, and colleagues that who are always with me for every work so I am happy about that. Last but not least my wife for her unrestricted and everlasting love and support. She has always supported me in every situation, with care and understanding.

ABSTRACT

The problem of bank distress in the World banking industry has been a major issue for all the stakeholders, investors in the economy, and also the business world at large. In order to tackle any ensuing conditions of bank collapse, predictive analysis of a bank's financial situation and customer connection is quite beneficial. This study will be conducted to design a bank distress predicting model for the banks. To comply with the research objectives, Secondary sources of data will be used. To predict or forecast bank distress, an efficient Bank Distress Prediction (BDP) model has become necessary. In this regard, a wide range of Machine Learning (ML) models has been developed to predict distress in the banks. But, those BDP models have insufficient performance due to challenges like the presence of redundant, irrelevant features, and imbalance class problems. Imbalanced class occurs with data samples from two groups, the minority group contains considerably smaller samples than the majority group. The imbalanced class nature of the distressed data increases the learning difficulty of the classification algorithms to train the model. The use of imbalanced data leads to off-target predictions of the minority class, but which is considered to be more important than the majority class. These challenges depreciate the performance of the distress prediction model depending on the predictor’s ability to tackle data frauds. In this study, we proposed a bank distress prediction model that addresses imbalance class problems using Feature selection techniques (for selecting the significant features), Synthetic Minority Oversampling Techniques (SMOTE) used to produce balanced data and Random Forest (RF) for classification algorithms. Further, we implement four classifier algorithms Logistic Regression (LR), K-Nearest Neighbors (KNN), Decision Tree (DT), and Support Vector Machine (SVM). We implement Random Forest (RF) on the transformed or resampled dataset. To evaluate the performance of the proposed model, we did experiments on imbalanced datasets of the Polish Bankruptcy dataset from the UCI Machine Learning repository. Hereafter, the proposed model is expected to allow them to anticipate the status of businesses in the future and make decisions accordingly. The Experimental results show that the proposed model makes a very good result, in which 83% prediction accuracy and 78% by Decision Tree accuracy is attained for Polish Bankruptcy datasets. So, we conclude that the proposed model improves the performance of BDP effectively, and provides a brand-new way of dealing with the imbalanced dataset problem.

Keywords: ***Decision Tree, Support Vector Machine, Bank Distress Prediction, Synthetic Minority Oversampling Techniques, Logistic Regression, K-Nearest Neighbors, & Random Forest***

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
AUC	Area Under the ROC
BDP	Bank Distress Prediction
CBE	Commercial Bank of Ethiopia
CBS	Core Banking System
CSV	Comma Separate Value
DB	Distress Bank
DBE	Development Bank of Ethiopia
DT	Decision Tree
EBIT	Earnings Before Interest and Taxes
KNN	K-Nearest Neighbors
LR	Logistic Regression
NBE	National Bank of Ethiopia
NDB	Non-Distress Bank
RBF	Radial Basis Function
RF	Random Forest
ROC	Receiver Operating Characteristic
SMOTE	Synthetics Minority Oversampling Technique
SMOTEENN	Synthetics Minority Oversampling Technique-Edited Nearest Neighbor
SVM	Support Vector Machine
UCI	Union Cycliste International

CHAPTER ONE

INTRODUCTION

1.1 Background

The banking industry is regarded as the lifeline of the contemporary economy and is crucial to an economy's success or collapse. It is one of the crucial financial pillars in the overall financial system as well as the financial institution in question. To assure and sustain the economic development of the nation, a sound banking system effectively mobilizes savings, deploys them, and disburses credit to numerous productive sectors.

One of the principal functions of a bank includes the provision of credit to customers and workers. Thus, a bank is defined as “a place of business that lends, issues, exchanges and takes care of money, extends credits and provides ways of sending funds quickly from place to place”. According to (Abaenewe, Ogbulu, and Ndugbu, 2019), the banking sector stands out in the financial sector as of prime importance because, in many emerging countries of the world, the sector is virtually the only financial means of appealing private savings on a large scale. The main three categories of the bank's scope of operations include accepting deposits, investing such deposits, and the provision of credits to customers. In these ways, banks assist in mobilizing funds at their disposal for economic development in a nation. Through credits (loans and advances) and investment, business transactions are previously being financed for and on behalf of individuals and private enterprises, and of course, sometimes government agencies.

Bank distress (i.e., a circumstance when a business finds it difficult to make enough revenues to cover its financial commitments) usually affects all sectors of the economy. Globalization has also contributed to the banks involved in distress in recent times. (John, Gianni, and Elena, 2008) stated that bank distress can be triggered by weakness in the banking system, characterized by persistent illiquidity, insolvency, undercapitalization, level of non-performing loans, and weak corporate governance among others.

Many individuals mistakenly lump together bank distress and bank failure, which are actually two different things. Bank failure is preceded by bank distress. A bank that is distress has a chance to get better, but a bank that has failure has no hope of survival.

Banking sectors are playing a crucial role in the development and promotion of the economy and allow the transfer of resources from savers to investors. The problem of bank distress in the banking industry has been of huge concern to all stakeholders of the economy and the world business community at large. Therefore, the issue of bank distress is critical in the area of banking or financial institution sectors more than in other sectors. If the banking sector of a given country faces financial crisis (Demirguc-kunt and Detragiache, 1998).

The recent global financial crisis that has hit various countries throughout the world indicates that a timely and realistic distress model is essential. The two primary methodologies used around the world to estimate the likelihood of a company's distress are: the structural approach, in which the rm's properties, as well as interest rates, are carefully evaluated, and the probability of default is projected using data mining tools. Balcaen and Ooghe (2006) investigated statistical methods in depth, and Kumar and Ravi (2007) investigated intelligent techniques in conjunction with statistical techniques to predict the rate of distress. In this field, various analytical techniques have already been investigated, including multidimensional analysis and extended linear models. However, due to the ever-increasing amount of data, linear models are no longer dependable or effective in determining the relationship between economic indicators. Well-described the usage of machine learning and artificial intelligence for predicting corporate distress in the last couple of decades.

The majority of studies in this field have used accounting-based factors as well as numeric-based market-based variables. It is commonly acknowledged that a company's distress status is determined by a broad mix of elements, making it difficult to precisely anticipate the chances. Purchases, sales, assets, liabilities, EBIT, trading data, profit per share, and other numerical attributes are examples of numerical attributes. Despite the importance of anticipating distress, it is currently necessary to use the appropriate attributes to improve prediction performance and reduce computing time and cost. Second, the number of non-bankrupt companies is enormous, whereas only a small number of companies become bankrupt, despite the fact that these few companies are enough to disrupt multiple industries and the economy as a whole. Dealing with this type of class imbalance is difficult since the model would be biased towards the majority class, i.e., non-bankrupt enterprises, without careful planning and implementation. It is critical to address these two challenges effectively to construct an effective prediction model.

1.2 Motivation

For creditors and investors, estimating the risk of bank distress is critical. Because there are significant indirect and direct costs involved with financial difficulties and bankruptcies (Altman, 1984), distress prediction has occupied a wide field of research for a long time (Linden et al., 2015). Employees, investors, consumers, suppliers, and their financiers are all affected when a firm goes distressed (De Haas and Van Horen, 2012); employees, investors, customers, suppliers, and their financiers are all affected when a company goes distressed (Engström, 2002). In extreme situations, business distress can result in the demise of an entire industry (Yang et al., 2015).

Until recently, the most widely used methods for predicting bank distress were statistical models; however, machine learning models have recently been developed (Linden et al., 2015). Machine learning models have recently been employed to solve a variety of classification and regression issues, and they have consistently outperformed traditional classification methods (Krizhevsky et al., 2012). The goal of distress prediction is to evaluate a bank's financial health and future prospects. This topic can be treated as a two-class classification problem for a specific amount of time (Zieba et al., 2016). Banks either survive or go distressed throughout the specified period. The challenge is to determine which of these two possible outcomes is most likely.

1.3 Statement of the problem

Bank distress occurs when a bank's performance has consistently failed to fulfill established standards for evaluating a bank's financial health, as well as when the bank becomes insolvent or illiquid (i.e., when it is unable to pay its present obligations when they are due). On the other hand, when a bank's entire realization assets are worth less than its total liabilities, the bank is deemed bankrupt. Therefore, predicting bank distress is a crucial task to guarantee the proper operation and dependability of the banks. According to a survey of the literature, the BDP model has been constructed using a variety of machine learning techniques. The class distribution of the training data has a significant impact on the bank distress prediction model, which is why it is noted that the performance of these methods is very varied and constrained. Unfortunately, class imbalance and high dimensional features of the datasets are BDP's two main problems (Haixiang, *et al*, 2017). Since predicting bank distress is a binary classification job, there are only two classes that could be anticipated. With data examples from two groups, there is a class imbalance since the minority group has significantly lower samples than the majority group. More data samples are present in the majority class than in the minority class. The majority group will often be overclassified by learners when there is a class imbalance in the training data.

Finally, by categorizing the minority class into the majority class, this issue significantly reduces the prediction accuracy of the model. However, BDP models were developed using a variety of high-dimensional bank distress features, which has an impact on the model's performance due to the lack of feature selection approaches because unbalanced data contains redundant and unnecessary features as well as class distributions that are not uniform.

In addition, datasets extracted from UCI Machine Learning repository usually contains more correlated information, paired with random error and noised data. Generally, we can organize it into two main problems imposed by data with unequal class distribution, high dimensional features, and noised redundant information of the datasets, listed as follows:

The machine learning: Machine learning algorithms are built to minimize errors. Since the probability of instances belonging to the majority class is significantly higher in the imbalanced dataset, the algorithms are much more likely to classify new observations to the majority class because there is a high number of instances in the training dataset.

Feature selection problem: The data features that we used to train the machine learning models to have a huge influence on the performance you can achieve. Both irrelevant and partially relevant features can negatively influence model performance.

1.4 Research Questions

The following research questions make up a typical collection of research questions found in the associated literature, and they are the ones we hope to identify and report on in our work.

RQ1: How to design a bank distress prediction model for class imbalance datasets?

RQ2: Which machine learning algorithm is feasible and effective to predict bank distress?

RQ3: which bank distress features are critical for class imbalance distress prediction?

1.5 Objective of the study

1.5.1 General Objective

The general objective of the study is to design a bank distress prediction model using a machine learning algorithm.

1.5.2 Specific Objectives

The objectives of the study include:

1. To identify the most significant and critical features for bank distress prediction in the class imbalance dataset.
2. To gather and preprocess the class imbalance datasets, training and testing the model.
3. To investigate the effect of attribute selection for bank distress prediction using class imbalance datasets.
4. To assess the effectiveness of the proposed Bank distress prediction model.

1.6 Scope and Limitation of the study

The aim of this research is concerned with designing, modeling, and developing a model for predicting bank distress in class imbalance datasets. The efficiency of the model increases the fitness of the distress prediction model by selecting the appropriate feature for the proposed model. The SMOTE sampling approach is used to solve the class imbalance problem of the distress bank dataset. In this study, we used the Polish Bankruptcy dataset from the UCI Machine Learning repository. Finally, performance measures show the statistical view of an experiment's results, Analysis, together with an explanation of the produced result and measure the accuracy of the proposed model. we used a free Jupiter notebook editor and Anaconda tool with different libraries for the implementation of the proposed model.

1.7 Significance of the Study

This study's importance stems from the fact that banking serves as the engine for the expansion and development of any economy, necessitating the necessity and urgency of controlling distressed banks to prevent a financial crisis.

The facts presented in this research work will be of immense assistance to economic planners, investors in the banking industry, and industrialists that need banks for growth and development. This research becomes more significant as a result of the increasing determination in the financial conditions of banks in Ethiopia today due to the threat of distress and the potential for economic chaos if uncontrolled.

In Ethiopia currently, different private banks are starting work, and in the future, if the number of banks increases bank distress may happen. So, this study will also be used by the National Bank of Ethiopia to forecast or predict bank distress before it occurs. And also, this study will be input for other researchers and for other banks in Ethiopia to follow up on their financial conditions or performance of the bank.

Technically, this study also has the following significance.

- we leverage SMOTE sampling techniques which efficiently solve the binary class classification problem of bank distress prediction.
- we proposed a multi-level ensemble of machine learning algorithms and evaluate the performance of each algorithm.

1.8 Organization of the Study

This section presents an overview of the contents of the remaining chapters and the study is organized into five chapters. The first chapter describes the introductory part of the study. In chapter two, the literature reviewed the concept of prediction of bank distress, and the approaches used for predicting distress are presented. Chapter three presents the specific research methodology used in this study and the detailed description of the proposed system and components that compose the system: data preprocessing that includes data cleaning, feature selection and classification, and regression using different machine learning algorithms. Chapter four presents an experimental evaluation of the proposed model for the prediction of bank distress described in aspect. Finally, chapter five presents a summary, conclusion, and recommendation of the study.

CHAPTER TWO

LITERATURE REVIEW

2.1 Conceptual Literature

2.1.1. What is Bank Distress?

Banks in distress are in a bad state, suffering greatly from their operational activities as a result of a number of highly volatile issues, including inconsistent and discontinuous policies, ineffective management, and undercapitalization.

Bank Distress, according (Investopedia, 2020), is a situation in which a corporation or individual is unable to earn sufficient sales or income to satisfy or pay its financial obligations. High fixed costs, a high percentage of illiquid assets, or revenue that is sensitive to economic downturns are all reasons for this. For individuals, bank distress can arise from poor budgeting, overspending, too high of a debt load, lawsuits, or loss of employment. Bank distress happens when revenues or income no longer meet or pay for the financial obligations of an individual or depositor, it is often a harbinger of bankruptcy and can cause lasting damage to one's creditworthiness. To remedy the situation, a bank or individual may consider options such as restructuring debt or cutting back on costs.

There is extensive literature on the various methods and analyses performed, regarding bank distress prediction. (Demyanyk and Hasan, 2009) provide a summary of various papers focusing on analyzing, forecasting, and providing remedial actions regarding potential financial crises or bank distress. This study contains a complete summary of existing literature regarding the problem of forecasting or predicting bank distress for reasons of completeness.

2.1.2 Determinants of Bank Distress

Bank's liquidity is the ability of an asset to be converted to cash quickly at a low cost. Brealey et.al. (2000) suggests that liquid assets can be converted into cash quickly and cheaply. Liquidity refers to the solvency and ability of the banks to pay short-term liabilities as they come due. Because a common precursor to Bank distress and bankruptcy is low or declining liquidity, these ratios are viewed as good leading indicators of cash flow problems (Gitman, 1991).

Leverage represents a risk to the bank, which covers a portion of the fixed costs. Operating leverage refers to fixed operating cost and a measure of operating risk found in the bank's income statement, whereas financial leverage is a measure of financial risk and financing of a portion of the bank's assets to raise and grow the return to the common stockholders. According to (Shim and Siegel, 1998) the higher the financial leverage, the higher the financial risk, and the higher the cost of capital.

2.1.3 Bank and the Soundness

According to Kasmir (2015), a bank is a financial institution whose main activities are collecting funds from the public and making funds available to the community, and providing other banking services. The bank's financial statements show the overall financial condition of the bank. These financial statements act as measures of the actual condition of the bank and can demonstrate the performance of the bank's management during the period.

The soundness of banks is the result of qualitative checks of numerous factors affecting the circumstance or overall performance of a financial institution thru the evaluation of capital, asset quality, management, profits, and liquidity. The health of a bank is the concern of all stakeholders, both owners and managers of banks, the society as users of bank services. Given their important role in the financial well-being of communities, it is necessary to assess the soundness of banks. The goal is to determine if the actual condition of the bank is in good health, less healthy or sick. If the condition of banks is healthy, it is necessary to maintain health. However, if the bank is in an unhealthy condition, then immediate action should be taken to solve it Kasmir (2015).

2.1.4. Machine Learning in Bank Distress

It is well known that machine learning is a group of methods focused on forecasting. The ability of machine learning algorithms to better account for potential complexities in the data is what makes them so strong when compared to traditional statistical models. Some machine learning methods use methods for merging predictive models collectively, such as random forest and boosted decision trees. It turns out that even if a single model is the best on a standalone basis, combining various correct models is likely to perform better.

By comparing and contrasting several machine learning techniques (random forest, boosted decision trees, K-Nearest neighbors, and support vector machines) with classical statistical approaches (logistic regression and random effects logistic regression) are used to predict bank distress. The relationship between the input and output data is learned by each machine learning and traditional technique in their own unique way, in the case of bank financial ratios, balance sheet growth rates, and macroeconomic data for the former and subjective supervisory assessment of bank risk for the latter.

2.1.5. Machine Learning Classification

There are perhaps four main types of classification:

- Binary Classification
- Multi-Class Classification
- Multi-Label Classification
- Imbalanced Classification

2.1.5.1 Binary Classification

It refers to the class responsibilities which have magnificence labels. Typically, binary class responsibilities contain one magnificence which is the ordinary nation, and some other magnificence which is the extraordinary nation. The magnificence for the ordinary nation is assigned the magnificence zero and the magnificence with the extraordinary nation is assigned the magnificence label 1.

Some algorithms are planned for binary classification and do not natively support more than two classes.

2.1.5.2 Multi-class Classification

Multi-class classification jobs are those that use more than two class labels and are referred to as such. The concept of normal and abnormal outcomes is absent from the multi-class classification, which also lacks binary classification. For multi-class classification, well-liked methods include k-Nearest Neighbors, Decision Trees, Naive Bayes, Random Forest, and Gradient Boosting.

Algorithms designed for binary classification can be adapted for use in multiclass problems. This involves using a strategy to adapt multiple binary classification models per class to all other classes (called one-vs-rest) or one model per pair of classes (called one-vs-one).

- **One-Vs-Rest:** For each class versus all other classes, it is a best fit one binary classification model.
- **One-Vs-One:** For each couple of classes, it is a best fit one binary classification model.

2.1.5.3 Multi-Label Classification

A multi-label classification is a classification task that has more than one class label and can predict one or more class labels.

Multi-label classification tasks are frequently modeled using a model that generates many outputs, each of which is predicted as a Bernoulli probability distribution. Essentially, this is a model that predicts several binary classifications.

The algorithms used for binary or multi-class classification cannot be simply used for multi-label classification. You can use a multi-label variation of the conventional classification algorithm, which includes the following features:

- Multi-label Decision Trees
- Multi-label Random Forests
- Multi-label Gradient Boosting

Another approach is to use a separate classification algorithm to predict the label for each class.

2..1.6 Imbalanced classification

An example of a classification issue where the distribution of instances among the recognized classes is biased or unbalanced is an imbalanced classification problem. One case in the minority class for hundreds, thousands, or millions of examples in the majority class or classes might indicate a mild bias all the way up to a serious imbalance.

Predictive modeling is challenged by imbalanced classifications since the majority of machine learning methods for classification were built on the premise that there should be an equal number of samples in each class. As a result, models perform poorly in terms of prediction, particularly for the minority class. This is a problem because, in general, the minority class is more significant and, as a result, the issue is more susceptible to mistakes in categorization for the minority class than the majority class.

2.1.7. Bank Distress Prediction Techniques

BDP techniques are the most important activities of the testing phase of the bank health that identifies distress and non-distress. Even if distress prediction is very important in testing bank status, it is not always easy to predict distress in banks. BDP has the task of binary classification problems (David, et al, 2018). These binary classification problems are widened when the target prediction class has a class imbalance problem. Different Machine Learning techniques have been studied to address imbalanced problems extensively over the last few decades (Xuan, et al, 2019). As such, there are three types of methods for handling this problem: Algorithm-level, Data-level, and combine approaches (Haonan, et al, 2018).

2.1.7.1 Algorithm-level methods

It is also called a cost-sensitive learning method for dealing with data imbalance. It considers different misclassification costs for different classes in such a way that the data samples of minority classes get importance. However, finding the cost metrics for cost-sensitive distress classifiers is another big challenge, and still, there is no systematic way of setting cost metrics.

2.1.7.2 Data-level methods

Data-level methods, such as random over-sampling, random under-sampling, and SMOTE oversampling techniques, are resampling strategies used to manipulate training data to fix tilted class distributions. It is a strategy for dealing with class imbalance data problems utilizing the sampling method, and it can improve the performance of classification algorithms by modifying training data before feeding it to the machine learning classification algorithm.

2.1.7.3 Ensemble methods

Ensemble methods in machine learning combine many learning algorithms to achieve greater predicted performance than any of the individual constituent learning algorithms (Haonan Tonga, et al, 2018).

Multi-level ensemble methods typically produce more exact solutions than a single model would. This was the case for many machine learning contests where the award-winning solution used ensemble techniques. So, to produce a more accurate prediction model we used a multi-level ensemble approach which included a support vector machine and decision tree model for different purposes. A support vector machine model or algorithm is used for classification and regression, while a decision tree algorithm is used to predict bank distress based on classified data.

Ensemble techniques are ideal for reducing the variance of the model and thereby improving the accuracy of the predictions. Dispersion is eliminated by combining multiple models to form a single prediction that is selected from all other possible predictions from the combined model. The model ensemble combines different models to make the best possible prediction of the result, taking into account all the predictions. (https://corporatefinanceinstitute.com/). Ensemble methods use many models to improve forecast accuracy. Bagging, boosting, and stacking are three ensemble methods commonly employed in BDP. These ensemble methods have the potential to outperform single classifiers and potentially identify bank distress.

2.1.7.3.1 Main Types of Ensemble Methods

1. Bagging

Bagging means in another term it is bootstrap aggregating and it is mostly applied in classification and regression. Through the decision tree, the accuracy of the model is improved and the variance is significantly reduced. Reducing the variance improves accuracy and eliminates the overfitting that is a challenge for many predictive models.

2. Boosting

Boosting learns from past predictor errors to make improved predictions in the future. This technique combines multiple weak base learners into one powerful learner, greatly improving the predictability of the model. Boost works by placing weak learners in a sequence, so weak learners learn from the next learner in the sequence to create a better predictive model.

3. Stacking

Stacking is another term referred to as stacked generalization. This method works by allowing training algorithms to be mixed with predictions from various additional learning algorithms.

Regression, density estimation, distance learning, and classification have all used stacking successfully. It can also be used to determine if error rates are sagging.

How does the Ensemble method work?

Since we want to develop a machine learning model that predicts bank distress based on historical data we have gathered from previous years. we train two machine learning models using a different algorithm: support vector machine and decision tree. However, even after many adjustments and configurations, none reach the desired 95% prediction accuracy. These machine-learning models are called weak learners because they fail to converge to the desired level.

But weak doesn't mean useless. You can combine them into an ensemble. For each new prediction, we run my input data through both two models and then compute the average of the result.

Because the ensemble method is efficient, your machine learning models work differently. Each model strength achieves well on some data and less accurately on others. When we combine both of them together, they cancel out each other's weaknesses.

