UNIVERSITY OF GHANA

COLLEGE OF BASIC AND APPLIED SCIENCES

ASSESSING THE PERFORMANCE OF NAÏVE BAYES, SUPPORT VECTOR MACHINE, AND GRADIENT BOOSTING MODELS FOR CREDIT SCORING AND BANKRUPTCY PREDICTIONS

BY

EVANS KWABENA MENSAH
(10876491)

THIS DISSERTATION IS SUBMITTED TO THE DEPARTMENT OF STATISTICS AND ACTUARIAL SCIENCE, UNIVERSITY OF GHANA, LEGON IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE AWARD OF A MASTER OF PHILOSOPHY (MPhil) IN ACTUARIAL SCIENCE DEGREE

JULY 2022

DECLARATION

I hereby declare that this thesis is the result of my own research work and that no part of it has been presented for another degree in this university or elsewhere.

…… 11/12/2023 ……

EVANS KWABENA MENSAH **DATE**

10876491

(CANDIDATE)

I hereby declare that the preparation and presentation of this thesis was supervised in accordance with the guidelines on the supervision of thesis laid down by the University of Ghana.

…… ….. 11/12/2023 ……

DR. FELIX O. METTLE **DATE**

(PRINCIPAL SUPERVISOR)

…… 13/12/2023 ……

DR. DENNIS ARKU **DATE**

(CO- SUPERVISOR)

DEDICATION

I dedicate this thesis to my late mother, Madam Akua Sarfoa who died during my studies of this programme.

ACKNOWLEDGEMENT

I have had the privilege of working with and getting to know some wonderful individuals over the past few years. I want to express my heartfelt gratitude to the people listed below.

First, I want to express my appreciation to my supervisors, Dr. Mettle, and Dr. Arku, for their tolerance, inspiration, knowledge, and trust. They did a great job of explaining their thoughts and taking in my rambling, inadequate justifications. Their friendly demeanor and upbeat approach turned an apparently technical scientific field into an attractive and engaging topic. Additionally, I want to thank the University of Ghana's Actuarial Science faculty staff members who assisted and mentored me over the last two years.

My guardians, Mr. Nii Aryee, and Madam Emelia have unquestionably made significant sacrifices to make sure that I obtained a solid education and taught me to be honest and fair to others no matter what. Thanks also to Richard, Hannah, and Comfort, my siblings, for their understanding criticism and encouragement.

I want to express my gratitude to Mr. Richard Mensah and Ms. Gifty Tetteh, my in-laws, for their unwavering belief in me and their unfailing generosity and patience. Rev. Dr. and Mrs. Dzirasah are also appreciated.

Finally, Christabel, my wife, and the love of my life, who is a remarkable lady and the foundation to my studies. Her support and devotion mean the world to me.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AUC	-	Area under Curve
AD	-	Artificial Data
AI	-	Artificial Intelligence
CHAID	-	Chi-squared Automatic Interaction Detection
CART	-	Classification and Regression Trees
CFS	-	Correlation-based Feature Selection
CS	-	Credit Scoring
CV	-	Cross Validation
CAP	-	Cumulative Accuracy Profile
DNA	-	Deoxyribonucleic Acid
ECOA	-	Equal Credit Opportunity Acts
EADs	-	Exposure at defaults
FN	-	False Negative
FP	-	False Positive
GB	-	Gradient Boosting
IRB	-	Internal Rating Based
KS	-	Kolmogorov-Smirnov
LOTUS	-	Logistic Trees with Unbiased Selection
LGDs	-	Loss given Defaults
LDPs	-	Low-default Portfolios
ML	-	Machine learning
MLAs	-	Machine Learning Algorithms
NB	-	Naïve Bayes
PDs	-	Probability of defaults
ROC	-	Receiver Operating Characteristic
SVM	-	Support Vector Machine
SMOTE	-	Synthetic Minority Oversampling Technique
TN	-	True Negative
TP	-	True Positive

UK	-	United Kingdom
USA	-	United States of America
UCI	-	University of California, Irvine

ABSTRACT

Credit risk assessment is critical to reducing defaults, and selecting the appropriate machine learning technique is crucial for this task. Despite the various research on credit scoring, the choice of machine learning techniques remains inconclusive, the problem to identify the most suitable machine learning techniques and models for credit scoring and bankruptcy prediction in the financial sector, considering the challenges of imbalanced data, interpretability, and feature selection. Also, there is currently limited research on the use of machine learning methods for credit scoring and bankruptcy prediction in the banking and finance sector on the dataset used. To address this gap, this study aims to identify the optimal machine learning model for predicting defaults and evaluate the strengths and weaknesses of three models: Naïve Bayes, Support Vector Machine, and Gradient Boosting. The study utilized credit dataset of selected countries downloaded from the UCI machine learning repository, that is, Australia, Germany, and Japan. The dataset was analyzed using the models, Naïve Bayes, Support Vector Machine, and Gradient Boosting embedded in the RStudio analytic program to make predictions of default. Conclusion on the best learning model was drawn on the accuracy of the models to predict default on the three-credit dataset. The analysis show that, the average accuracy prediction of the models is Gradient Boosting (73.4%), Support Vector Machine (57.3%) and Naïve Bayes (69.8%). As a result, the study concluded that Gradient Boosting model is the best machine learning technique to predict default. Based on this, the study recommends using the Gradient Boosting model as it provides the best balance of accuracy and computational efficiency for this dataset. In addition to the accuracy value, the study recommends considering other evaluation metrics such as precision and recall when selecting a model. Depending on the specific goals of the analysis, these metrics may be more or less important.