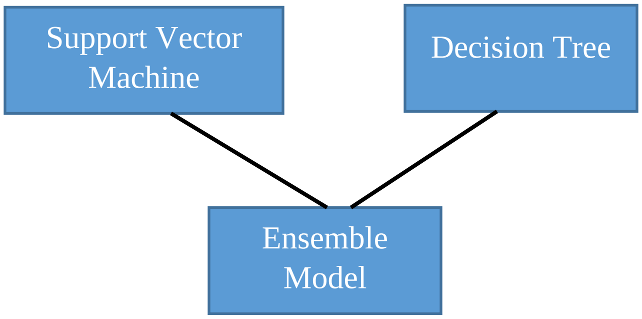


Figure 2 1. Multi-Level Ensemble

Challenges of the Ensemble Method

Using an ensemble means spending more time and resources on training machine learning models. Another problem with ensemble methods is explaining ability. The soundness of banks is the result of qualitative checks of numerous factors affecting the circumstance or overall performance of a financial institution through the evaluation of capital, asset quality, management, profits, and liquidity. A single machine learning model such as a decision tree is easy to trace, but when you have hundreds of models contributing to an output, it is much more difficult to make sense of the logic behind each decision.

2.1.8. Supervised Machine Learning techniques for distress prediction

Barboza et al. (2017) take the research on this topic to the next level by contrasting machine learning models with statistical models and determining which methodological approach is superior. The predictor variables were liquidity, profitability, leverage, productivity, and asset turnover, and the dataset was balanced. Bagging, boosting, Random Forest, ANN, SVM with two kernels (linear and radial basis), logistic regression, and MDA were among the techniques used. Traditional statistical models performed worse than machine learning models, according to the findings. One of the study's key flaws is that no feature selection strategy was used, which is a common practice nowadays.

Geng et al. (2014) used the KDD methodology to create numerous classifiers, including neural networks, decision trees, and support vector machines, building on prior research. The dataset utilized in this study included 31 financial variables from three to four years before the companies filed for bankruptcy. Several models based on statistical probabilistic theory were developed, including the CR tree, DT, NN, C5.0, logit, Bayes probability, and SVM DA. SVM, C5.0, and NN models outperformed other techniques and were employed in the data mining process. Accuracy, recall, and precision were employed as evaluation measures. The neural network model, with a prediction accuracy of 78 percent, outperformed the SVM and DT models. The authors used a variety of train-test splits, and the results were noticeably different, demonstrating that prediction accuracy is also affected by the train-test split ratio.

Ding has suggested a prediction model that is based on SVM (2008). For determining the best parameter value, the author used a grid-search technique combined with a 10-fold CV. The financial ratios of 250 enterprises were extracted from A-share market data from two Chinese cities. RBF SVM produced better results than MDA and BPNN.

Devi and Radhika conducted a thorough examination of some of the most common data mining strategies used to solve the bankruptcy prediction problem (2018). This study looked at a variety of strategies, from statistical methods to machine learning techniques. Machine learning techniques based on meta-heuristic optimization have also been proposed as a way to increase prediction accuracy. Accuracy, sensitivity, specificity, and precision were the evaluation measures used. The data mining process was carried out using the Apache Mahout tool, and it was discovered that SVM-PSO had the best metrics performance, with an accuracy of 95%, specificity of 95%, and precision of 94%.

2.1.9 Machine Learning Models

2.1.9.1 Decision Tree

A data structure made up of nodes and edges is a tree. The root node, branch/internal nodes, and leaf nodes are the three different types of trees that make up a tree. The decision tree is a straightforward illustration of a method for classifying objects into a limited number of categories. A leaf node is tagged with several classes, while the edges are labeled with the potential values for the attribute and the internal and root nodes are marked with the name of the attribute. One of the most often used categorization models is the decision tree since it is simple to use and the results are simpler to comprehend.

The decision tree is a visual representation of the choice made using trees. The construction of classification or regression models by decision trees use a tree structure. Using data mining, create a decision tree. The decision tree technique entails modeling the data or label into a tree and then turning that model tree into a rule. The main benefit of adopting a decision tree is its ability to simplify complicated decision-making processes so that problem-solvers may more easily grasp them. The decision tree may be used to investigate the data and uncover undiscovered connections between certain candidates and the input variables for the goal variable.

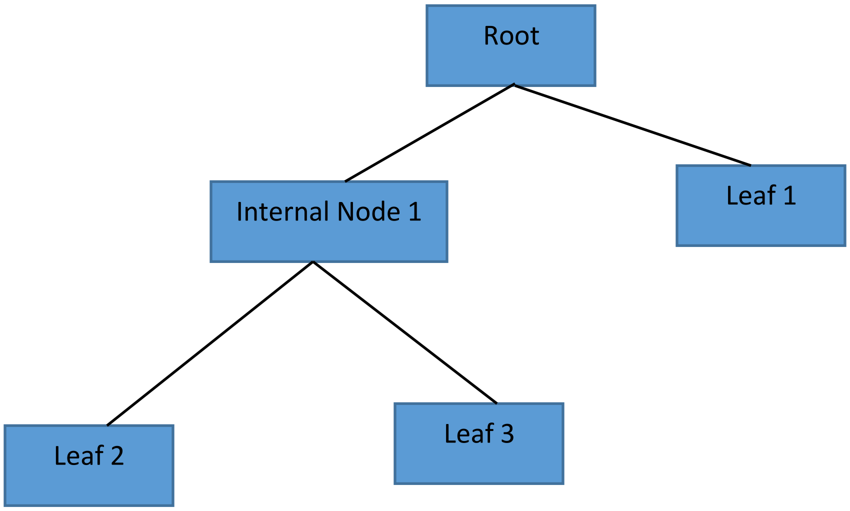


Figure 2 2. The basic concept of Decision Tree

2.1.9.2 Support Vector Machine

Support Vector Machine is a technique to make predictions, both in the case of classification and regression. SVM is in a class by Artificial Neural Network regarding functionality and condition problems that can be solved. Both are included in the class supervised learning. Support Vector Machine is a selection method that compares the standard parameter set of discrete values, called the candidate set, and takes the one that has the best classification accuracy. By changing the kernel function, it can be possible to find a hyperplane to determine the classification of non-linear by making hyperplane lines that emerge through the data set. This determination is based on Gaussian radial basis and tangents. Support Vector Machine is a learning system to classify data into two or more groups.

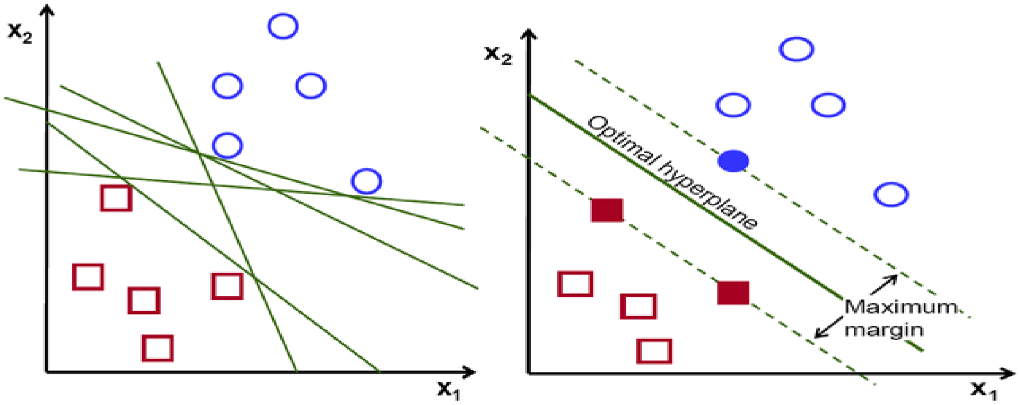


Figure 2 3. Possible SVM hyperplanes

There are many possible hyperplanes to choose from to separate the two classes of data points. The objective is to find a plane that has the maximum margin, i.e., the maximum distance between data points of both classes. Maximizing the edge distance provides some gain for a more reliable classification of future data points.

Hyperplanes and Support Vectors

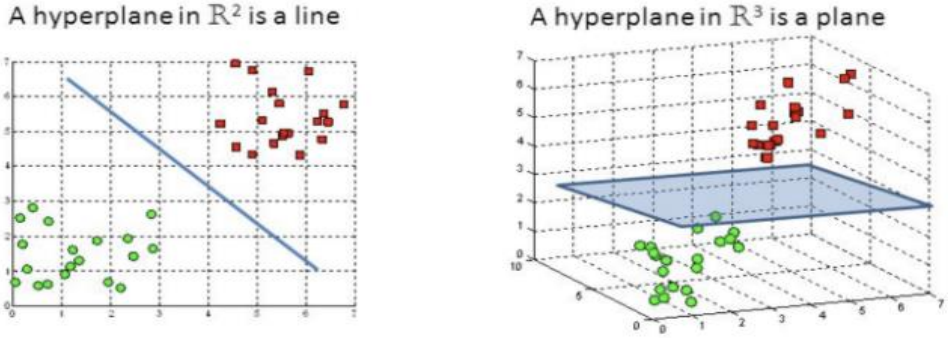


Figure 2 4. Hyperplanes in 2D and 3D feature space

Hyperplanes are decision boundaries that help classify the data points. Data points dropping on either side of the hyperplane can be attributed to different classes. Also, the dimension of the hyperplane depends upon the number of features. If the number of input features is 2, then the hyperplane is just a line. If the number of input features is 3, then the hyperplane becomes a two-dimensional plane. It becomes hard to visualize when the number of features exceeds 3.

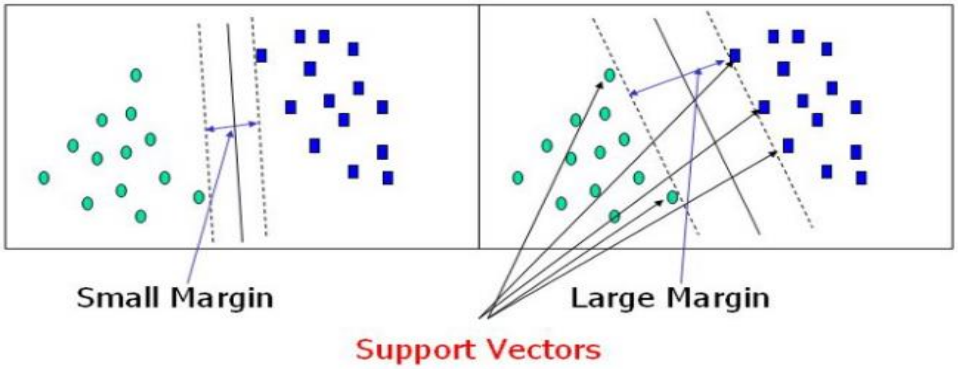


Figure 2 5.Support Vector Machine

Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane. Using these support vectors, maximizes the margin of the classifier. Deleting the support vectors will change the position of the hyperplane.

2.1.9.3 Random Forest

Random forests are a type of ensemble learning. The algorithm was first proposed by Ho (1995). A final model is created by combining multiple weak learners. In the case of random forest decision trees, the weak learners are, however, a weak learner is simply a model that outperforms chance. A random forest is essentially a forest of decision trees that have been trained on data that has been randomly sampled. The primary premise is that by merging many decision trees, better prediction performance can be achieved than by using just one of them alone. Both classification and regression applications can benefit from the random forest approach.

Bagging, which is short for bootstrap aggregating, is the foundation of the random forest training algorithm. Each decision tree is trained with a replacement on a data set randomly picked from the training data. A tree learner is presented with a different subset of the training data for each tree learner, and the same observation can be chosen multiple times in a sample (Breiman, 1996).

Deep decision trees typically have a large variance and low bias, indicating that they are prone to overfitting. However, by including several trees trained on different data sets in the model, the variance can be decreased. When creating a random forest model, this is what is done. Each decision tree is trained on a portion of the data points in the training set, with a vote determining the final forecast. As a result, the variance is reduced at the expense of a slight bias increase.

As shown in the diagram below, RF is working with the following four steps:

- **First Step:** start with the selection of random samples from a given dataset.
- **Second Step:** Create a decision tree for every sample of the dataset. Then it gets the defect prediction result from every sample of the decision tree.
- **Third step:** In this step, voting is performed for every predicted result of the decision.
- **Fourth Step:** Finally, selecting the most voted prediction result among the samples as the final defect prediction result.

The following diagram illustrates how random forest algorithms are working.

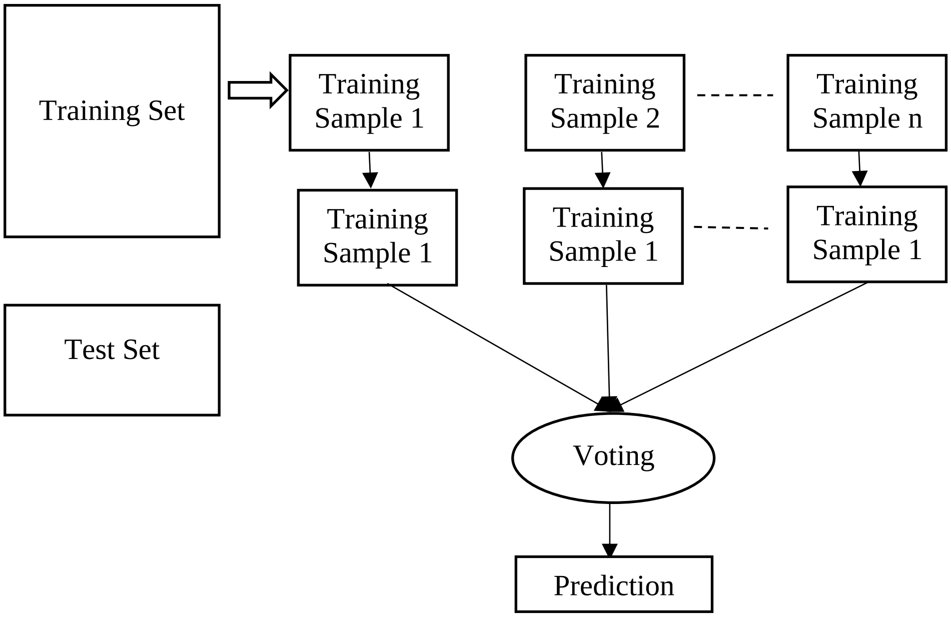


Figure 2 6.Random Forest Algorithm

2.1.9.4 Logistic Regression

For binary (two-class) classification issues, one of the most prominent machine learning techniques is logistic regression (LR). LR is a predictive analytic technique that uses a more complicated cost function and is based on the concept of probability. (Rana & Tarhan, 2018) It defines and evaluates the link between one dependent binary variable and independent factors. One dependent variable can only assume one of two potential situations when using the LR classification algorithm.

2.1.9.5 K-Nearest Neighbors

In this algorithm, the labeled point will be x, and the closest point (k = 1) or many points (k = n) near it will be labeled as y. A significant amount of data x may lead to a similar outcome for y. For instance, if a coin is tossed one million times with a bias (where one outcome is more likely than the other), the outcome is nine hundred thousand times heads. Naturally, the following toss will almost certainly be ahead. KNN employs a similar strategy in this instance.

$$C(x) = Y \ (1)$$

The K nearest neighbor classifier divides this dataset into three sets of clusters, each indicated by a different color in the figure below. The closest neighbors of X are shown by the arrows. Because it has the greatest number of nearest neighbors in the K value set (K = 5 in the figure), X is categorized as orange in this instance. Depending on the value of k (in this case, k = 1), the classifier C assigns x to its nearest neighbor Y. The error rate would be assured to be less than twice the Bayes error rate, which is the minimum error rate based on the distribution of the data, as the size of the data set grows.

$$P^*\leq P\leq P^*(2-(C/(C-1))\ P^*)$$

The formula for calculating the tight error boundary is given above. P* refers to the Bayes error rate, C is the number of classes, and P denotes the error rate of the nearest neighbor approach. For instance, it is quite likely that X will be categorized with Y when there is a lot of data, Y is the closest neighbor, and X needs to be classified. The likelihood of a mistake arising from this classification will be larger than the population-based minimum rate and less than twice that error rate, as indicated by the calculation above.

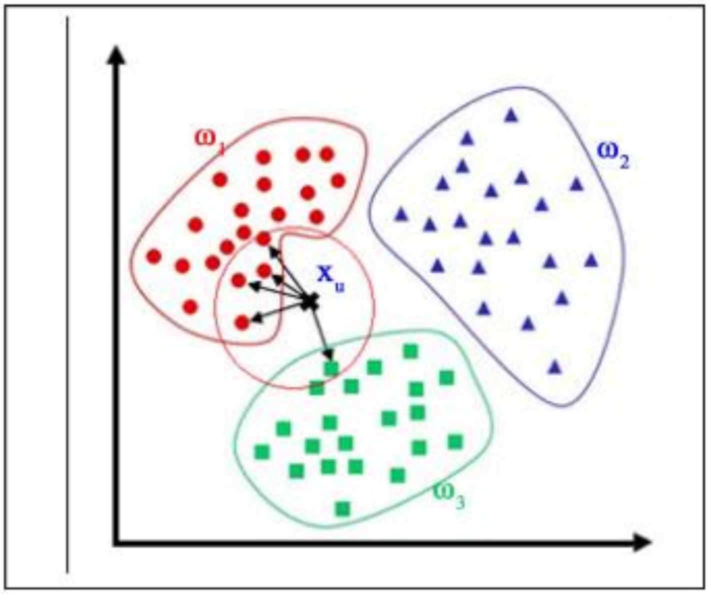


Figure 2 7. K nearest Neighbor model

2.1.10 Data pre-processing

Data pre-processing techniques are important and widely used in machine learning and data mining. There are many factors negatively affecting the performance of distress prediction models such as redundant, duplicate, and irrelevant information or noisy data. These problems can be solved by using data pre-processing techniques including normalization, missing value imputation, data sampling, data cleaning, and attribute(feature) selection.

2.1.10.1. Normalizing

Normalization of the data is done by for each feature *xi* replace it by *yi* calculated as:

$$yi = \frac{xi - \bar{x}}{s},$$

Where $\bar{x}$ and s are the estimated mean and standard deviation computed on the training data set.

2.1.10.2 Missing value imputation

There are few complete real-world data sets. The majority of them lack certain data points. Missing values are a concern, and the ideal way to deal with them may easily be the subject of a whole thesis. In machine learning, the most crucial factor is making sure that no crucial information is lost when dealing with these missing data. Therefore, the imputation can often be split into two distinct procedures. First, a technique must be developed to substitute a legitimate value for the absent one (Donders et al., 2006). In the absence of this, the algorithm is unable to handle observations with missing values. Second, an optional technique can be employed to advise the algorithm of which values were missing. For instance, a new feature may be created to represent the number of features that an observation lacked.

2.1.10.3. Data Sampling

Data sampling is a data resampling technique that modifies training data to balance out the dataset's skewed class distributions. These methods alter the training distributions to lessen imbalance or noise caused by anomalies or incorrectly labeled samples. To achieve an equal distribution of classes between the two classes, data resampling techniques will be used to add or delete samples from the training dataset. The common ML classification methods can be successfully fitted on the altered datasets after obtaining the balanced dataset. In machine learning, three different types of data sampling approaches are employed. These sampling techniques include under-sampling, oversampling, and SMOTE.

A. Under-sampling

In the machine learning community, under-sampling is a popular method of data sampling. It is characterized by the removal of some observations from the majority class, which equalizes the number of examples in each class and produces balanced data. Random under-sampling is the sole under-sampling technique that is employed (RUS). The RUS technique randomly discards the bulk of class instances until a more balanced distribution is attained. Because the majority of sensitive information would be removed randomly, this could result in poor generalization to the test set.

A new model that examines the effects of over and under-sampling on fault-prone module detection was developed in (Haixiang, et al., 2017). They conclude that although RUS is useful for balancing data, it is ineffective if sensitive and valuable information is removed at random. Furthermore, repeated sampling frequently results in significant underfitting and creates extra issues with noisy data samples.

B. Over-sampling

Another data sampling approach that is frequently utilized in machine learning for class imbalance data is oversampling. Oversampling techniques make an effort to balance the data by either creating fresh synthetic samples of the minority class or reproducing the minority class. Random oversampling, which merely copies randomly chosen characteristics from the minority class, is the simplest form of 27 oversampling. A new software defect prediction model employing oversampling methods is proposed in (Cholmyong Pak, et al., 2017). They explained how oversampling works with unbalanced data. They conclude that oversampling works well and is used by many classifiers. However, difficulties with overfitting and losing sensitive information can arise with both random-oversampling and random-under-sampling techniques.

C. SMOTE Sampling

Synthetic Minority Over-sampling Technique (SMOTE) is a special and more advanced sampling method that aims to overcome the drawback of random oversampling techniques where new instances are created by combining features of the target instance and its nearest neighbors

(Cholmyong Pak, et al, 2017). This sampling method generates artificial minority class instances from existing ones, instead of duplicating existing instances, and it works in the feature space rather than the data space. SMOTE is an effective data sampling technique that achieves a balanced dataset by creating extra training sample data for a minority group, in which the minority class is over-sampled by creating synthetic examples rather than replicating (Cholmyong Pak, et al, 2017).

SMOTE is an effective data sampling technique that achieves a balanced dataset since it works based on nearest neighbors judged by Euclidean distance between instances in the feature space of the data samples of the minority class. To create the new artificial minority class instance, SMOTE randomly selects one existing minority class sample m Next, the algorithm should find its k-nearest neighbors and should select at random one of the k samples, called n. Subsequently, it is necessary to calculate the difference between samples m and n and then multiply this with a random number between 0 and 1.

2.1.10.4 Data cleaning

Since Polish enterprise's datasets are directly taken from the source UCI Machine Learning Repository, which contains duplicate, redundant, and irrelevant features that negatively affect the performance of the classification model. Therefore, data cleaning alleviates these challenges when we apply it before the distress prediction model is built.

Missing Values: The attributes for which at least one instance value is not present are known as attributes with missing values. It was mentioned that missing values in datasets can occur due to division by zero error (Misha & Sarika, 2016). The possible solution is either to remove all instances which contain missing values or replace the missing values with zero.

Duplicate Values: When two or more banks' features or attributes have similar values for all instances then those attributes are said to contain identical values. Therefore, only one of them can be preserved and the remaining can be removed as redundant data in which they depreciate the performance of the prediction model.

Constant Values: Those attributes in which every instance has the same value. Since such attribute no information to the data, they can be deleted.

2.1.10.5 Feature selection

The process of identifying and selecting the most relevant features to build a robust prediction model is known as feature selection. The feature selection method selects a subset of features that are used as independent variables in the prediction model. It was found that feature selection methods produce the subset of features that can be used for the creation of a model without affecting the classification quality of the prediction model (Shuib Basri, et al, 2019). Nowadays, there is an increase in the size and complexity of banks, and an accurate prediction of defects is a crucial issue based on several attributes.

In the software engineering area, there are two common feature selection methods: filter-based feature selection and wrapper-based feature selection methods.

Filter-Based Feature Selection (FBFS) method evaluates and ranks features in datasets that are discovered to be independent of the prediction model independently using the computational characteristics of the datasets. (Shuib Basri, et al, 2019). The filter-based approach practices the intrinsic features of the data based on a given metric for feature selection and does not depend on the training of the learner algorithm (Satria Wahono, et al, 2014). Features are chosen based on how well they performed individually in the numerous statistical tests that looked at how well they correlated with the end variable. To identify a linear combination of features that distinguishes between two or more classes of a categorical variable, linear discriminant analysis is utilized. Filtering techniques are excellent for removing irrelevant, redundant, constant, duplicated, and linked characteristics since they have a short processing time. The majority of class imbalance learning techniques necessitate precise parameter settings to train the prediction model and feature selection to regulate the degree of the minority class's emphasis before learning.

The wrapper-based feature selection methods involve training learner algorithms during the feature selection process. This method involves analyzing a subset of features using a machine learning algorithm that uses a searching approach to browse through the space of potential feature subsets and evaluates each subset based on the effectiveness of a certain classification algorithm. Because the wrapper approach seeks out the best feasible feature combination that yields the best prediction model, it is also known as a greedy searching algorithm. For a given dataset, a wrapper-based technique may produce different feature subsets when using different learners. These problems of a wrapper-based technique lie in its high computational cost and risk of overfitting to the model (Prasanth, et al, 2017).

2.1.11. Performance Evaluation Metrics

To measure the presentation of the proposed model, we used confusion, which is commonly used for a binary classification problem. It is a tabular format that is used to describe the performance of classification of the model or classifier on the given dataset of test data for which the true values are well-known. Accuracy, Precision, Recall, and F1-score, are widely used evolution metrics in bank distress prediction.

2.1.11.1 Confusion Matrix

A two-by-two matrix formed by counting the number of four outcomes of a binary ML classifier is confusion matrix. Confusion Matrix is needed for finding Accuracy, Precision, Recall, and F1-score, which represents in the following section: from these four principles, four evaluation metrics have been calculated. Through a variety of measurements derived from the confusion matrix, the performance is assessed and evaluated. A confusion matrix consists of the following four parameters:

True Positive (TP)

- An instance that is classified and is positive correctly as a positive instance
- The predicted value matches the actual value
- The actual value was positive and the model predicted a positive value

True Negative (TN)

- An instance that is classified and is negative correctly is negative.
- The predicted value matches the actual value.
- The model anticipated a negative result, while the actual value was negative.

False Positive (FP)

- An instance that is negative but is classified wrongly as a positive instance.
- The prediction was incorrectly made
- The model predicted a positive value but the actual value was negative

False Negative (FN)

- An instance that is positive but is classified incorrectly as negative instances.
- The predicted value was falsely predicted
- Despite the model's predictions being positive, the actual result was negative.

Predicted Class			
Actual Class		Class = Yes	**Class = No**
	Class = Yes	True Positive	**False Negative**
	Class = No	False Positive	**True Negative**

Figure 2 8. Confusion Matrix

Accuracy: - is the most widely used spontaneous presentation measurement, and it is simply a the ratio of correctly predicted samples to the total samples. It delivers the best result if the cost of false positives and false negatives are similar. It measures the proportion of the files classified correctly, to the total number of files.

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$$

Precision: The percentage of files that were accurately labeled as defective over the total number of files labeled as either faulty or non-faulty is known as precision. In other words, precision or Confidence denotes the proportion of Predicted cases that are indeed real faulty files (Menzies & Greenwald, 2007). This is a measure of how good a prediction model is at identifying actual faulty files. It talks about how accurate your model is out of those predicted positives, how many of them are positive.

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

Recall: Recall is the percentage of defective files over all accessible defective files that are accurately identified as defective. The percentage of actual problematic files that are appropriately identified as faulty files is known as recall or sensitivity.

$$Recall = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

F1-Score: The weighted harmonic average of accuracy and recall is used to calculate the F1-Score, as illustrated in the equation. F1 often has more applications than accuracy, particularly in cases of unequal class distribution. It is preferable to include both precision and recall if the costs of false positives and false negatives differ significantly.

$$F1 - Score = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

2.2 Related Literature Review

A large number of bank distress prediction models have been proposed so far using different machine learning on bank distress datasets. Only those works contributions are related to our work are discussed.

Though a variety of earlier studies have successfully used machine learning techniques for predicting and detecting defects in bank operations on the balanced dataset, these techniques produce inadequate results when applied on class imbalanced datasets. Therefore, in this section, we reviewed the researches that are most related with this study on imbalanced data. The use of imbalanced datasets leads to off-target predictions of the majority class, which is generally considered to be more important than the majority class.

Data imbalance can lead to unexpected mistakes and even serious consequences in data analysis, especially in classification tasks. This is because the skewed distribution of class instances forces the classification algorithms to be biased to the majority class. Therefore, the concepts of the minority class are not learned adequately. As a result, the standard classifiers tend to misclassify the majority samples into majority samples when the data is imbalanced, which results in quite a poor classification performance. Though data imbalance has been proved to be a serious problem, it is not addressed well in the standard classification algorithms.

Rahayu, D. S., & Suhartanto, H. (2020) conduct research on “Ensemble Learning in Predicting Financial Distress of Indonesian Public Company” and both authors applied both Random Forest ensemble learning and AdaBoost to Indonesia Public Company data with 6 variables based on Altman Z-Score and one additional variable. Based on the result the accuracy, precision, recall, and f1-score have an average of 91% regardless of the data imbalance and as a gap it is better if more different machine learning model was applied for improved prediction, best feature, and oversampling and under sampling are not selected for perfect prediction.

A study in (Qu, Y., Quan, P., Lei, M., & Shi, I. (2019)) present a Review of bankruptcy prediction using machine learning and deep learning techniques and they used the classical machine learning models and major deep learning methods. As a dataset, they used a new concept called Multiple-Source Heterogeneous Data which include financial statement data, accounting data, textual data, like news or public report even some comments from experts are used as prediction dataset. From the study it is hard to find out which of the inputs has a stronger impact for prediction and it is better if numerical data was selected for bank distress prediction.

Zieba, M., Tomczak, S. K., & Tomczak, J.M. (2016) conduct studies on "Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction" and they use a novel approach for bankruptcy prediction that utilizes Extreme Gradient Boosting for learning an ensemble of decision trees. They used financial condition of Polish companies from 2007 to 2013 for bankrupt and from 2000 to 2012 for non-bankruptcy. As a finding they proposed some extension of the Extreme Gradient Boosting that randomly generates new synthetic features and the gap of study is testing the dataset only by one machine learning model.

In a study conducted by Le et al. (2019), the authors employed a combination of two strategies to solve the issue of class imbalance. On the Korean bankruptcy dataset, the authors used a hybrid strategy that included cost-sensitive learning and an oversampling technique. To begin, an optimal balancing ratio is employed in conjunction with an oversampling module to determine an ideal performance on the validation set. Second, the CBoost method is utilized to predict bankruptcy using a cost-sensitive learning model. In the sample, there were 307 bankrupt firms and 120,048 non-bankrupt firms, resulting in a balancing ratio of 0.0026. Although it has a few drawbacks, the oversampling strategy adds synthetic data to the minority class, enhancing prediction accuracy. The model is likely to be overt because it makes perfect duplicates of existing minority samples. It also expands the number of training examples, lengthening training time and increasing the amount of memory required to hold the training set. The SMOTE-ENN technique was employed for oversampling, which generates synthetic minority samples based on feature similarities between the minority classes. The CBoost algorithm is also used to create majority class clusters. The following approaches were then used to test this approach: Bagging, AdaBoost, Random Forest, and Multilayer Perceptron. AUC and Gmean were utilized as evaluation metrics. The use of oversampling improved the results, although feature selection methods were not used in this study.

Alrasheed et al. (2017) also looked into using oversampling approaches to improve the accuracy of bankruptcy prediction. In this scenario, the minority class was duplicated at random to a particular percentage of the majority class. The top characteristics were chosen using three different feature selection techniques: Mutual Information, Random Forest Genetic Algorithm, and Genetic Algorithm. The results of the aforesaid methodologies were fed into machine learning algorithms such as neural networks, decision trees, logistic regression, K-nearest neighbor, support vector machine, and random forest, which were then evaluated using ROC AUC, F1 score, Precision, and Recall. Overall, the findings revealed that oversampling improves prediction accuracy.

(Goitom Tariku, 2019) conducted research on “Financial Distress and its determinants in the bank and insurance industry” and He applied Altman’s Z-score model as a proxy for financial distress with its fixed effect estimate. To process or analyze data the researcher uses STATA 13 statistical packages. He’s results show that Liquidity and Profitability have positive and significant influences on Z-Score as a proxy of financial distress. In another way, Leverage, Efficiency, and inflation have a negative and significant relation with Z-Score. Other variables such as Firm Size and Economic Growth have no significant impact on the status of a firm’s financial status in Bank and Insurance Companies found in Ethiopia.

(Tadesse Yirgu, 2016) researched “Determinants of Financial Distress: Empirical Evidence from Banks in Ethiopia” and the research was conducted on eight banks' data and analyzed using descriptive statistics and a fixed effect regression model. The model identified capital adequacy, management efficiency, earning ability, and bank size as having negative effects on banking financial distress and except for size all of them appeared significant; whereas asset quality and liquidity appeared as having positive effects, but liquidity was only significant.

(Robel Yohannes, 2018) conducted research on “Determinants of Financial Distress: Empirical Evidence from Private Commercial Banks in Ethiopia” and He applied Altman’s 1993 Z-Score model as the proxy for financial distress and panel data model with its fixed effect estimate. The data was processed using the Eviews 8 statistical package. As a researcher, variables such as capital adequacy, earning ability, liquidity, bank size, and inflation have been found significant influence on the financial distress of private commercial banks in Ethiopia. On the other hand, asset quality, management efficiency, and GDP do not influence financial distress.

2.2.1 Summary of Related works

Following an examination of the relevant literature, it was discovered that the most recent methods for foreseeing bank trouble had been employed. Unfortunately, those techniques are ineffective for creating robust models when the amount of prior data is little and unbalanced.

Additionally, the majority class in these research shows higher categorization accuracy than the minority class. As a result, those studies can serve as the starting point for our research so that the outcomes of the suggested model can be contrasted and confirmed against them. In addition, while much work has been done on the feature selection and class imbalance problems independently, there is little study available on looking at them both at once, particularly in the field of computer science. Therefore, in the area of bank distress prediction, we suggested a combination of feature selection and data sample methodologies.

The following table show some summary of related works.

Table 2. 1*.*Summary of related works

No.	Authors and year	Title	Methodology	Gap	Finding
1.	Rahayu, D. S., & Suhartanto, H. (2020)	Ensemble Learning in Predicting Financial Distress of Indonesian Public Company	Financial statement dataset by using Random Forest ensemble learning and AdaBoost	Use of all features for predictions, Oversampling and under sampling is not applied.	The accuracy, precision, recall, and f1-score have an average of 91%
2.	Qu, Y., Quan, P., Lei, M., & Shi, I. (2019)	Review of bankruptcy prediction using machine learning and deep learning techniques	Financial statement data, accounting data, textual data, news, public report	Non- numerical data, hard to find inputs for predictions	Multiple-Source Heterogeneous Data for prediction
3.	Zieba, M., Tomczak, S. K., & Tomczak, J.M. (2016)	Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction	Polish companies from 2007 to 2013 for bankrupt and from 2000 to 2012 for non-bankruptcy by extension of the Extreme Gradient Boosting	Testing the dataset only by one machine learning model	some extension of the Extreme Gradient Boosting
4.	Alrasheed et al. (2017)	Oversampling approaches to improve the accuracy of bankruptcy prediction	machine learning algorithms such as neural networks, decision trees, logistic regression, K-nearest neighbor, support vector machine, and random forest,	Three different feature selection techniques were selected but not applied	oversampling improves prediction accuracy.
5.	Le et al. (2019)	Combination of two strategies to solve the issue of class imbalance on the Korean bankruptcy dataset	Hybrid strategy that included cost-sensitive learning and an oversampling technique	oversampling strategy adds synthetic data to the minority class, enhancing prediction accuracy and feature selection methods were not used	The use of oversampling improved the results. although

CHAPTER THREE

METHODOLOGY

3.1. INTRODUCTION

The research technique is briefly covered in this part of the paper. The stages or process that the research project will use from beginning to end in order to complete the job that has been presented. This may include, a way used for data collection, data preparation, analysis, and interpretation that shows how the researcher achieves his/her objectives and answers the research questions and also the data collection as well as the data analysis techniques or machine learning algorithm to be used to achieve the objective of the study.

The purpose of this study is to design a bank distress prediction model for the banks and for the study purpose we have used the financial information about Polish companies in the manufacturing sector. The data set included details on both bankrupt companies and still operating ones. After 2004, a large number of Polish manufacturers filed for bankruptcy (Zieba et al., 2016).

3.2. Business Understanding

Insight the business process, defining and framing the problem, defining the goal, and coming to an agreement on success criteria are all crucial components of business knowledge.

Gathering the project's complete objective is the first step before starting any project. Based on different financial indicators of the bank, such as the profit, sales, assets, etc., The main goal of this research is to determine whether a bank is likely to file for distress or not. The goal of this effort is to create a predictive distress model that will be helpful to creditors, management, and employees and that can warn them of any impending distress threat to a bank. Since there are thousands of non-distress banks worldwide but only a small number of distress banks, it is difficult for the model to learn and train itself based on the limited data about the distress cases, as was noted in the earlier literature work.

Second, there are numerous factors not just a few that contribute to distress. Finding the most significant financial factors that push a bank toward distress is another goal of this study. The results of this study will advance the field of distress forecasting and benefit a large number of people worldwide.

3.2. Data Understanding

Data understanding to conduct this research it is important to recognize data touchpoints in relation to research needs, learn where the data comes from, how it is processed, and how choices are made, check if it will be correct and appropriate for the research objectives, check it each data the purpose they are used for and understand each data that are used for the research objectives detail.

It is essential to have a thorough understanding of a prediction model's components before building; otherwise, it is practically impossible to create a solid model. Although the necessary information might be available from several sources, it might be unethical to retrieve it from only some of them. It also requires a lot of effort to properly obtain the data from a reliable source because the data from certain sources could not be reliable.

We need the financial ratios of businesses in a specific location over a specific period, including distress and non-distress cases, to move forward with this research. Consequently, the dataset and data source used is explained below:

UCI Machine Learning Repository: The selected dataset, which is hosted for free by the UCI Machine Learning Repository, consists of Polish enterprises. Poland is the sixth-largest economy in the European Union, and according to a 2015 McKinsey assessment, Poland will replace Russia as Europe's new growth engine by 2025. The dataset consists of 5910 instances, or financial statements, of which 5500 instances represent non-bankrupt enterprises and 410 instances represent companies that have filed for bankruptcy. Different classification cases have been identified based on the predicting period.

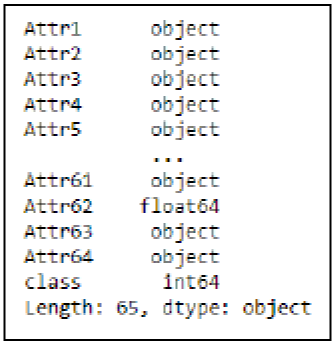
The data for the 5th year, which we will be using, include the financial statements from that year as well as the proper class label, which indicates the status of distress after one year. Net profit, liabilities, working capital, EBIT, and other numerical qualities are among the 64 ratio-based numerical attributes in total.

3.2.1 Data Exploration

In Machine Learning, proper data exploration is the major critical task that has a significant effect on the learning prediction's correctness since data exploration is the process of describing the data employing using statistical and graphical methods to highlight key components of the data for additional examination. Under data exploration, the following major tasks will be done: Data feature variable exploration, Data collection, management of outliers, dimensionality reduction, and data redundancy resolution.

Utilizing visuals is one of the quickest ways to examine and comprehend the data. When it comes to understanding the data's structure, the distribution of its values, and the existence of any corrections within the dataset, the results of visual data exploration can be quite effective. The results of the Polish bankruptcy dataset are as follows:

- Determining the datatype for each variable in the dataset. As part of data pre-processing, the majority of the variables need to be changed from their object datatype to float datatype.



```
Attr1      object
Attr2      object
Attr3      object
Attr4      object
Attr5      object
            ...
Attr61     object
Attr62    float64
Attr63     object
Attr64     object
class       int64
Length: 65, dtype: object
```

Figure 3 1. Data types of variables

- Verifying whether the dataset contains any null values. Only one column in the dataset has more than 40% missing values, whereas the other columns have relatively few missing values that may be substituted by the column's mean value.
- A significant class imbalance can be seen when comparing the distribution of distressed and non-distress instances because the dataset contains 5500 non-distress enterprises and only 410 distressed firms. Prior to the implementation of the suggested model, this problem must be resolved.
- The multicollinearity and correlation between the dependent variable and the independent variables. The dependent variable in this scenario, which indicates the bank's state of crisis, is the class variable.

3.4. Data Preprocessing and Transformation

To provide accurate and high-quality results from the model, data preparation is a crucial step in the data mining process. Big data should be treated carefully to ensure that important information is kept and not altered throughout the data preparation phase. Big data frequently contains noise, and the existence of special characters, missing values, and blank spaces frequently affects the performance of the model. Some of the variables may be utterly meaningless, and only a small group of traits may be more essential and responsible for predicting the target variable (Kotsiantis et al., 2006). The data preprocessing stage is frequently regarded as the most difficult and time-consuming stage since having redundant and unnecessary variables would raise computational time and expense. The steps undertaken to prepare the data required for this study are listed below:

- Converting the file type: The data file was initially in an ARFF format after being extracted from the source. It has to be converted to a CSV format to satisfy the demands of the proposed model. The .arff file was changed into a CSV file using Python code.
- Changing the datatype of all variables: The datatypes of all the variables are typically seen once the dataset has been sourced. All of the variables' values were numerical with various datatypes. These variables were converted to Float datatype.
- Handling the missing values or NA's: For the proposed model presence of missing values is troublesome and can hinder model performance. The proportion of missing values in each column has been noticed after these values were changed to NAs to deal with them. Based on this, the mean of the column was used to impute missing values for columns that had a low percentage of missing values. It was necessary to fully eliminate from the dataset the column whose values were missing in approximately half of the cases.
- Multicollinearity: it is known that these variables are influencing the target variables and the correlation between the variables by using a correlation test. It can be argued that variables with high correlations (p values more than 0.90) are redundant data that are equally helpful in predicting the target variable. It is best to keep the other variable and eliminate one of the variables. 26 of the 64 predictor variables in the dataset were significantly linked with one another, necessitating their removal from the study.

3.4.1 Feature Selection

Feature selection is one method that can further enhance the performance of the suggested model by cleansing the data and eliminating pointless and superfluous variables. The feature space is compressed with this technique, which might be helpful for the model. Benefits could include increased accuracy, decreased over-treating risk, quicker computation times and better model explain ability. When the dataset has too many features, explain ability is lost (Liu and Motoda, 1998).

Table 3. 1. Various Features Selected

Attributes Name	Attributes Description
Attr58	Total costs / total sales
Attr39	Profit on sales/sales
Attr42	Profit on operating activities/sales
Attr24	Gross profit (in 3 years) / total assets
Attr41	Total liabilities / ((profit on operating activities + depreciation) * (12/365))
Attr27	Profit on operating activities / financial expenses
Attr21	Sales (n) / sales (n-1)

One of the preprocessing steps used before creating a classification model is feature selection, which addresses the curse of dimensionality issue that has a negative impact on the algorithm. A small number of the 64 features in the dataset utilized in this study may not help predict bankruptcy. In this study, the best characteristics were chosen using the Random Forest feature selection technique, which also helped to eliminate any extraneous features. The effectiveness of the tree-based strategies used in random forests is measured by how well they can increase node purity. This is referred to as a mean decrease in impurity or Gini impurity. At the beginning of the tree, there is the highest loss in node purity, but at the conclusion, there is the least loss.

The following table describes the set of features considered in the classification process. Source (Zieba,M., Tomczak, S. K., & Tomczak, J.M. (2016))

Table 3. 2. Bank Distress Features

ID	Description	ID	Description
Attr1	net profit / total assets	Attr33	operating expenses / short-term liabilities
Attr2	total liabilities / total assets	Attr34	operating expenses / total liabilities
Attr3	working capital / total assets	Attr35	profit on sales / total assets
Attr4	current assets / short-term liabilities	Attr36	total sales / total assets
Attr5	[(cash + short-term securities + receivables - short-term liabilities) / (operatingexpenses - depreciation)] * 365,	Attr37	(Current assets - inventories) / Long-term liabilities
Attr6	retained earnings / total assets	Attr38	constant capital / total assets
Attr7	EBIT / total assets	Attr39	profit on sales/sales
Attr8	book value of equity / total liabilities	Attr40	(Current assets - inventory - receivables) / short-term liabilities
Attr9	sales / total assets	Attr41	total liabilities / ((profit on operating activities + depreciation) * (12/365))
Attr10	equity / total assets	Attr42	profit on operating activities/sales
Attr11	(Gross profit + extraordinary items + financial expenses) / total assets	Attr43	rotation receivables + inventory turnover in days
Attr12	gross profit / short-term liabilities	Attr44	(receivables * 365) / sales
Attr13	(Gross profit + depreciation) / sales	Attr45	net profit/inventory
Attr14	(Gross profit + interest) / total assets	Attr46	(Current assets - inventory) / short-term liabilities
Attr15	(Total liabilities * 365) / (gross profit + depreciation)	Attr47	(inventory * 365) / cost of products sold
Attr16	(Gross profit + depreciation) / total liabilities	Attr48	EBITDA (profit on operating activities - depreciation) / total assets
Attr17	total assets / total liabilities	Attr49	EBITDA (profit on operating activities - depreciation) / sales
Attr18	gross profit / total assets	Attr50	current assets / total liabilities
Attr19	gross profit/sales	Attr51	short-term liabilities / total assets
Attr20	(inventory * 365) / sales	Attr52	(short-term liabilities * 365) / cost of products sold)
Attr21	sales (n) / sales (n-1)	Attr53	equity / fixed assets
Attr22	profit on operating activities / total assets	Attr54	constant capital / fixed assets
Attr23	net profit/sales	Attr55	working capital
Attr24	gross profit (in 3 years) / total assets	Attr56	(Sales - the cost of products sold) / sales
Attr25	(Equity - share capital) / total assets	Attr57	(Current assets - inventory - short-term liabilities) / (sales - gross profit - depreciation)
Attr26	(Net profit + depreciation) / total liabilities	Attr58	total costs /total sales
Attr27	profit on operating activities / financial expenses	Attr59	long-term liabilities/equity
Attr28	working capital / fixed assets	Attr60	sales/inventory
Attr29	the logarithm of total assets	Attr61	sales/receivables
Attr30	(Total liabilities - cash) / sales	Attr62	(short-term liabilities *365) / sales
Attr31	(Gross profit + interest) / sales	Attr63	sales / short-term liabilities
Attr32	(Current liabilities * 365) / cost of products sold	Attr64	sales / fixed assets

The features considered in the research studies are described in detail in the table above and based on the collected data five classification classes were identified, that depends on the predicting period:

- 1st Year: - the data contains financial rates from 1st year of the predicting period and a corresponding class label that indicates distress status after 5 years.
- 2nd Year: - the data contains financial rates from the 2nd year of the predicting period and the corresponding class label that indicates distress status after 4 years.
- 3rd Year: - the data contains financial rates from the 3rd year of the predicting period and the corresponding class label that indicates distress status after 3 years.
- 4th Year: - the data contains financial rates from the 4th year of the predicting period and the corresponding class label that indicates distress status after 2 years.
- 5th Year: - the data contains financial rates from the 5th year of the predicting period and the corresponding class label that indicates distress status after 1 year.

The description of the data about the five different classification tasks present in this dataset was as follows.

Dataset	*Features From*	*Distress After*	*No. Distress*	*No. Not Distress*	*Total*
1st Year	1st Year	5th Year	271	6,756	7,027
2nd Year	2nd Year	4th Year	400	9,773	10,173
3rd Year	3rd Year	3rd Year	495	10,008	10,503
4th Year	4th Year	2nd Year	515	9,277	9,792
5th Year	5th Year	1st Year	410	5,500	5,910

3.4.2 Tackling Imbalanced Data

One of the frequent issues with classification modeling is the existence of uneven class distribution (Elrahman and Abraham; 2013). This indicates that there is a significant disparity in the number of observations made by one class and the other class. Because most machine learning algorithms are built to function best when both classes are equally balanced, this is a difficult problem. The created predictive model may out to be biased and erroneous in the situation of class imbalance. This might make it more likely that the minority class will be incorrectly classified.

A sampling-based strategy includes either an oversampling of the minority class, an under sampling of the majority class, or a hybrid technique that combines both. In this study, we will address the issue of class imbalance utilizing a novel hybrid strategy called SMOTEENN (Monard; 2017). This method is appropriate for this issue because the dataset we working with is highly unbalanced; on the one hand, it is necessary to increase the minority class, and on the other hand, because the majority class contains a large number of instances, it is feasible to partially under-sample the majority class.

The Synthetic Minority Oversampling Technique (SMOTE) uses nearest neighbors, which are determined using the Euclidean distance between the data points, to artificially produce new minority instances between the true minority class. When a class label is different from its two closest neighbors, ENN isolates those occurrences. The occurrences are then eliminated using the ENN approach based on the KNN prediction. Only those examples that differ from the majority class in terms of prediction are eliminated.

3.9. The Proposed Model

Figure 3.2 shows the proposed bank distress prediction model initially, we collected the data from the source and then underwent preprocessing methods including deleting unnecessary columns and impute NAs using mean values. Additionally, we have used the most crucial features and removing the remainder using the feature selection method (i.e., Random Forest). Then split the dataset into trains and tested shadowed by a hybrid resampling technique, SMOTE to resample the dataset. Finally, five distinct classifiers are fed the processed data, and the performance of each is assessed using the test data.

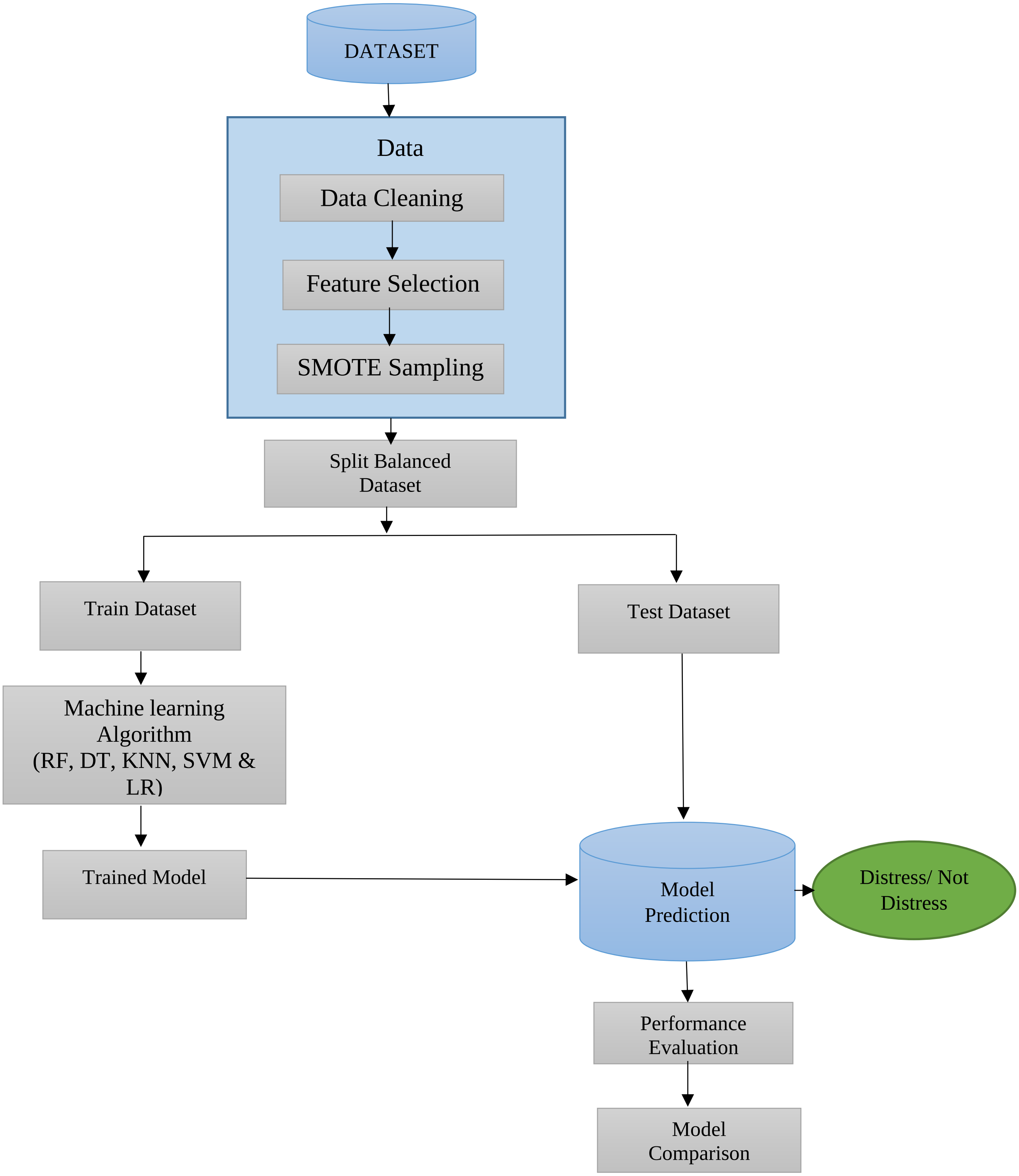


Figure 3 2.The architecture of the proposed model

Dataset: - the Polish Bankruptcy dataset.

Processing Dataset: - The collected row dataset should be processed due to three reasons, those are for fixing missing values, imbalanced problem and normalize the dataset

For training 70 %, and for test 30% of Datasets

SVM, DT, RF, LG, and KNN: - are methods or algorithm training applied in each method and for each the result of accuracy.

Final Prediction Result: - the result of the predicted bank distress

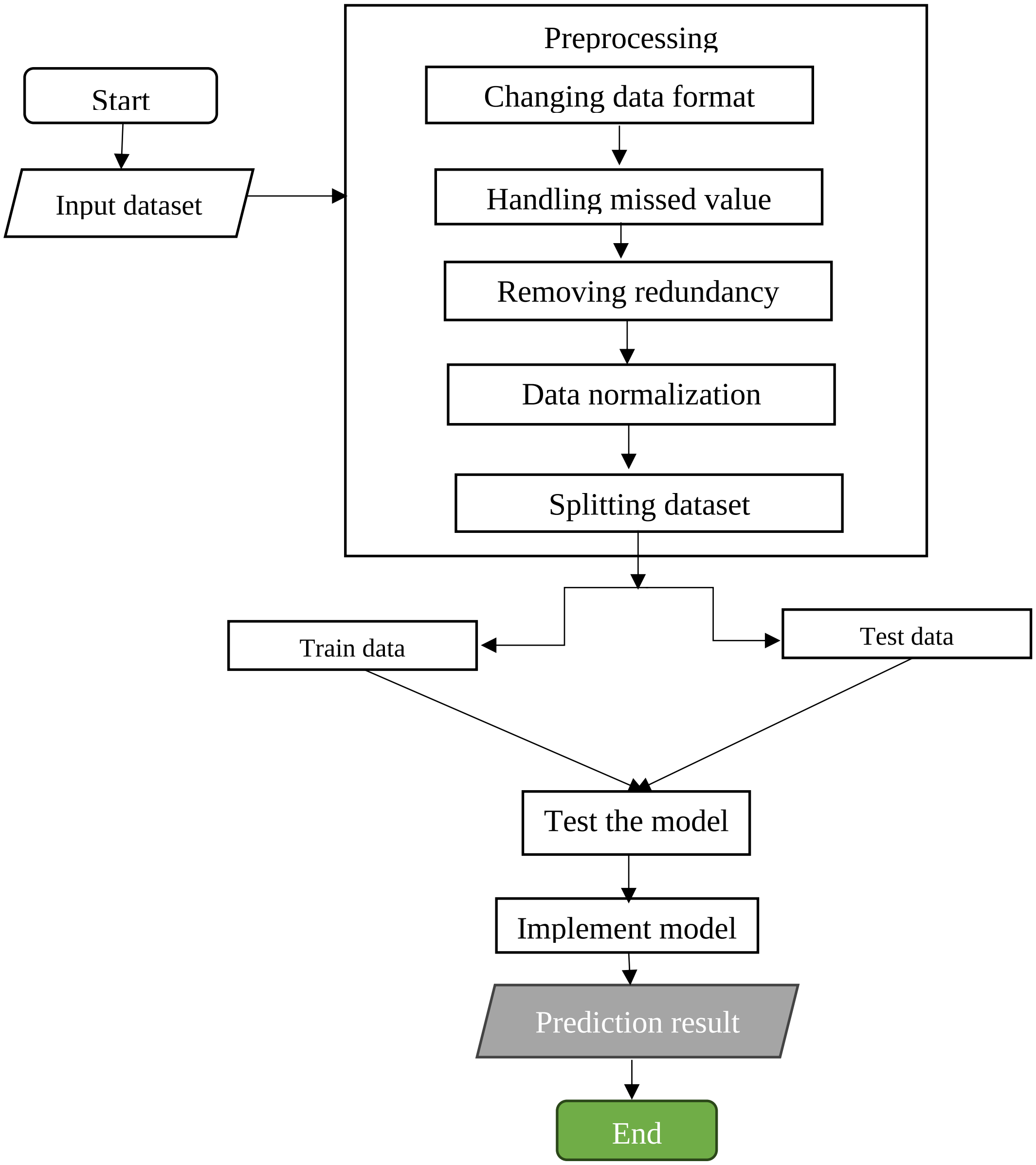


Figure 3 3.Process flow diagram

3.11. Tools

In this section, we discussed the tools that we have used during developing the proposed model.

3.11.1. Python

Python is an interpreted, object-oriented, general-purpose, high-level programming language with dynamic semantics (Rossum, 2020). Its data structures, in conjunction with dynamic type and dynamic binding, make it particularly appealing for Rapid Application Development (RAD). It is simple and easy to learn a programming language. Python has support for modules and packages, which promotes program modularity and code reuse (Huenerfauth et al., 2009). And python has many good features among that are:

- Easy to code
- Free and open source
- It follows an object-oriented approach
- Portable
- High-level language
- It has a large and standard library

3.11.2. Anaconda

Anaconda is a tool that is used to develop machine learning, deep learning, and artificial intelligence. It is a Python and R-programming distribution used for data analysis and scientific computing. It is an open-source project developed by Continuum Analytics, and it can be run on Windows, Mac OS X, and Linux. Anaconda consists of many packages like Numpy, Scipy, Matplotlib, Pandas, IPython, and Synthon (Weston and Bjornson, 2016). Anaconda delivers the tools needed to easily:

- Import data from databases, files, and data lakes
- Manage environments with conda
- Share, collaborate on, and reproduce projects
- Deploy projects into production with a single click of a button

We have used an anaconda navigator to develop the proposed prediction model. It is a desktop and graphical user interface-based application.

3.11.3. Pandas

It is a software library written in Python programming language for data manipulation and analysis. Pandas provides data structures and procedures for working with numerical tables and time series. Pandas can be used to deal with statistical data, import CSV files, data alignment, handle missing values, and generally for data preprocessing (Mckinney, 2010). Some of Pandas' features are:

- Data Frame object for data operation with combined indexing.
- Tools for reading and writing data or importing datasets.
- Data alignment and integrated handling of missing data.
- Reshaping and pivoting of data sets.
- Data structure column insertion and deletion.
- Group by engine permitting split-apply-combine processes on datasets.
- Hierarchical axis indexing to work with dimensional data in a lower-dimensional data structure.

3.11.4 NumPy

It is Python's foundational and all-purpose scientific computing library. It is an abbreviation for 'Numerical Python'. It is a Python library that offers a multidimensional array object, several derivative objects (such as arrays and matrices), and a collection of functions for performing quick array operations, including mathematical, logical, shape manipulation, sorting, selecting, input/output, linear algebra, statistical operations, random simulation and the like (Oliphant, 2006). Some of the features of NumPy are:

- Tools for integrating another programming language
- Sophisticated functions
- A powerful N-dimensional array object
- Useful for linear algebra and random number capabilities

3.11.5. Jupyter Notebook

JupyterLab is the most recent web-based interactive development environment for notebooks, code, and data. Users may create and arrange workflows in data science, scientific computing, computational journalism, and machine learning using its versatile interface. Extensions to enhance and improve functionality are encouraged by a modular architecture.

The original web application for producing and sharing computational documents is Jupyter Notebook. It provides a straightforward, simplified, document-centric interface.

CHAPTER FOUR

RESULT AND DISCUSSION

4.1 INTRODUCTION

In this chapter, we described the experimental results and evaluation of the experimental result of the proposed model for the prediction of distress in the bank. The experiment was especially focused on how different machine learning techniques can be used for bank distress prediction and also an experimental bench implemented in the Python programming language. The experimental evaluation of the proposed model for the bank distress prediction model is described in detail. Experimental results are evaluated using the ROC-score curve, confusion matrix, and statistical performance that approves the perception of the proposed class imbalance distress prediction model are described. The experimental dataset used, their characteristics, and the implementation of the proposed distress prediction model is also described thoroughly. And also, the effect of sampling techniques and feature selection are evaluated and compared before and after these techniques are applied.

4.2 Dataset

The dataset used for this study is collected from Polish companies for distress prediction and hosted by the UCI Machine Learning repository for free. According to a 2015 McKinsey report, Poland has the sixth-largest economy in the European Union, which is why Polish businesses were chosen. The dataset consists of 5910 instances i.e., financial statements within 5500 instances represented the non-distress companies and 410 instances represented the distress companies. Five distinct categorization situations have been identified based on the predicting period. The 5th year data set includes financial statements from the forecasting period's fifth year, as well as the applicable class designation, which reflects the level of distress after one year. In all, 64 numerical characteristics ratios are obtained from net profit, liabilities, working capital, EBIT, and so on. To select the above datasets, we have identified two reasons. First, these datasets have a high-class imbalance problem which is an unequal distribution of the majority and minority classes. Second, the other study on this dataset recommends that the class imbalance issue still needs more findings.

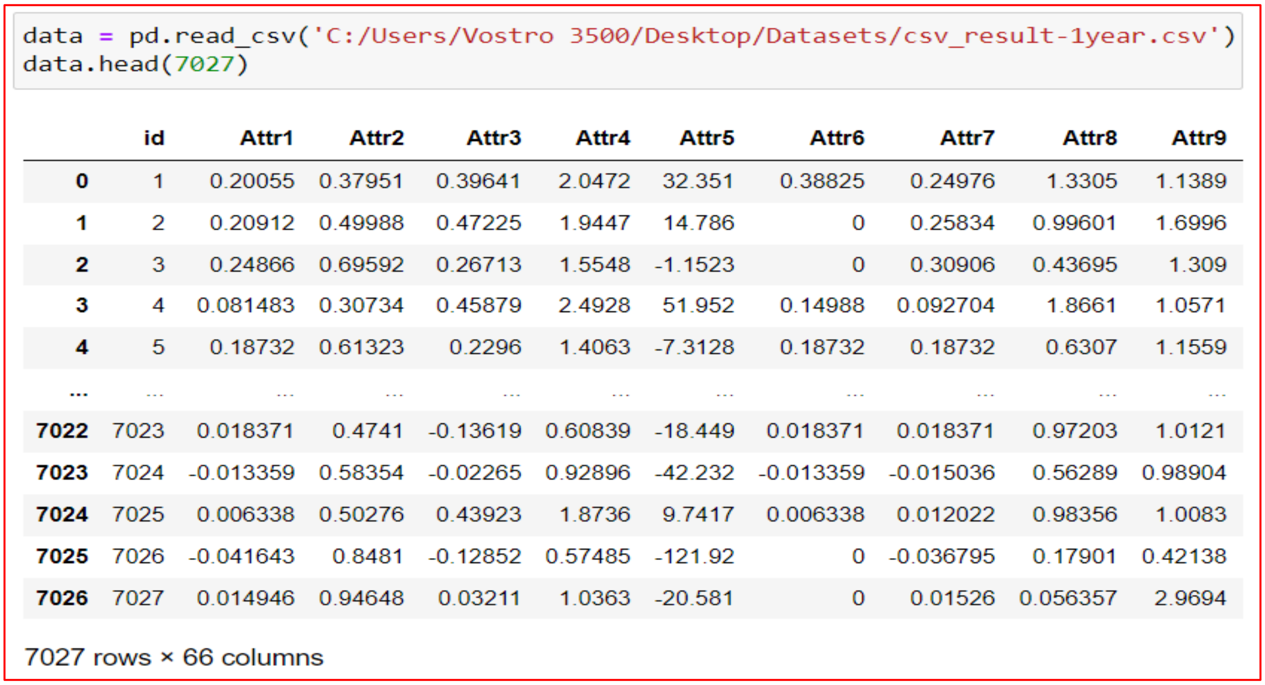
4.2.1 Dataset Details

The table below shows a detailed summary of the datasets. It contains five years of data and each year dataset contains 64 attributes (i.e., financial indicators) and 1 target class or label.

Table 4. 1. Experimental Dataset

Dataset	*Features from*	*Distress After*	*No. Distress*	*No. not Distress*	*Total*
1st Year	1st Year	5 years	271	6,756	7,027
2nd Year	2nd Year	4 years	400	9,773	10,173
3rd Year	3rd Year	3 years	495	10,008	10,503
4th Year	4th Year	2 years	515	9,277	9,792
5th Year	5th Year	1 year	410	5,500	5,910

Table 4. 2. 1st Year dataset values

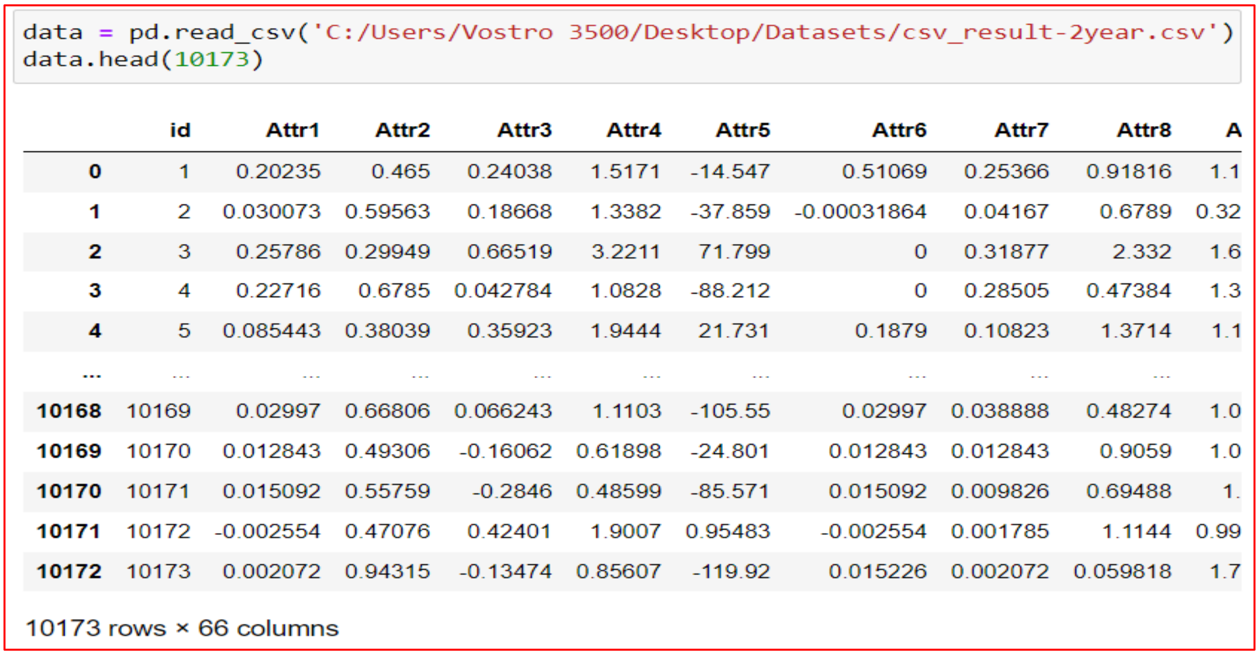


```
data = pd.read_csv('C:/Users/Vostro 3500/Desktop/Datasets/csv_result-1year.csv')
data.head(7027)
```

	id	Attr1	Attr2	Attr3	Attr4	Attr5	Attr6	Attr7	Attr8	Attr9
0	1	0.20055	0.37951	0.39641	2.0472	32.351	0.38825	0.24976	1.3305	1.1389
1	2	0.20912	0.49988	0.47225	1.9447	14.786	0	0.25834	0.99601	1.6996
2	3	0.24866	0.69592	0.26713	1.5548	-1.1523	0	0.30906	0.43695	1.309
3	4	0.081483	0.30734	0.45879	2.4928	51.952	0.14988	0.092704	1.8661	1.0571
4	5	0.18732	0.61323	0.2296	1.4063	-7.3128	0.18732	0.18732	0.6307	1.1559
...	...	...	...	...	...	...	...	...	...	...
7022	7023	0.018371	0.4741	-0.13619	0.60839	-18.449	0.018371	0.018371	0.97203	1.0121
7023	7024	-0.013359	0.58354	-0.02265	0.92896	-42.232	-0.013359	-0.015036	0.56289	0.98904
7024	7025	0.006338	0.50276	0.43923	1.8736	9.7417	0.006338	0.012022	0.98356	1.0083
7025	7026	-0.041643	0.8481	-0.12852	0.57485	-121.92	0	-0.036795	0.17901	0.42138
7026	7027	0.014946	0.94648	0.03211	1.0363	-20.581	0	0.01526	0.056357	2.9694

7027 rows × 66 columns

Table 4. 3.2nd Year dataset values

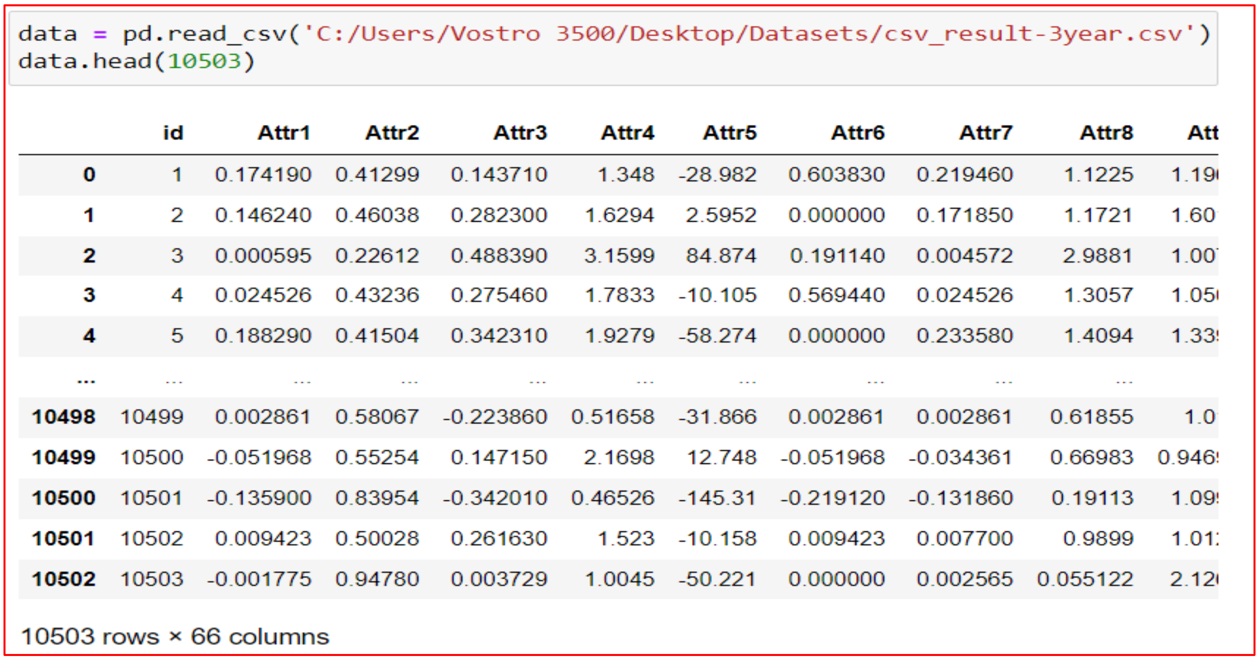


```
data = pd.read_csv('C:/Users/Vostro 3500/Desktop/Datasets/csv_result-2year.csv')
data.head(10173)
```

	id	Attr1	Attr2	Attr3	Attr4	Attr5	Attr6	Attr7	Attr8	A
0	1	0.20235	0.465	0.24038	1.5171	-14.547	0.51069	0.25366	0.91816	1.1
1	2	0.030073	0.59563	0.18668	1.3382	-37.859	-0.00031864	0.04167	0.6789	0.32
2	3	0.25786	0.29949	0.66519	3.2211	71.799	0	0.31877	2.332	1.6
3	4	0.22716	0.6785	0.042784	1.0828	-88.212	0	0.28505	0.47384	1.3
4	5	0.085443	0.38039	0.35923	1.9444	21.731	0.1879	0.10823	1.3714	1.1
...	...	...	...	...	...	...	...	...	...	
10168	10169	0.02997	0.66806	0.066243	1.1103	-105.55	0.02997	0.038888	0.48274	1.0
10169	10170	0.012843	0.49306	-0.16062	0.61898	-24.801	0.012843	0.012843	0.9059	1.0
10170	10171	0.015092	0.55759	-0.2846	0.48599	-85.571	0.015092	0.009826	0.69488	1.
10171	10172	-0.002554	0.47076	0.42401	1.9007	0.95483	-0.002554	0.001785	1.1144	0.99
10172	10173	0.002072	0.94315	-0.13474	0.85607	-119.92	0.015226	0.002072	0.059818	1.7

10173 rows × 66 columns

Table 4. 4.3rd Year dataset values

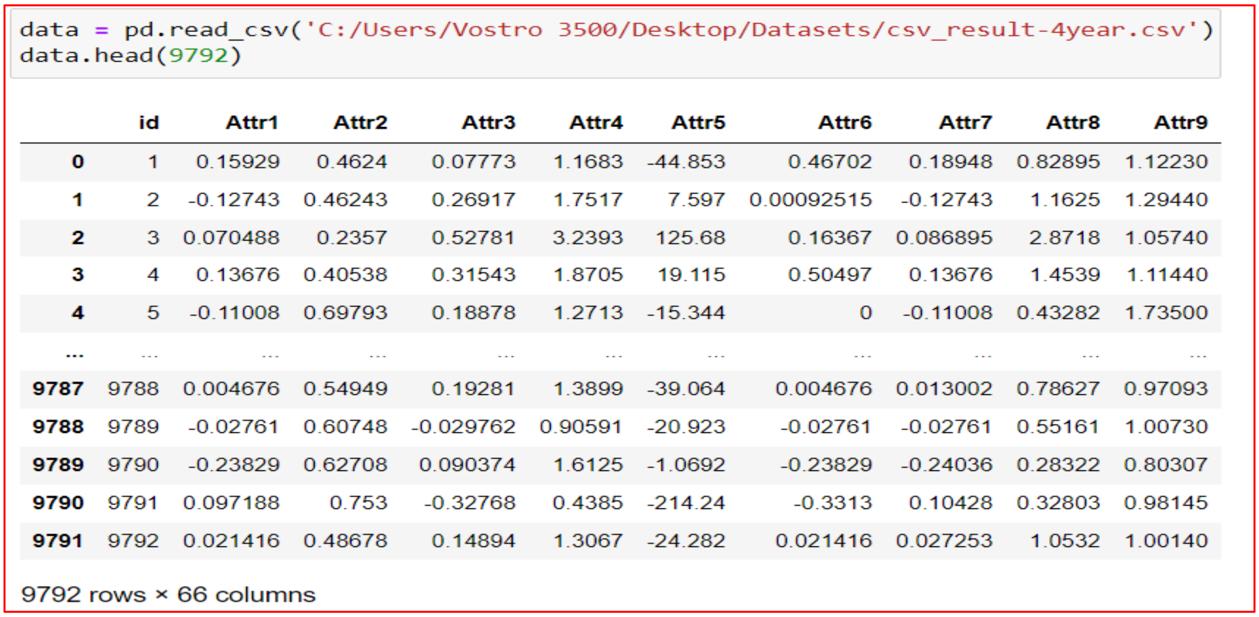


```
data = pd.read_csv('C:/Users/Vostro 3500/Desktop/Datasets/csv_result-3year.csv')
data.head(10503)
```

	id	Attr1	Attr2	Attr3	Attr4	Attr5	Attr6	Attr7	Attr8	Att
0	1	0.174190	0.41299	0.143710	1.348	-28.982	0.603830	0.219460	1.1225	1.19
1	2	0.146240	0.46038	0.282300	1.6294	2.5952	0.000000	0.171850	1.1721	1.60
2	3	0.000595	0.22612	0.488390	3.1599	84.874	0.191140	0.004572	2.9881	1.00
3	4	0.024526	0.43236	0.275460	1.7833	-10.105	0.569440	0.024526	1.3057	1.05
4	5	0.188290	0.41504	0.342310	1.9279	-58.274	0.000000	0.233580	1.4094	1.33
...	...	...	...	...	...	...	...	...	...	
10498	10499	0.002861	0.58067	-0.223860	0.51658	-31.866	0.002861	0.002861	0.61855	1.0
10499	10500	-0.051968	0.55254	0.147150	2.1698	12.748	-0.051968	-0.034361	0.66983	0.946
10500	10501	-0.135900	0.83954	-0.342010	0.46526	-145.31	-0.219120	-0.131860	0.19113	1.09
10501	10502	0.009423	0.50028	0.261630	1.523	-10.158	0.009423	0.007700	0.9899	1.01
10502	10503	-0.001775	0.94780	0.003729	1.0045	-50.221	0.000000	0.002565	0.055122	2.12

10503 rows × 66 columns

Table 4. 5.4th Year dataset values

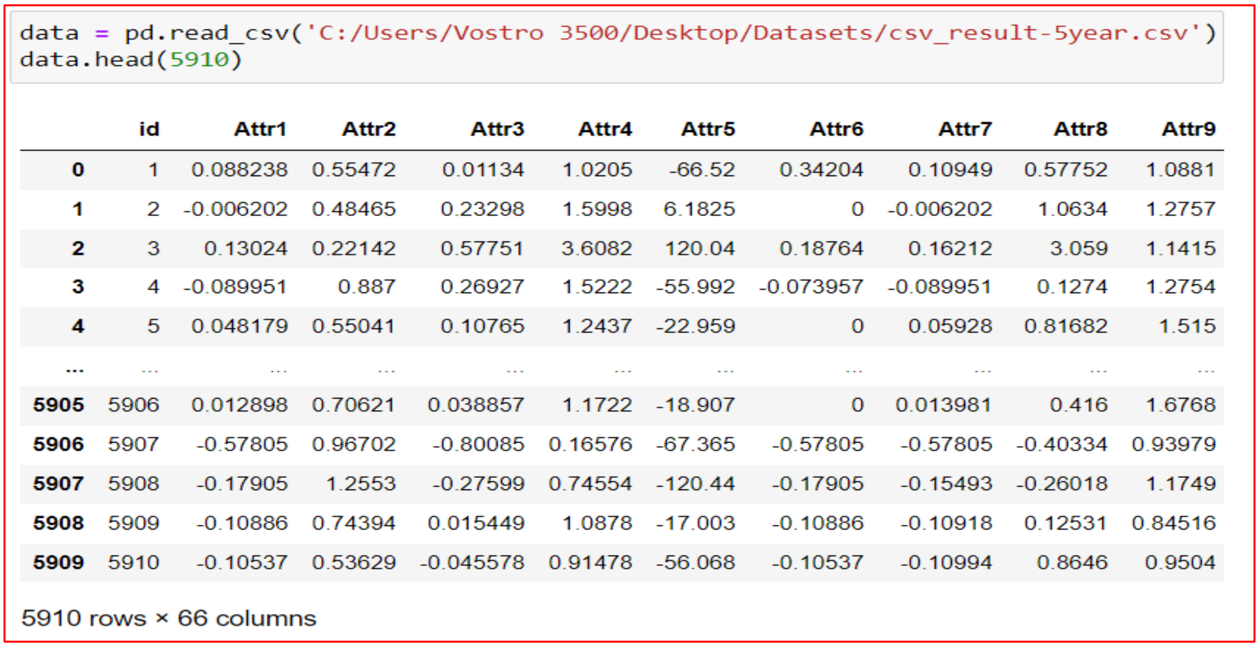


```
data = pd.read_csv('C:/Users/Vostro 3500/Desktop/Datasets/csv_result-4year.csv')
data.head(9792)
```

	id	Attr1	Attr2	Attr3	Attr4	Attr5	Attr6	Attr7	Attr8	Attr9
0	1	0.15929	0.4624	0.07773	1.1683	-44.853	0.46702	0.18948	0.82895	1.12230
1	2	-0.12743	0.46243	0.26917	1.7517	7.597	0.00092515	-0.12743	1.1625	1.29440
2	3	0.070488	0.2357	0.52781	3.2393	125.68	0.16367	0.086895	2.8718	1.05740
3	4	0.13676	0.40538	0.31543	1.8705	19.115	0.50497	0.13676	1.4539	1.11440
4	5	-0.11008	0.69793	0.18878	1.2713	-15.344	0	-0.11008	0.43282	1.73500
...	...	...	...	...	...	...	...	...	...	...
9787	9788	0.004676	0.54949	0.19281	1.3899	-39.064	0.004676	0.013002	0.78627	0.97093
9788	9789	-0.02761	0.60748	-0.029762	0.90591	-20.923	-0.02761	-0.02761	0.55161	1.00730
9789	9790	-0.23829	0.62708	0.090374	1.6125	-1.0692	-0.23829	-0.24036	0.28322	0.80307
9790	9791	0.097188	0.753	-0.32768	0.4385	-214.24	-0.3313	0.10428	0.32803	0.98145
9791	9792	0.021416	0.48678	0.14894	1.3067	-24.282	0.021416	0.027253	1.0532	1.00140

9792 rows × 66 columns

Table 4. 6.5th Year dataset values



```
data = pd.read_csv('C:/Users/Vostro 3500/Desktop/Datasets/csv_result-5year.csv')
data.head(5910)
```

	id	Attr1	Attr2	Attr3	Attr4	Attr5	Attr6	Attr7	Attr8	Attr9
0	1	0.088238	0.55472	0.01134	1.0205	-66.52	0.34204	0.10949	0.57752	1.0881
1	2	-0.006202	0.48465	0.23298	1.5998	6.1825	0	-0.006202	1.0634	1.2757
2	3	0.13024	0.22142	0.57751	3.6082	120.04	0.18764	0.16212	3.059	1.1415
3	4	-0.089951	0.887	0.26927	1.5222	-55.992	-0.073957	-0.089951	0.1274	1.2754
4	5	0.048179	0.55041	0.10765	1.2437	-22.959	0	0.05928	0.81682	1.515
...	...	...	...	...	...	...	...	...	...	...
5905	5906	0.012898	0.70621	0.038857	1.1722	-18.907	0	0.013981	0.416	1.6768
5906	5907	-0.57805	0.96702	-0.80085	0.16576	-67.365	-0.57805	-0.57805	-0.40334	0.93979
5907	5908	-0.17905	1.2553	-0.27599	0.74554	-120.44	-0.17905	-0.15493	-0.26018	1.1749
5908	5909	-0.10886	0.74394	0.015449	1.0878	-17.003	-0.10886	-0.10918	0.12531	0.84516
5909	5910	-0.10537	0.53629	-0.045578	0.91478	-56.068	-0.10537	-0.10994	0.8646	0.9504

5910 rows × 66 columns

4.3. Implementation

The process of choosing the most crucial characteristics and resampling the dataset is covered in length in this part, along with the implementation of the suggested models for predicting bank distress. The entire implementation process was carried out using Anaconda Open-Source software and the Jupyter Notebook (v.6.4.5), which is an open-source Python computing offering. Python was selected for the implementation phase because it has a large online support community, ranks highest for readability of code, and is simple to use.

Based on the forecasted period, five separate files made up the data source for this study, and the fifth file was selected for implementation. Financial rates from the fifth year were included in the dataset, along with class labels showing the state of bank distress after a year. The supplied files were in ARFF format, however there was online Python code that could be used to change the format of the file to CSV. The dataset was then examined for missing values after being imported into Python as a Data frame. Using the pandas profiling package, the special character ('?') in place of the missing value was changed to NAs, and certain columns were investigated. Following the fundamental cleaning process, a feature selection strategy needs to be chosen in order to narrow down the features and pick just the finest ones. The choose from the Model package from the Sklearn feature selection library was used to implement the random forest feature selection strategy.

A high-class imbalance was discovered while investigating the data. After the data was separated into classes, we utilized the SMOTE approach to balance the classes in order to prevent overfitting and create a stable model. We employed a hybrid strategy that can oversample the minority class and under sample the majority class because of the large imbalance. Resampling was performed using the SMOTE package from the imbalance learns library. Finally, the dataset is split into training and testing datasets, with 30% of the dataset chosen for testing and 70% of the dataset given to model training.

4.4. Experiment

This section describes the experimental study carried out for Bank distress prediction performance analysis using Polish bankruptcy datasets. The performance of these datasets is studied using feature selection and synthetic minority over sampling techniques.

4.4.1. Data Cleaning

We employed a filter threshold approach that recognizes the multicollinearity and correlation between the dependent variable and the independent variables in order to detect and eliminate these issues since the dataset contains missing values, duplicate occurrences, and correlated features. The 'class' variable in this case is the dependent variable which denotes the distress status of the bank. This is done using sklearn in the Python library. These correlated and redundant features are selected and removed since they affect the classification performance.

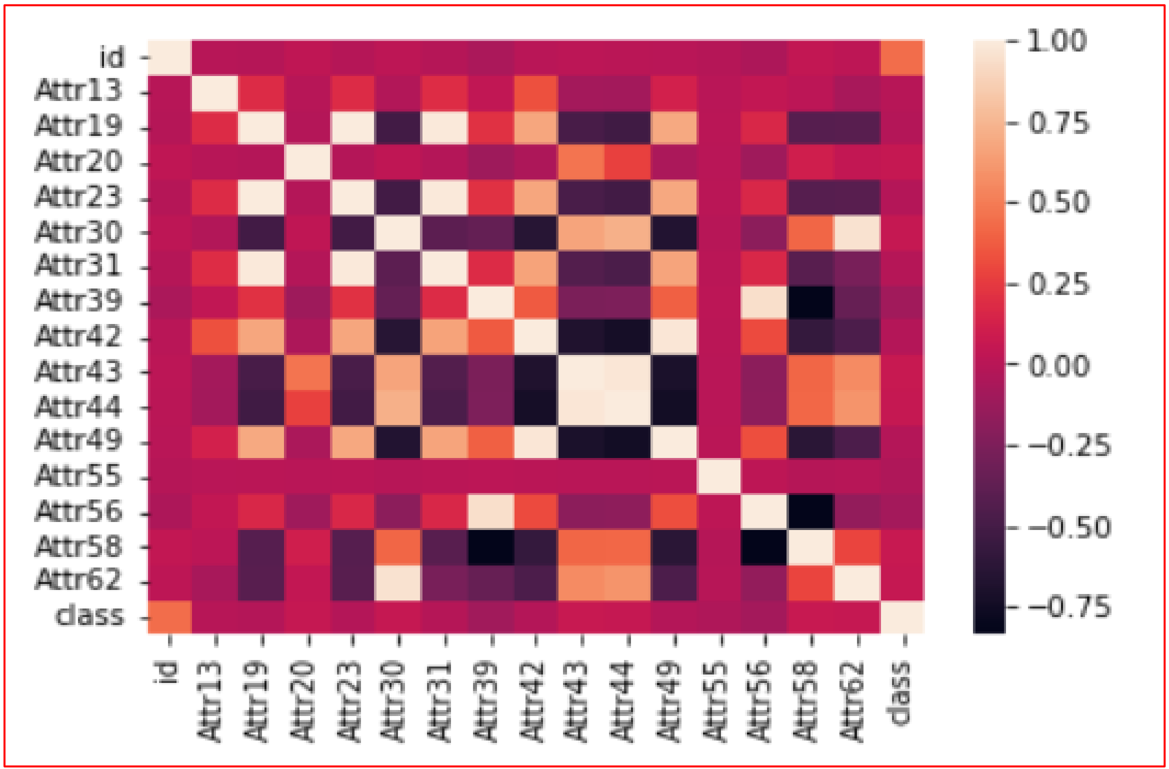
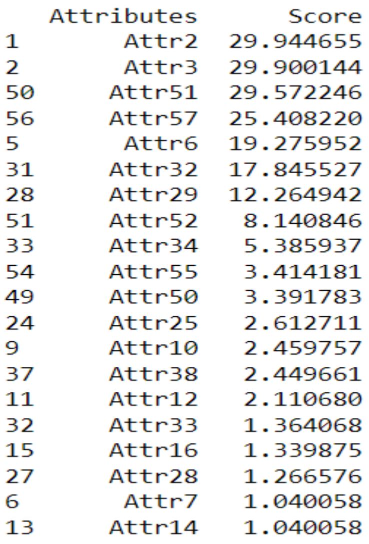


Figure 4 1. Correlation Matrix

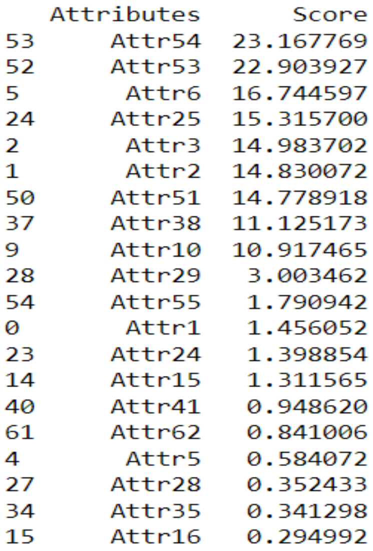
4.4.2 Feature Selection

In the experiments, we determine the associational relationship among the Bank distress attributes by identifying the most effective by giving those scores and considering their effect on distress proneness. While performing the feature selection process, there are two steps. The first one is, to calculate each individual score of the attribute with their values corresponding to the desired output. As shown in the following table, the score of an individual attribute of all five years is listed.



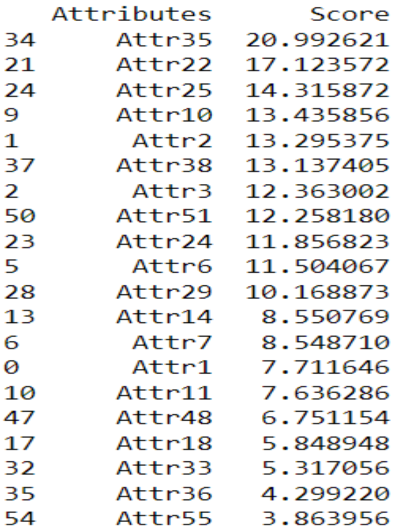
```
   Attributes      Score
1       Attr2  29.944655
2       Attr3  29.900144
50     Attr51  29.572246
56     Attr57  25.408220
5       Attr6  19.275952
31     Attr32  17.845527
28     Attr29  12.264942
51     Attr52   8.140846
33     Attr34   5.385937
54     Attr55   3.414181
49     Attr50   3.391783
24     Attr25   2.612711
9      Attr10   2.459757
37     Attr38   2.449661
11     Attr12   2.110680
32     Attr33   1.364068
15     Attr16   1.339875
27     Attr28   1.266576
6       Attr7   1.040058
13     Attr14   1.040058
```

1st year selected Features



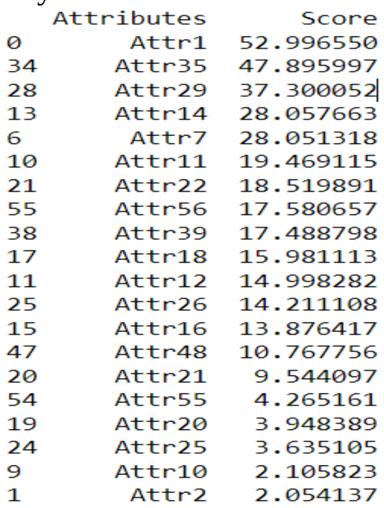
```
   Attributes      Score
53     Attr54  23.167769
52     Attr53  22.903927
5       Attr6  16.744597
24     Attr25  15.315700
2       Attr3  14.983702
1       Attr2  14.830072
50     Attr51  14.778918
37     Attr38  11.125173
9      Attr10  10.917465
28     Attr29   3.003462
54     Attr55   1.790942
0       Attr1   1.456052
23     Attr24   1.398854
14     Attr15   1.311565
40     Attr41   0.948620
61     Attr62   0.841006
4       Attr5   0.584072
27     Attr28   0.352433
34     Attr35   0.341298
15     Attr16   0.294992
```

2nd year selected Features



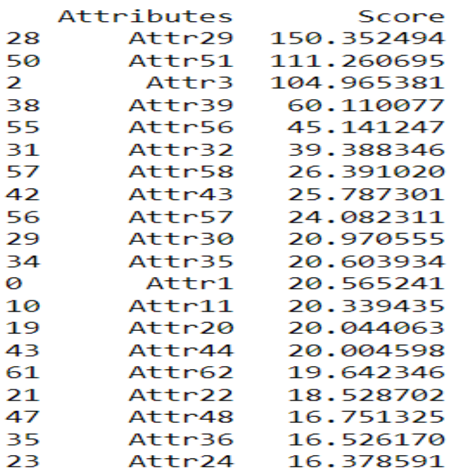
```
   Attributes      Score
34     Attr35  20.992621
21     Attr22  17.123572
24     Attr25  14.315872
9      Attr10  13.435856
1       Attr2  13.295375
37     Attr38  13.137405
2       Attr3  12.363002
50     Attr51  12.258180
23     Attr24  11.856823
5       Attr6  11.504067
28     Attr29  10.168873
13     Attr14   8.550769
6       Attr7   8.548710
0       Attr1   7.711646
10     Attr11   7.636286
47     Attr48   6.751154
17     Attr18   5.848948
32     Attr33   5.317056
35     Attr36   4.299220
54     Attr55   3.863956
```

3rd year selected Features



```
   Attributes      Score
0       Attr1  52.996550
34     Attr35  47.895997
28     Attr29  37.300052
13     Attr14  28.057663
6       Attr7  28.051318
10     Attr11  19.469115
21     Attr22  18.519891
55     Attr56  17.580657
38     Attr39  17.488798
17     Attr18  15.981113
11     Attr12  14.998282
25     Attr26  14.211108
15     Attr16  13.876417
47     Attr48  10.767756
20     Attr21   9.544097
54     Attr55   4.265161
19     Attr20   3.948389
24     Attr25   3.635105
9      Attr10   2.105823
1       Attr2   2.054137
```

4th year selected Features



```
   Attributes       Score
28     Attr29  150.352494
50     Attr51  111.260695
2       Attr3  104.965381
38     Attr39   60.110077
55     Attr56   45.141247
31     Attr32   39.388346
57     Attr58   26.391020
42     Attr43   25.787301
56     Attr57   24.082311
29     Attr30   20.970555
34     Attr35   20.603934
0       Attr1   20.565241
10     Attr11   20.339435
19     Attr20   20.044063
43     Attr44   20.004598
61     Attr62   19.642346
21     Attr22   18.528702
47     Attr48   16.751325
35     Attr36   16.526170
23     Attr24   16.378591
```

5th year selected Features

In the second step, select the relevant and influential features based on their score. The highest score value of the attributes is the most important feature to build the predictive model. The results of feature selection tests with filter feature ranking and selection method and where the rankings of the attributes are shown for each dataset. So, ***SelectKBest*** methods rank each attribute based on its individual score and select the best attributes which have the highest score corresponding to the output of the target class which is defective or defect-free.

Therefore, as shown above in the table, the 20 most important features for each of the considered classification cases for 5 years of datasets. Analyzing the results presented in the table above, it can be said that only one indicator ***Attr29*** (logarithm of total assets) appeared in each research year. And also, six indicators such as ***Attr2***(total liabilities / total assets), ***Attr3*** (working capital / total assets), ***Attr6*** (retained earnings / total assets), ***Attr10*** (equity / total assets), ***Attr25*** ((equity - share capital) / total assets) and ***Attr55*** (working capital) are a most important attribute for distress prediction. Because they occurred in 4 out of 5 years. These selected Bank features have the strongest relationship with the output of the target class, which are distress or not distress. Then, the proposed model is trained using the selected Bank features after SMOTE data processing is conducted.

4.4.3 SMOTE Sampling

In our experiments, we solved the class imbalance problem using SMOTE data sampling on selected Bank distress features. It is done by modifying the training data distributions of the minority class of all research years. SMOTE works by selecting one existing minority class sample m, next it finds its *k*-nearest neighbors and should select at random one of the *k* samples, called *n*. Then calculate the difference between samples m and n and then multiply this with a random number between 0 and 1, and the resulting value is Synthetic data and it is added to the feature vector the train data.

For example: First, let us show the original dataset class distribution of the 1st year dataset, the 0 lines labeled in false non-distress, and the 1 line labeled distress data. This shows how imbalanced our original dataset is. Most of the transactions are non-distress. After using SMOTE sampling, the dataset is balanced as shown in the right of Figure 4.3. which means, the number of the minority class and the majority class training samples are equal. Now, the proposed model has enough data to learn from features in both classes. Then we use this data frame to build our predictive models and analysis we get an accurate result, and it delivers better performance than the previous dataset. Hence, both classes have an equal number of instances to learn the model.

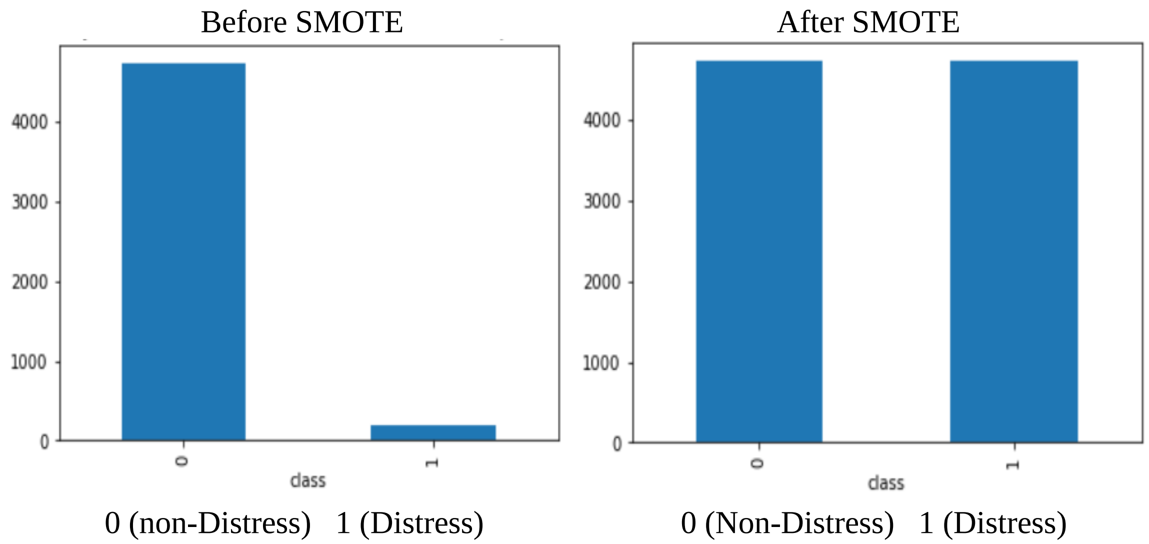


Figure 4 2. Class Distribution before and after sampling

4.5. Result and Discussion

For the experimental purpose, we have used all the datasets as experimental analysis. The first-year dataset means distress after three years, the second-year dataset means distress after four years, the third-year dataset was distressed after three years, the fourth-year dataset was distressed after two years fifth-year dataset was distressed after one year

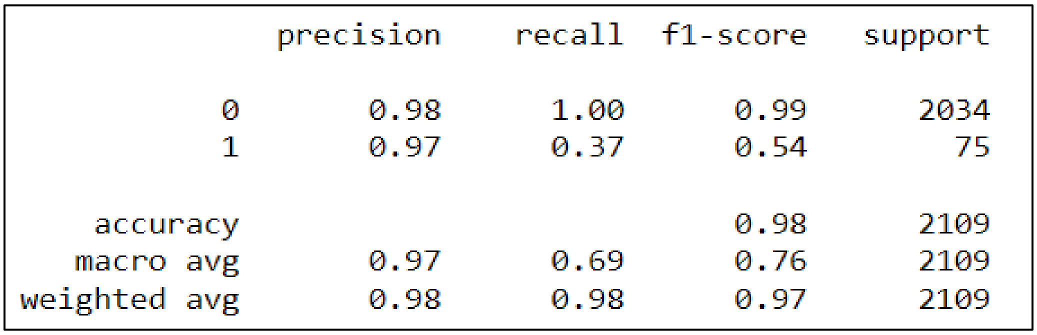
4.5.1. Experimental Result for 1st year Dataset

In this unit, we present an experimental study on the 1st year dataset using the proposed approach for Bank distress prediction. The experiments have two scenarios, and the performance of the experimental results is evaluated using statistical, ROC-score curve metrics exhaustively, and a confusion matrix.

A. Statistical Performance Analysis

In the first scenario: making distress classification on the original dataset without feature selection and SMOTE sampling techniques are applied. The outcome is shown in the table below, the statistical performance for the 1st year dataset before data preprocessing using the Random Forest classifier.

Table 4. 7. Result of 1st Year Dataset before data preprocessing

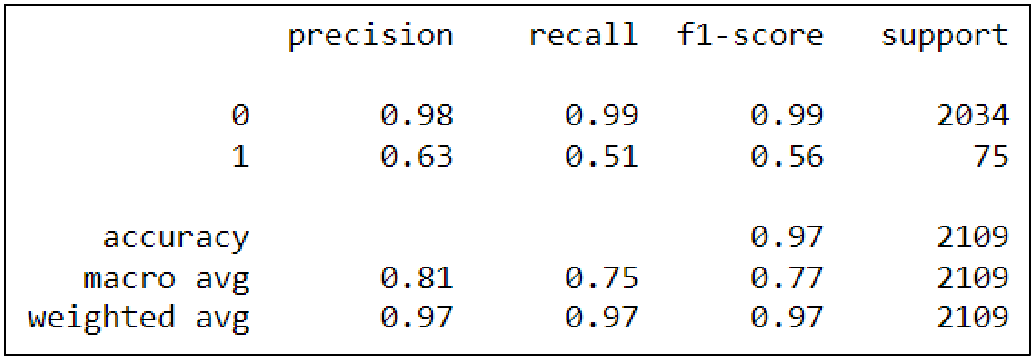


```
              precision    recall  f1-score   support

           0       0.98      1.00      0.99      2034
           1       0.97      0.37      0.54        75

    accuracy                           0.98      2109
   macro avg       0.97      0.69      0.76      2109
weighted avg       0.98      0.98      0.97      2109
```

Table 4.7 shows that the performance of the majority class is higher than the minority class. The precision, Recall, and F1-score value of the majority class is 98%, 100%, and 99% respectively. Whereas, the classification result of the minority is a little low when compared with the majority class since the Precision, Recall, and F1-score value of the minority class is 97%, 37%, and 54%. This demonstrates how the dominant class completely dominates the minority class's categorization accuracy.

In another scenario, we conducted experiments on feature selection and selected the most important features on the 1st year datasets. Second data sampling experiments are conducted on the selected features using SMOTE sampling. After that, we train our model and classify the test data using the proposed class imbalance distress prediction model. Lastly, we evaluate the statistical performance of the proposed model as follows.

Table 4. 8. Result of 1st Year Dataset after data preprocessing



```
              precision    recall  f1-score   support

           0       0.98      0.99      0.99      2034
           1       0.63      0.51      0.56        75

    accuracy                           0.97      2109
   macro avg       0.81      0.75      0.77      2109
weighted avg       0.97      0.97      0.97      2109
```

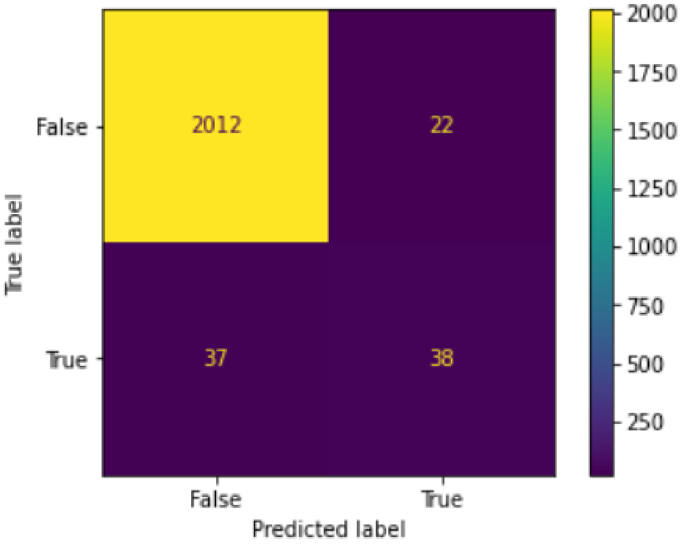
Table 4.8 shows that the minority class classification result of Precision, Recall, and F1-Scores are 63%, 51%, and 56%. At the same time, the classification result is almost similar to the majority class. This shows the proposed model achieved good performance in both the majority and minority classes unlike scenario one results since the Recall and F1-Score value of the minority class is 37% and 54%. This indicates there is a high difference in results when it is compared with the experimental scenario one. Therefore, the proposed class imbalance bank distress prediction model achieved good results by combining feature selection and SMOTE sampling methods since the result of the minority and majority classes have achieved good performance metrics on the 1st year dataset.

B. Confusion matrix performance analysis

A binary classifier's four possible outcomes are counted to create a two-by-two matrix known as a confusion matrix. **Table 4.9** shows the confusion matrix performance analysis for the 1st year dataset using the proposed method. The 1st year dataset has 7,027 instances, from the total 7,027 instances, we used 70% (4,918) instances for training data and 30% (2,109) instances for testing data.

As the confusion matrix result, the proposed model achieved results with 2012 instances is true positive samples. Based on the result of 2012 bank distress classes are classified correctly as non-distress. Similarly, 38 instances are true negative samples, which means 38 instances are classified correctly as distress. Totally 2050 instances are correctly classified as intended from the given dataset from their intended class. And 59 instances are still not correctly classified. This means, all the testing data are not classified properly with their corresponding distress and non-distress class.

Table 4. 9. Confusion matrix for 1st-year Dataset after preprocessing



C. ROC analysis for 1st-year dataset

Figure 4.3 shows the result of the ROC-AUC curve for the 1st year dataset using the proposed distress prediction model. The blue diagonal line indicates a random distress classifier. As shown in the figure, the blue line is a default classifier that represents a completely uninformative test or weak classifier, which corresponds to an AUC of 0.2.

The blue line in the upper curve represents the proposal model, now it is clear that the ROC-AUC of the 1st year dataset is equal to 76%. This shows the prediction accuracy of the proposed model is higher which means that the value of the true positive rate is greater than the value of the false-positive rate.

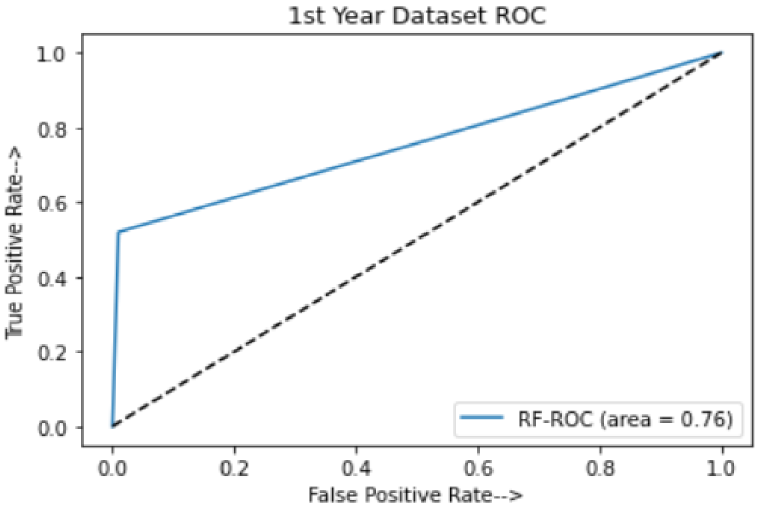


Figure 4.3.ROC analysis for 1st-year Dataset

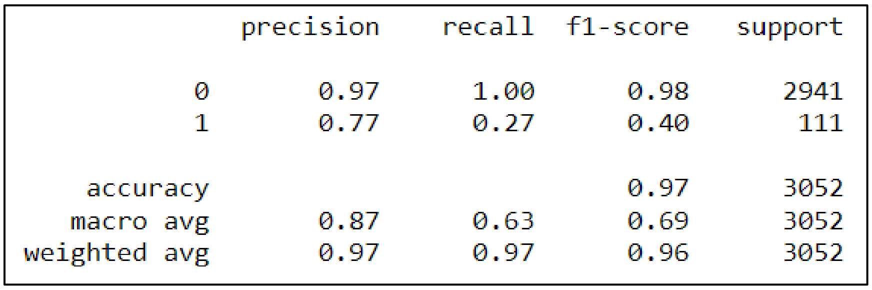
4.5.2. Experimental Result for 2nd year Dataset

In this section, we present an experimental study on the 2nd year dataset using the proposed approach for bank distress prediction. This experiment also has two scenarios first one is before data preprocessing and after preprocessing.

A. Statistical Performance Analysis

First scenario: similarly, we made distress classification on the original dataset without feature selection, and the SMOTE sampling technique is applied. The following **Table 4.10** shows the result of the statistical performance of the 2nd year dataset before data preprocessing using the RF classifier.

Table 4. 10. Result of 2nd Year Dataset before data preprocessing

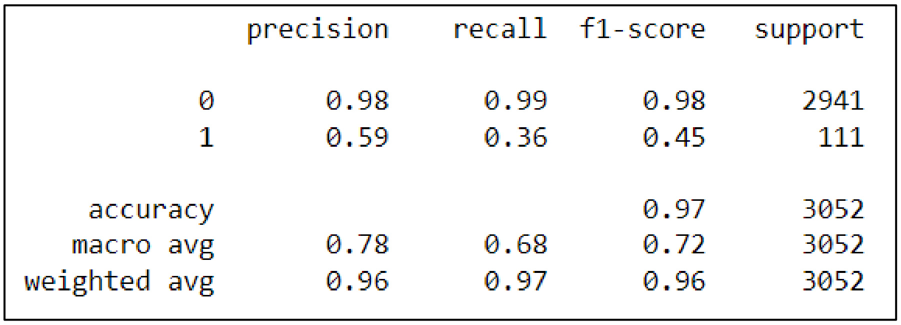


```
              precision    recall  f1-score   support

           0       0.97      1.00      0.98      2941
           1       0.77      0.27      0.40       111

    accuracy                           0.97      3052
   macro avg       0.87      0.63      0.69      3052
weighted avg       0.97      0.97      0.96      3052
```

As shown above, the performance of the majority class is higher than the minority class. The precision, Recall, and F1-score value of the majority class is 97%,100%, and 98% respectively. Whereas the performance of the minority class is very low when compared with the majority class. This shows the classification accuracy of the minority class is dominated by the majority class.

Second scenario: we conducted similar experiments in the previous 1st-year dataset scenario of feature selection using SMOTE sampling techniques. **Table 4.11** shows, it is clear that the result of the classification accuracy of the proposed model on the 2nd year dataset is 97%, which is a promising result and this shows the total correctly classified true positive and true negative instances from the two distress and non-distress classes together.

Table 4. 11. Result of 2nd Year Dataset after data preprocessing



```
              precision    recall  f1-score   support

           0       0.98      0.99      0.98      2941
           1       0.59      0.36      0.45       111

    accuracy                           0.97      3052
   macro avg       0.78      0.68      0.72      3052
weighted avg       0.96      0.97      0.96      3052
```

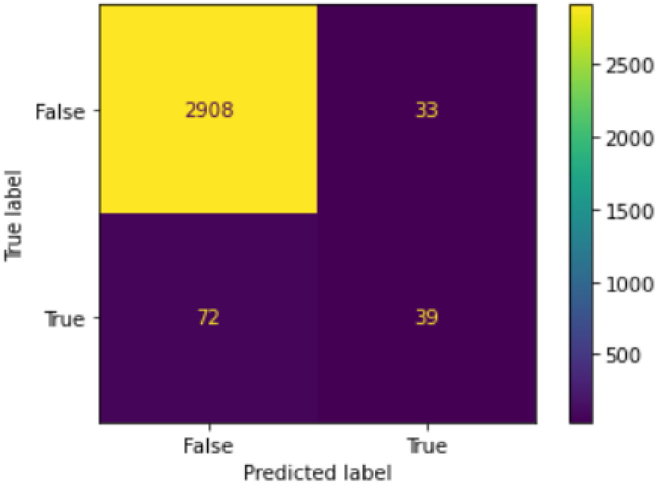
As shown in **Table 4.11**, the minority class classification result of Precision, Recall, and F1-Scores is 59%, 36%, and 45% respectively. Which is dominated by the majority class. This shows the proposed model achieved higher performance in the majority class and achieved lower performance. Therefore, the proposed class imbalance bank distress prediction model achieved promising results by using feature selection and SMOTE sampling methods since the result of the minority classes have achieved better performance in all performance metrics on the 2nd year dataset.

B. Confusion matrix performance analysis

Table 4.12 shows the confusion matrix performance analysis of the proposed distress prediction model on the 2nd year dataset.

As a result of the confusion matrix, the proposed model achieved results with 2908 instances that are true positive. Obviously, 2908 bank distress classes are classified correctly as non-distress. Similarly, 39 instances are true negative, which means 39 instances are classified correctly as distress classes. Totally 2947 instances are correctly classified as intended from the given dataset. From the total given dataset, 105 instances are not correctly classified.

Table 4. 12. Confusion matrix for 2nd year Dataset after preprocessing



True label \ Predicted label	False	True
False	2908	33
True	72	39

On another, 33 instances are false negative which means these data are classified as distress and the number of false positive instances is 72, which means 72 instances are misclassified as false positive. Therefore, 105 instances are misclassified data or false positive instances that have negatively affected the prediction result.

C. ROC analysis for 2nd-year Dataset

Figure 4.4 shows the result of the ROC-AUC curve for the 2nd year dataset using the proposed distress prediction approach, where the precision rate is compared to show the good performance of the proposed model.

The blue line in the bottom curve represents the proposed model, now it is clear that the ROC-AUC of the 2nd year dataset is equal to 69%. This shows the prediction accuracy of the proposed model is high which means that the value of the true positive rate is greater than the value of the false positive rate.

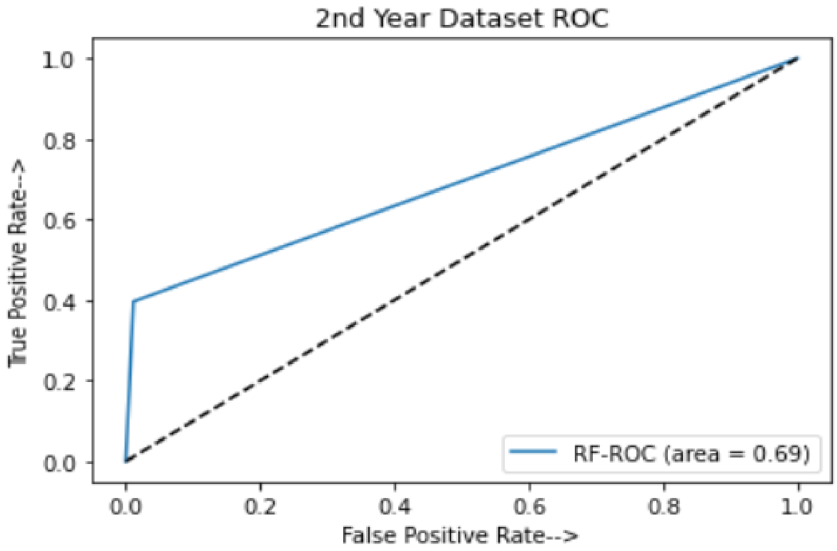


Figure 4.4. ROC analysis for 2nd-year Dataset

In the graph above, the ROC-AUC for the blue curve is 0.69 meaning the proposed model is good at achieving a combination of precision and recall for the 2nd year dataset.

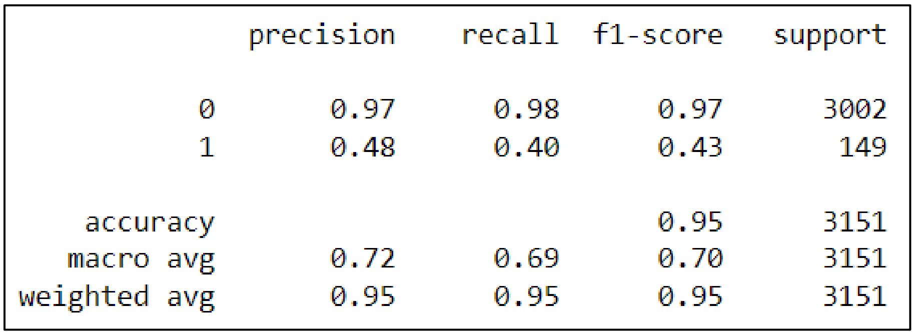
4.5.3. Experimental Result for 3rd year Dataset

In this section, we present an experimental study on the 3rd year dataset using the proposed model for bank distress prediction.

A. Statistical Performance Analysis

The results of the 3rd-year dataset are shown in **Table 4.13**, the accuracy of the majority class is high. It is reflected that before data preprocessing the results of the minority class are low. The precision, recall, and F1-score value of the majority class performed better than the minority class.

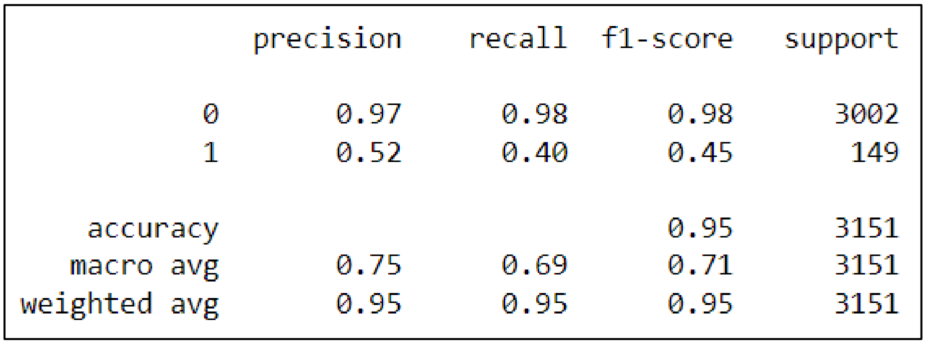
Table 4. 13. Result of 3rd Year Dataset before data preprocessing



```
              precision    recall  f1-score   support

           0       0.97      0.98      0.97      3002
           1       0.48      0.40      0.43       149

    accuracy                           0.95      3151
   macro avg       0.72      0.69      0.70      3151
weighted avg       0.95      0.95      0.95      3151
```

The precision, Recall, and F1-score value of the majority class is 97%, 98%, and 97% respectively. However, in contrast, the precision, recall, and F1-score value of the minority class is 48%, 40%, and 43% respectively. It is a low performance when compared with the majority class. This shows that the classification accuracy of the minority class is dominated by the majority class. **Table 4.14** shows the result of the numerical performance analysis for the 3rd year dataset using the proposed bank distress prediction model.

Table 4. 14. Result of 3rd Year Dataset after data preprocessing



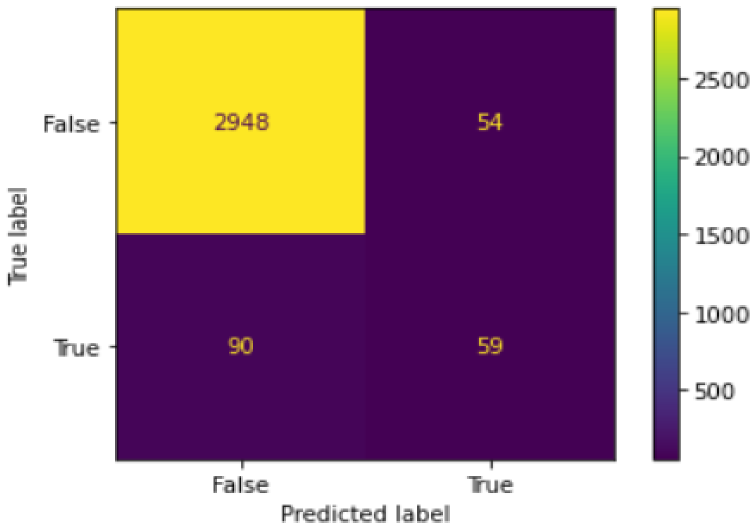
```
              precision    recall  f1-score   support

           0       0.97      0.98      0.98      3002
           1       0.52      0.40      0.45       149

    accuracy                           0.95      3151
   macro avg       0.75      0.69      0.71      3151
weighted avg       0.95      0.95      0.95      3151
```

As shown in the table above, in the minority class the classification result of Precision, Recall, and F1-Scores are 52%,40%, and 45%. This shows the proposed model achieved good performance in both the majority and minority classes. When compared with the experimental scenario one there is a difference in results of minority class.

B. Confusion matrix performance analysis

Table 4.15 shows the confusion matrix performance analysis for the 3rd year dataset using the proposed method. As shown results in the confusion matrix, the proposed model achieved results with 2948 true positive instances. More clearly, 2948 instances are classified correctly as non-distress banks. On the other hand, 59 instances are true negative, which means 59 instances are classified correctly as distress banks. Therefore, the result shows a significant high performance in terms of true positive rate.

Table 4. 15. Confusion matrix for 3rd year Dataset after preprocessing



On the contrary, 54 instances are false negatives which means these data are classified as distress data. Almost 54 instances are misclassified as a false negative, which has a negative effect on the prediction since the cost of false negative is usually much less than false positive.

As we have seen in **Table 4.15**, the number of false positive instances is 90, which means 90 instances are misclassified as false positive, which means there are 90 instances of data distress but classified as non-distress. Therefore, only 144 instances that are misclassified data have negatively affected the prediction result.

C. ROC analysis for 3^{rd} year Dataset

Figure 4.5 shows the result of the ROC-AUC curve for the 3^{rd} year dataset using the proposed distress prediction approach, where the precision rate and recall rate are compared to show the good performance of the proposed model. As we have discussed in the previous section, the black diagonal line indicates a random or default distress classifier which represents an awkward test.

The blue line in the bottom curve represents the proposed model, it is clear that the ROC-AUC value of the 3^{rd}-year dataset is 70%. This indicates the prediction accuracy of the proposed model is predicting the distress goodly.

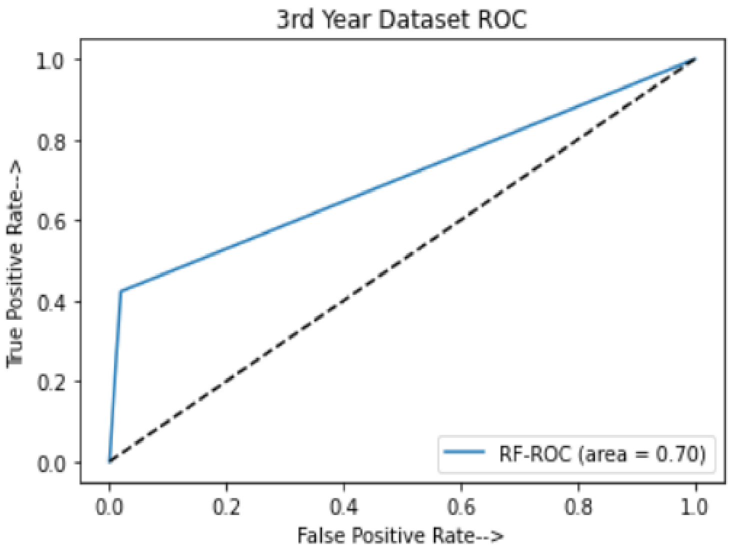


Figure 4.5. ROC analysis for 3rd year Dataset

In the graph above, the ROC-AUC for the blue curve is 0.70, meaning the proposed model is good at achieving a mixture of precision and recall for the 3rd year dataset. With high accuracy corresponding to a low false-positive rate and high recall corresponding to a low false-negative rate, the area under the curve indicates that both recall and precision are good.

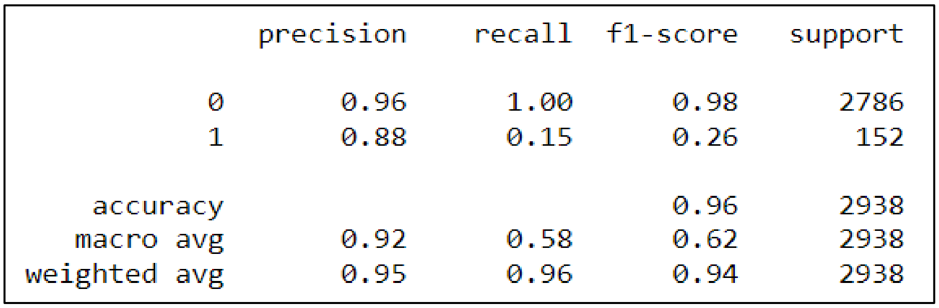
4.5.4. Experimental Result for 4th year Dataset

In this section, we present an experimental study on the 4th year dataset using the proposed model for bank distress prediction.

A. Statistical Performance Analysis

The results of the 4th year dataset are shown in **Table 4.16** and it is reflected that before data preprocessing the results of the minority class in Precision, Recall, and F1-Scores are 88%, 15%, and 26% respectively. Moreover, recall and F1-Score values are very low. Whereas the majority class is performed better than the minority class.

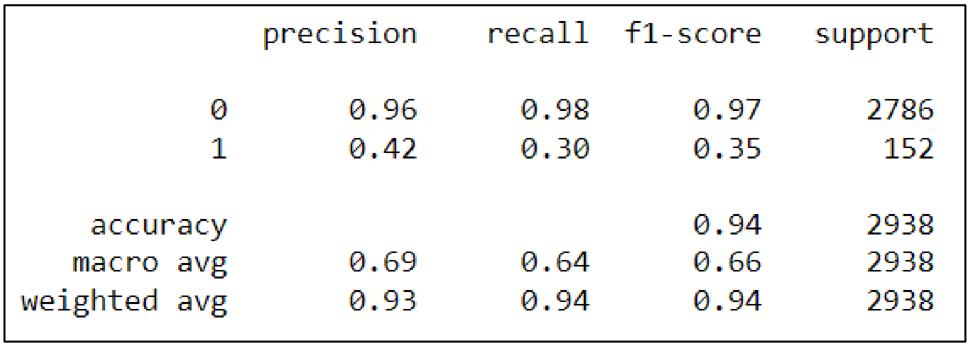
Table 4. 16. Result of 4thYear Dataset before data preprocessing



```
              precision    recall  f1-score   support

           0       0.96      1.00      0.98      2786
           1       0.88      0.15      0.26       152

    accuracy                           0.96      2938
   macro avg       0.92      0.58      0.62      2938
weighted avg       0.95      0.96      0.94      2938
```

As shown in **Table 4.16**, the performance of the majority class is higher than the minority class. Precision, Recall, and F1-Score value of the majority class is 96%, 100%, and 98%. However, in contrast, the Precision, Recall, and F1-Score value of the minority class is 88%, 15%, and 26% respectively. It is a very performance when compared with the majority class. This shows that the classification accuracy of the minority class is dominated by the majority class. The following **Table 4.17** shows the result of the numerical performance analysis for the 4th year dataset using the proposed bank distress prediction model.

Table 4. 17. Result of 4thYear Dataset after data preprocessing



```
              precision    recall  f1-score   support

           0       0.96      0.98      0.97      2786
           1       0.42      0.30      0.35       152

    accuracy                           0.94      2938
   macro avg       0.69      0.64      0.66      2938
weighted avg       0.93      0.94      0.94      2938
```

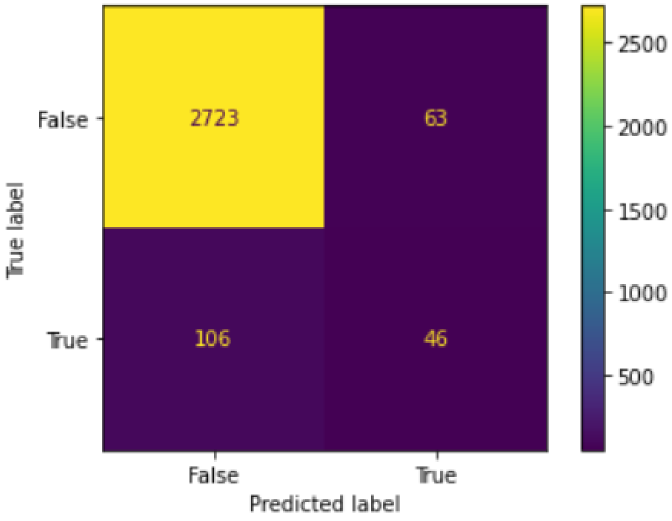
As shown in the table, the classification performance of the proposed model achieved higher accuracy in both the majority and minority classes. Precision, Recall, and F1-Score value of the minority class are 42%, 30%, and 35% respectively. The result shows that there is a difference when it is compared with the experimental scenario one. The proposed distress prediction model achieved good results with feature selection and data sampling methods.

B. Confusion matrix performance analysis

Table 4.18 shows the confusion matric performance analysis for the 4th year dataset using the proposed bank distress prediction model. Similarly, among the given dataset we have used 70% of instances for training data and 30% of instances for testing data.

As shown results in the confusion matrix, the proposed model achieved results with 2723 instances that are true positive. That is 2723 bank distress are classified correctly as non-distress banks. On the other hand, 46 instances are true negative which means 2723 instances are classified correctly as distress banks. This result shows that the proposed approaches achieved high performance in terms of the true positive rate.

Table 4. 18. Confusion matrix for 4th year Dataset after preprocessing



On the opposing, 63 instances are false negatives which means those data are classified as distress data. As shown in the table, the number of false positive instances is 106, which means there are 106 instances classified incorrectly. Therefore, among the total 2938 test data, 169 instances are misclassified data are false positive instances which have negatively affected the prediction result.

C. ROC analysis for 4th-year Dataset

Figure 4.6 shows the result of the ROC-AUC curve for the 4th year dataset using the proposed distress prediction approach, where the precision rate and recall rate are compared to show the good performance of the proposed model. The blue line in the bottom curve represents the proposed model, now it is clear that the ROC-AUC of the 4th year dataset is equal to 65%. This shows the prediction accuracy of the proposed model is not higher which means that the value of the true positive rate is less than the value of the false-positive rate.

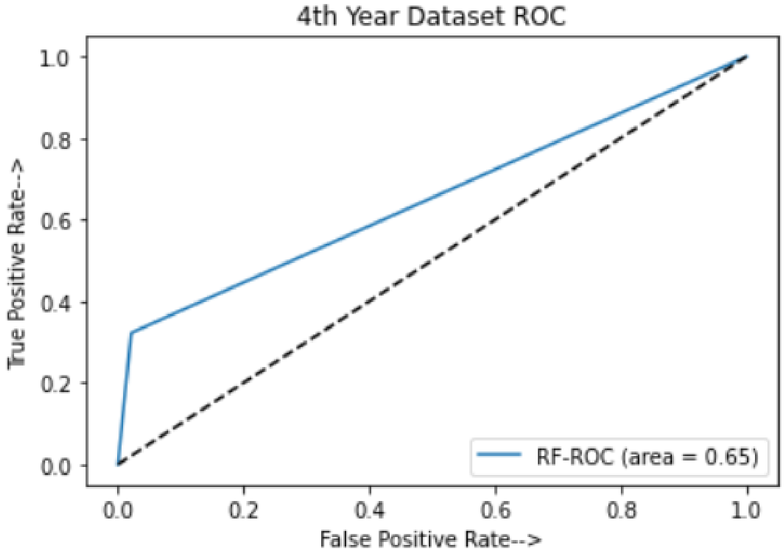


Figure 4.6. ROC analysis for 4th-year Dataset

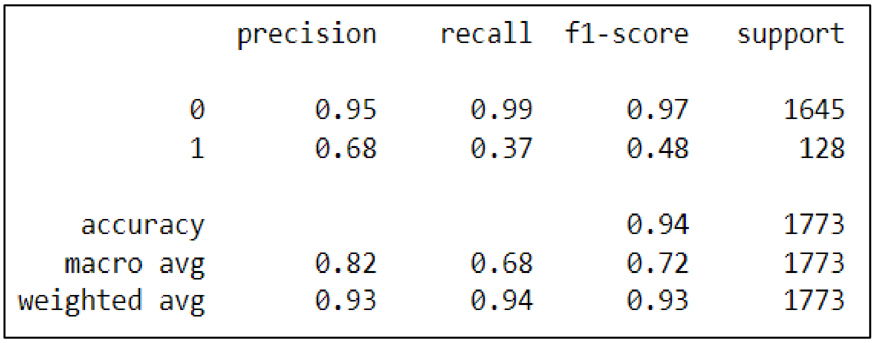
In the graph above, the ROC-AUC for the blue curve is 0.65 which means the proposed model is good at achieving a combination of precision and recall on the 4th year dataset.

4.5.5. Experimental Result for 5th year Dataset

A. Statistical Performance Analysis

In this section, we present an experimental study on the 5th-year data set using the proposed approach for bank distress prediction. The results of the 5th-year dataset are shown in **Table 4.19**, the accuracy of the majority class is high. It is reflected that before data preprocessing the results of the minority class are low. Whereas the majority class performed better than the minority class since the recall, and F1-score value of the minority class is 37% and 48% respectively.

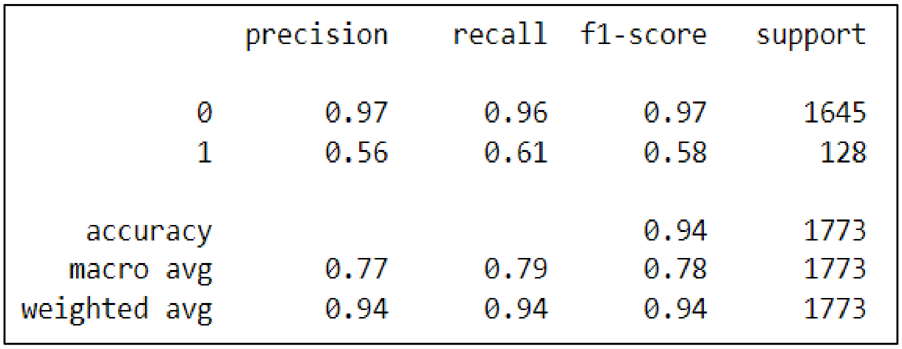
Table 4. 19. Result of 5thYear Dataset before data preprocessing



```
              precision    recall  f1-score   support

           0       0.95      0.99      0.97      1645
           1       0.68      0.37      0.48       128

    accuracy                           0.94      1773
   macro avg       0.82      0.68      0.72      1773
weighted avg       0.93      0.94      0.93      1773
```

Table 4.20 shows the results of the 5th-year dataset using the proposed bank distress prediction model after data preprocessing. As shown in the table, the recall and F1-score value of the minority class is 37% and 48% respectively. But now, it is reflected that the recall and F1-score value of the minority class is 61% and 58%. This indicates there is a difference in results when it is compared with the experimental scenario one.

Table 4. 20. Result of 5thYear Dataset after data preprocessing

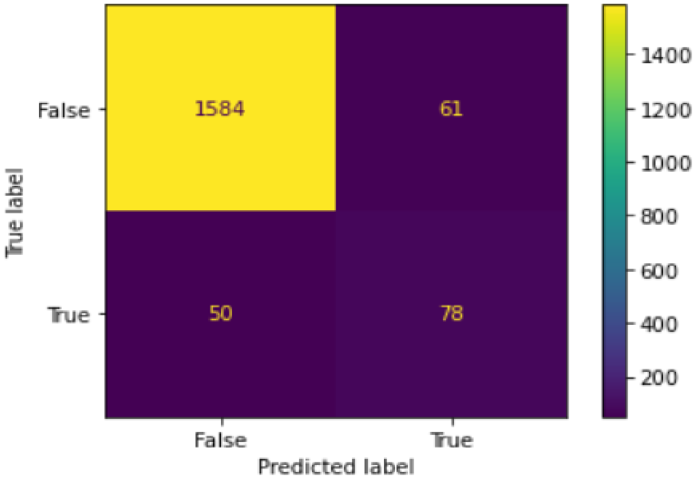


```
              precision    recall  f1-score   support

           0       0.97      0.96      0.97      1645
           1       0.56      0.61      0.58       128

    accuracy                           0.94      1773
   macro avg       0.77      0.79      0.78      1773
weighted avg       0.94      0.94      0.94      1773
```

B. Confusion matrix performance analysis

Table 4.21 shows the confusion matrix performance analysis of the proposed bank distress prediction model on the 5th-year dataset which contains 5,910 data. We used 70% of 4137 instances for training data and 30% of 1,773 instances for testing data. As displayed results in the confusion matrix, the proposed model achieved results with 1584 true positive instances. From the total test dataset, 1584 instances are classified correctly as non-distress banks. Similarly, 78 instances are true negatives. Totally 1662 instances are correctly classified properly with their corresponding class.

Table 4.21 Result of 5thYear Dataset after data preprocessing



On the other hand, 61 instances are false negatives which means those data are classified as distress banks. This means those data are non-distress but classified as distressed. In addition, 50 instances are false positive, which means there are 50 instances are classified as distress but classified as non-distress banks. Therefore, 111 instances have negatively affected the prediction result.

C. ROC analysis for the 5th-year Dataset

Figure 4.7 shows the result of the ROC-AUC curve for the 5th-year dataset using the proposed distress prediction approach, where the precision rate and recall rate are compared to show the precise performance of the proposed model. As shown in the graph, the blue line in the bottom curve represents the proposed model ROC-AUC of the 5th-year dataset is 83%. This shows the prediction accuracy of the proposed model is higher. This means the value of the true positive rate is greater than the value of the false positive rate. A curve pulled upper left corner indicates a better-performing test.

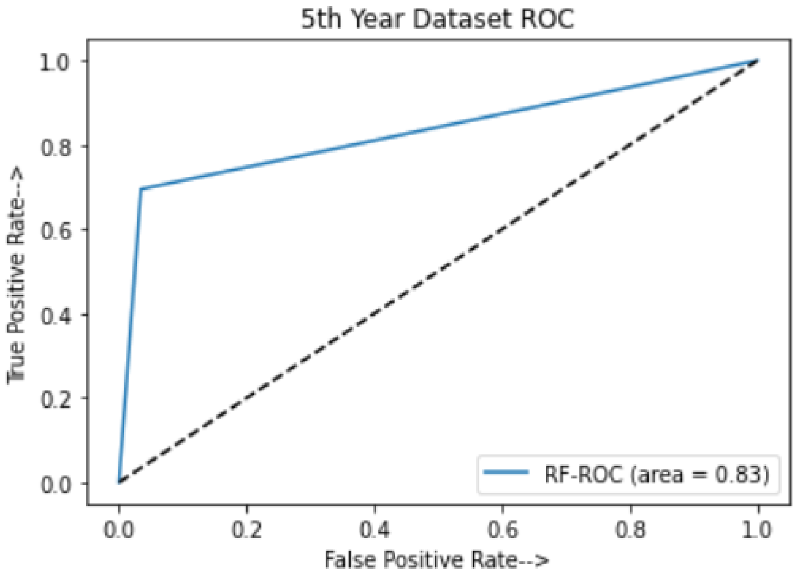


Figure 4.7. ROC analysis for 5th year Dataset

4.6. Machine Learning Classifier Comparison

In this section, the performance of the proposed bank distress predictor is compared with the commonly used machine learning classification algorithms for distress prediction.

Figure 4.8, shows a graphical performance analysis for bank distress prediction. This figure shows a comparative analysis of SVM, DT, KNN, LR, and proposed RF classification. The graph shows the performance of the different machine learning classifiers based on the classification accuracy in terms of ROC-AUC scores attained by all classifiers. As shown in the graph RF and DT achieved better classification ROC scores. From the graph, it is clear that the proposed approach using feature selection and sampling techniques gives better accuracy when compared to other algorithms. Whereas, KNN, SVM, and LR achieved ROC scores of 74%, 62%, and 72% respectively on the 5th-year dataset. The orange line represents the DT classifier achieved a ROC score of 78% which attained a good classification result next to the RF. However, those standard classifiers have less classification accuracy than the proposed distress prediction approaches. From Figure 4.10, the blue line in the upper curve represents the proposed model, the ROC-AUC score of a 5th-year dataset is 83%. Therefore, it is clear the proposed approach using RF classifier is the dominant classifier over the other machine learning classifier.

In general, the comparative analysis shows that the proposed bank distress prediction approaches obtained the highest classification accuracy for all the five datasets category than other machine learning algorithms.

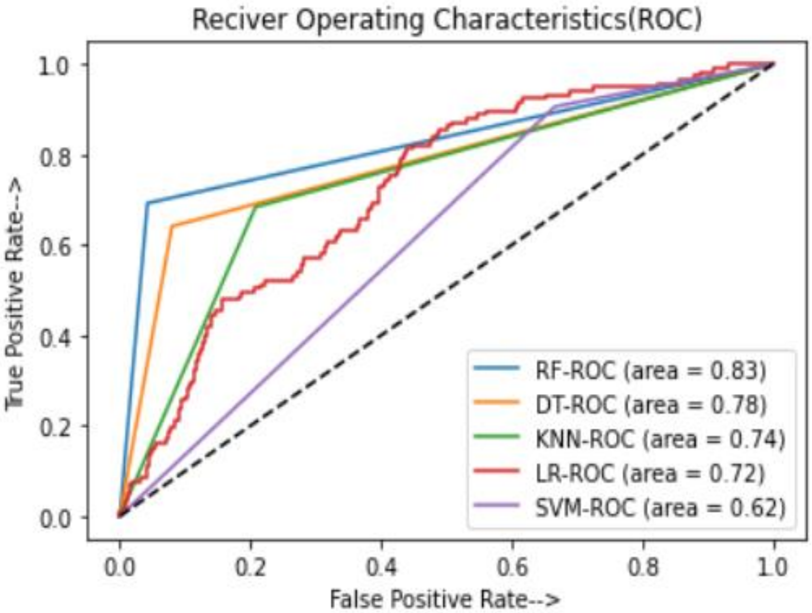


Figure 4.8.Comparison of ML classifier performance

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

Early preparation for declaring distress is of utmost importance to several stakeholders in response to the present expansion of the loan sector and the global economic crisis. The greatest and most modern financial system is crucial for a certain economy. Economic performance is largely reliant on the banking sector. By providing a service for those looking to save and by providing capital to companies looking to develop and grow, banks play a crucial role in the economy. These corporate loans and investments are crucial for enabling economic growth.

In a certain binary classification problem, selecting the relevant and significant features is a great challenge when the data has unequal class distribution and high dimensionality. In this study, we present a bank distress prediction model using feature selection and SMOTE sampling techniques on class imbalanced data. Feature selection is applied to deal with a selection of the most relevant and important features with respect to the predictive actual class and SMOTE sampling techniques are applied to the training data to employ the class imbalance problem.

The prediction result of the proposed bank distress prediction model is very good. The secret behind the upper accuracy is the application that the number of decision trees in the RF classifier to increasing hyperparameter improves the performance of the model but also increases the computational cost of training and predicting.

The conclusions of this study entail that selecting the right set of attributes or variables for bank distress prediction is very important and it is a critical accomplishment. From a practical point of view this investigation, working with a smaller set of features for distress prediction modeling is more effective than working with a large number of bank distress features.

Attr1 (net profit / total assets), Attr2 (total liabilities / total assets), **Attr3 (working capital / total assets),** ***Attr6 (retained earnings / total asset), Attr10 (equity / total assets), Attr14 ((gross profit + interest) / total assets), Attr22 (profit on operating activities / total assets), Attr24 (gross profit / total assets), Attr25 ((equity-share capital) / total assets), Attr29 (logarithm of total assets), Attr32 (current liabilities / cost of products sold), Attr35 (profit on sales / total assets), Attr36***

(total sales / total assets), Attr39 (profit on sales / sales), Attr43 (rotation receivables + inventory turnover in days), Attr48 (profit on operating activities- depreciation / total assets), Attr51 (short-term liabilities / total assets), Attr53 (equity / fixed assets) Attr55 (working capital), and ***Attr62 (short-term liabilities *365 / sales)*** are the most significant features of all datasets.

The proposed method is applied to Polish Bankruptcy datasets. Experimental results show that the proposed method achieved better classification results. Therefore, we can conclude that the proposed approaches improve the classification performance of BDP approaches and it provides a brand-new way of dealing with bank distress.

5.2 Contributions

The main contribution of this study is to predict or forecast bank distress before it may happen. That means the study design prediction model or alarm system has a significant impact on such depositors, managers, investors, banks, decision-makers, and others. This research has two main contributions: scientific and organizational contributions.

As a Scientific, this study has a dynamic scientific contribution to the computer science research field. we proposed a Random Forest classifier approach bank distress prediction (BDP) model that can effectively predict bank distress. The proposed BDP model is quite intelligent to handle binary classification challenges of bank distress prediction. In addition, we have used the Polish Bankruptcy dataset from the UCI Machine Learning repository.

Organizational contribution: using the proposed bank distress prediction model, the banking industry can predict their financial status before it can be distressed or a financial crisis will happen or the Banks can follow up on the financial performance. In addition, this proposed model provides a great advantage for other banks to use these models for their financial health purpose.

5.3 Recommendation and Future Work

In this thesis, the proposed BDP model can be used for the prediction of distress in the banking industry at the early stage before it happens. Since it achieved better performance, we can say that the proposed model is fit to the existing requirements. However, there are two issues, and to provide possible scenes for future works.

It is proposed in the first that several machine learning methods for predicting bank distress be investigated, with the outcomes for achieving the maximum accuracy being compared. Additionally, machine learning algorithms must be researched to study data and locate hidden patterns in the other accountancy-related talks stated, such as share basket analysis, risk analysis, and distress detection.

The second one is it is better if another financial ratio was selected for the best variable or attribute selection for predicting bank distress. And also conducted an experiment on the merged dataset of all years together and tested using different ML classifiers. In addition, for feature work testing the prediction performance using deep learning algorithms.

REFERENCES

A. F. Atiya. Bankruptcy prediction for credit risk using neural networks: A survey and new results. IEEE Trans. Neural. Net., 12(4), 2001.

A. Vieira and N. P. Barradas (2003). A training algorithm for classification of high dimensional data. Neurocomputing, 50C:461–472, 2003.

Abaenewe, Z.C., Ogbulu, O.M., and Ndugbu, M. O. (2019). Electronic banking and bank performance in Nigeria. West African journal of industrial and academic research, 6(1), 171-187.

Aderaw Gashayie & Dr. Manjit Singh. (2016). Development of Financial sector in Ethiopia: Literature Review. Vol.7 No.7, 2016

Alpaydin, E. (2010). Introduction to Machine Learning. 2nd Edition. Cambridge, United States: MIT Press.

Alrasheed, D., Che, D. and Stroudsburg, E. (2017). Improving Bankruptcy Prediction Using Oversampling and Feature Selection Techniques, pp. 440-446.

Arora, N., & Kaur, P.D. (2020). A Bolasso based consistent feature selection enabled random forest classification algorithm: *An application to credit risk assessment. Applied Soft Computing Journal, 86, 1-15*. https://doi.org/10.1016/j.asoc.2019.105936

Ayyadevara, V.K. (2018). Random Forest, Pro Machine Learning Algorithms (Iciccs): 105-116.

Aziz, M., & Dar, H. (2006). Predicting corporate bankruptcy: Where we stand? Corporate Governance, 6(1), 18-33.

Balcaen, S. and Ooghe, H. (2006). 35 Years of Studies on Business Failure: An Overview of the Classic Statistical Methodologies and Their Related Problems. The British Accounting Review, 38, 63-93.

Brealey, R. and Myers, S. (2000). Principles of Corporate Finance. 6th Edition, McGraw – Hill/Irwin, Boston.

Cholmyong Pak, et al. (2017). An Empirical Study on Software Defect Prediction Using over sampling SMOTE. *International Journal of Software Engineering, 811-819.*

Cole, Rebel A., and Lawrence J. White. "Déjà vu all over again: The causes of US commercial bank failures this time around." Journal of Financial Services Research 42.1-2 (2012): 5-29.

Cortes, Corinna and Vapnik, Vladimir, Support Vector Networks, Machine learning, 20:273-297, 1995.

Cunnningham, P. and Delany, S. J. (2014). k-Nearest neighbour classifiers k-Nearest Neighbour Classifiers (2007).

Demirguc-kunt, A., & Detragiache, E. (1998). The determinants of banking crises in developing and developed countries. *Staff Papers*, 45(1), 81-109.

Demyanyk, Yuliya, and Iftekhar Hasan. "Financial crises and bank failures: A review of prediction methods." Omega 38.5 (2010): 315-324.

E. Altman, (1968) "Financial ratios, discriminant analysis and the prediction of corporate bankruptcy," J. Finance, vol. 13, pp. 589–609, 1968.

E. I. Altman (1984). A further empirical investigation of the bankruptcy cost question. *Journal of Finance.*

E. I. Altman. Corporate Financial Distress and Bankruptcy: A Complete Guide to Predicting and Avoiding Distress and Profiting from Bankruptcy. John Wiley & Sons, New York, 2nd edition, 1993.

F. Varetto, (1998) "Genetic Algorithms Application in the analysis of insolvency risk," Journal of Banking & Finance, Vol. 22, 1421-1439

Francisco Navarro. (2011). A dynamic over-sampling procedure based on sensitivity for multi-class problems. Pattern Recognition 443, 1821-1833.

G. Zhang, M. Y. Hu, B. E. Patuwo, and D. C. Indro. Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis. Europ. J. Op. Research., 116:16, 1999.

Garcia, et al. (2012). On the effectiveness of preprocessing methods when dealing with different levels of class imbalance, knowledge-Based Systems 25, 13-21.

Goitom Tariku, (2019). Financial Distress and its determinants in the bank and insurance industry.

Haas, R. D., & Horen, N. V. (2012). International shock transmission after the Lehman Brothers collapse: Evidence from syndicated lending. American Economic Review 102(3), 231-237.

Haixiang, et al. (2017). Learning from class-imbalanced data: Review of methods and application. Expert Systems with Applications, 37-43

Haonan Tonga, et al. (2018). Software defect prediction using stacked denoising autoencoders and two stage ensemble learning. *Information and Software Technology, 94-111*.

https://stackabuse.com/random-forest-algorithm-with-python-and-scikit-learn/

Huenerfauth, M., State,P., Google, G. V. R., & Caltech, R. P. M. (2009). *Introduction to Python*.

J. Ohlson, (1980) “Financial ratios and the probabilistic prediction of bankruptcy,” J. Accounting Res., vol. 18, pp. 109–131, 1980.

John, B., Gianni, D.N., & Elena, L. (2008). Banking Crises and Crises dating: Theory and evidence. *International Monetary Fund Working Paper*, 145: 1-51.

Kumar, P. R. and Ravi, V. (2007). Bankruptcy prediction in banks and firms via statistical and intelligent techniques. A review, 180: 1-28.

Le, H. H., & Viviani, J. L. (2018). Predicting bank failure: An improvement by implementing a machine-learning approach to classical financial ratios. *Research in International Business and Finance, 44, 16–25. doi:10.1016/j.ribaf.2017.07.104*

Le, T., Vo, M. T., Vo, B., Lee, M. Y., & Baik, S. W. (2019). A hybrid approach using oversampling technique and cost-sensitive learning for bankruptcy prediction. *Complexity, 2019.*

Linden, A. (2015). Conducting interrupted time-series analysis for single and multiple group comparisons. *The State Journal*, 15(2), 480-500.

Mckinney, W. (2010). Data Structures for Statistical Computing in Python. Proceeding of the 9th Python in Science Conference, 56-61.

Messai, Ahlem Selma, and Mohamed Imen Gallali. "Financial Leading Indicators of Banking Distress: A Micro Prudential Approach-Evidence from Europe." Asian Social Science 11.21 (2015): 78.

Noreen Kausar, et al. (2016). A Review of Classification Approaches Using Support Vector Machine in Intrusion Detection. International Journal of Advanced Research in Computer Engineering.

Oliphant, T.E. (2006). Guide to NumPy.

Qu, Y., Quan, P., Lei, M., & Shi, I. (2019). Review of bankruptcy prediction using machine learning and deep learning techniques. Procedia Computer Science, 162, 895–899. *https://doi.org/10.1016/j.procs.2019.12.065*

Rahayu, D. S., & Suhartanto, H. (2020). Ensemble Learning in Predicting Financial Distress of Indonesian Public Company. In 2020 8th International Conference on Information and Communication Technology (ICoICT) (pp. 1-5). IEEE.

R. Beaver, “Financial ratios as predictors of failure,” Empirical Research in Accounting: Selected Studies 1966, J. Accounting Research, vol. 4, pp. 71–111, 1966.

Rana & Tarhan. (2018). Early Software Defect Prediction: A systematic map and review. *The Journal of Systems & Software Engineering, 216-239.*

Rossum, G.V. and Python development team. (2020). Python Tutorial Release 3.8.1.1.

Robel Yohannes, (2018). Determinants of Financial Distress: Empirical Evidence from Private Commercial Banks in Ethiopia.

Samuel Ronnqvist and Peter Sarlin (2015). Detect & describe: Deep learning of bank stress in the news. In 2015 IEEE Symposium Series on Computational intelligence (pp.890-897). IEEE.

Shuib Basri, et al. (2019). Performance Analysis of Feature Selection Methods in Software Defect Prediction. *Journal of Applied Science, 276-283.*

Tadesse Yirgu, (2016). Determinants of Financial Distress: Empirical Evidence from Banks in Ethiopia.

UCI machine learning, https://archive.ics.uci.edu/ml/datasets/Polish+companies+bankruptcy+data.

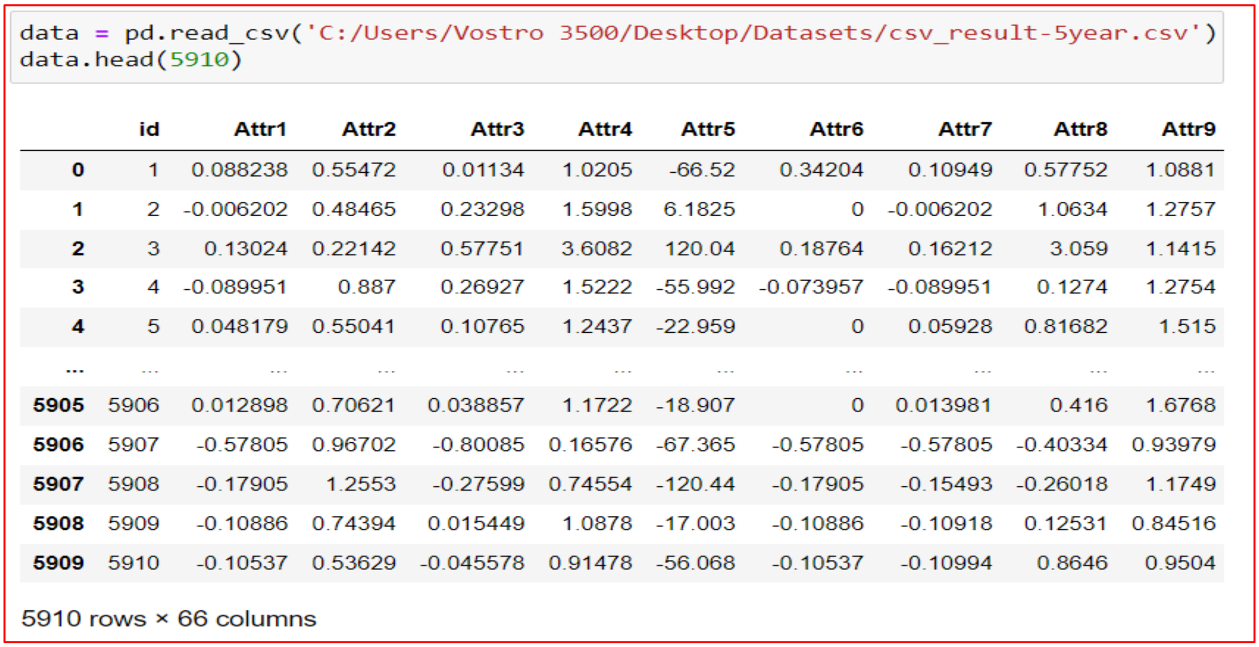
Weston, S., & Bjornson, R. (2016). Introduction to Anaconda. *Yale Center for Research Computing Yale University*.

Wu, Hsu-Che, et al. "Evaluating credit rating prediction by using the KMV model and random forest." Kybernetes 45.10 (2016): 1637-1651.

Xiao-Xiao Niu, Ching Suen. (2012). A novel hybrid CNN–SVM classifier for recognizing handwritten digits. Pattern Recognition 45, 318–1325.

Zieba, M., Tomczak, S. K., & Tomczak, J.M. (2016). Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction. Expert Systems with application, 58, 93-101.

Appendix 1: Dataset

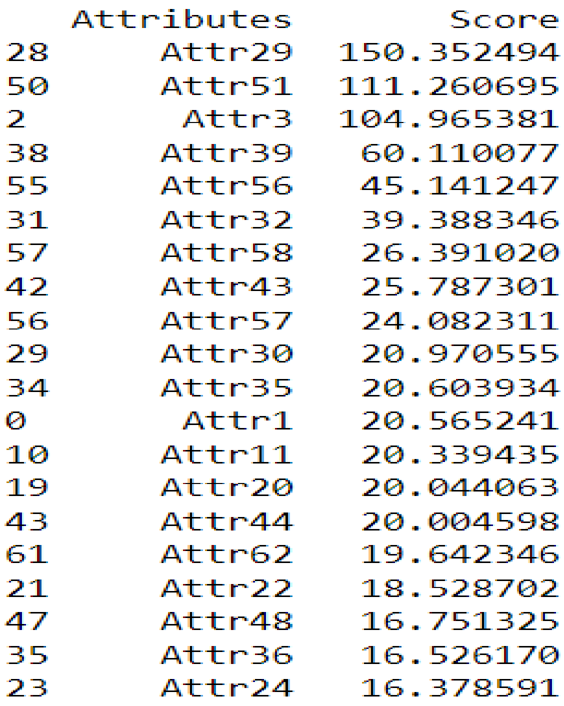


```
data = pd.read_csv('C:/Users/Vostro 3500/Desktop/Datasets/csv_result-5year.csv')
data.head(5910)
```

	id	Attr1	Attr2	Attr3	Attr4	Attr5	Attr6	Attr7	Attr8	Attr9
0	1	0.088238	0.55472	0.01134	1.0205	-66.52	0.34204	0.10949	0.57752	1.0881
1	2	-0.006202	0.48465	0.23298	1.5998	6.1825	0	-0.006202	1.0634	1.2757
2	3	0.13024	0.22142	0.57751	3.6082	120.04	0.18764	0.16212	3.059	1.1415
3	4	-0.089951	0.887	0.26927	1.5222	-55.992	-0.073957	-0.089951	0.1274	1.2754
4	5	0.048179	0.55041	0.10765	1.2437	-22.959	0	0.05928	0.81682	1.515
...	...	...	...	...	...	...	...	...	...	...
5905	5906	0.012898	0.70621	0.038857	1.1722	-18.907	0	0.013981	0.416	1.6768
5906	5907	-0.57805	0.96702	-0.80085	0.16576	-67.365	-0.57805	-0.57805	-0.40334	0.93979
5907	5908	-0.17905	1.2553	-0.27599	0.74554	-120.44	-0.17905	-0.15493	-0.26018	1.1749
5908	5909	-0.10886	0.74394	0.015449	1.0878	-17.003	-0.10886	-0.10918	0.12531	0.84516
5909	5910	-0.10537	0.53629	-0.045578	0.91478	-56.068	-0.10537	-0.10994	0.8646	0.9504

5910 rows × 66 columns

Appendix 2. Feature Selection



```
     Attributes        Score
28       Attr29   150.352494
50       Attr51   111.260695
2         Attr3   104.965381
38       Attr39    60.110077
55       Attr56    45.141247
31       Attr32    39.388346
57       Attr58    26.391020
42       Attr43    25.787301
56       Attr57    24.082311
29       Attr30    20.970555
34       Attr35    20.603934
0         Attr1    20.565241
10       Attr11    20.339435
19       Attr20    20.044063
43       Attr44    20.004598
61       Attr62    19.642346
21       Attr22    18.528702
47       Attr48    16.751325
35       Attr36    16.526170
23       Attr24    16.378591
```