CHAPTER ONE

INTRODUCTION

1.1 Background

Data has become increasingly important in numerous areas of our lives over the past several years. Researchers, decision makers and other users rely on relevant data to make decision. As the years go by with data increasing in volume, machine learning has gain recognition in advance applied mathematics (Tian, Shi, & Liu, 2012) and has become a key technique for solving problems in finance, image processing, energy production and many more because of its accuracy and efficiency when analyzing a large data (Waad, 2015). For decades, people (researchers) have favored using machine learning over more established techniques (Tsai, Hsu, and Yen, 2014).

Machine learning techniques have been increasingly used in the financial industry, particularly in credit scoring and bankruptcy prediction. The use of machine learning models has been shown to improve the accuracy and efficiency of these tasks (Boughaci and Alkhawaldeh, 2021).

Traditional credit scoring methods use statistical models that are based on a set of pre-defined rules and variables, which can limit their accuracy and effectiveness (Elhoseny et al., 2020). In contrast, machine learning models can analyze large and complex datasets, identify patterns and relationships, and make accurate predictions without relying on pre-defined rules (Kim and Won, 2017).

There are different types of machine learning techniques that have been applied in credit scoring and bankruptcy prediction, including decision trees, logistic regression, neural networks, support vector machines, and random forests (Shinde and Patil, 2022). Each of

these techniques has its own strengths and weaknesses, and the choice of a particular technique depends on the nature of the problem and the available data.

One important aspect in credit scoring and bankruptcy prediction is the evaluation of a client's credit risk unbiased. Machine learning models can help to reduce bias and discrimination by using a wide range of variables that are not typically considered in traditional credit scoring models (Boughaci and Alkhawaldeh, 2021).

Moreover, the use of machine learning techniques in credit scoring and bankruptcy prediction can lead to issues of faster and informed credit decisions, lower the cost of determining credit risk, and help financial institutions to minimize their liabilities and give credit to clients who will pay it back completely (Chen et al., 2022).

The idea behind the design of machine learning models is to identify and differentiate significant patterns in data from those that are not pertinent. Machine learning is applied in a variety of fields, including the medical profession for medical picture analysis and separation of lung nodules, induction motor failure diagnostics, car license plate identification, automation in the agriculture sector, and oil price prediction, among others. (Kennedy, 2013). Generally, the purpose and similarities of machine learning algorithms determines their grouping. According to SuperAnnotate (2021), an AI company, machine learning is categorized into semi-supervised learning, reinforcement learning, supervised learning, and unsupervised learning.

- Semi-supervised machine learning falls between supervised and unsupervised techniques. It is used when labels are only available for a small number of observations, making data labeling a costly and time-consuming process that requires specialized human resources. An example of this is a photo archive that contains both

labeled and unlabeled images. This method aims to find the structure of input variables by working with both labeled and unlabeled data.

- Reinforcement learning is a method where an algorithm, also known as an agent, learns from its interactions with the environment to make decisions that maximize rewards or minimize risks. The agent continually explores its environment until all possibilities have been evaluated. This approach allows machines to determine the optimal behavior in a specific context to achieve the best results. Deep adversarial networks, q-learning, and temporal difference are a few of the well-liked methods in this area. These algorithms are used in areas such as self-driving cars, robotic hands, and computer-controlled board games. Reinforcement learning is a rapidly growing research area and is expected to become widely used in the future.
- Supervised learning involves utilizing previously acquired information, known as labeled data, to generate a desired output. This method involves using a machine learning model to establish a connection between an output variable (y) and input variable (x) through a mapping process.
- Unsupervised learning is a type of machine learning where the algorithm is trained on a dataset without any supervision or predefined labels. Unlike supervised learning, where the algorithm is trained on a labeled dataset and then makes predictions on new, unseen data, unsupervised learning algorithms analyze unstructured or unlabeled data to identify patterns and relationships. In unsupervised learning, the algorithm must learn to identify patterns and structure within the data on its own. This is typically achieved through clustering or association analysis. Unsupervised learning can be beneficial in identifying undiscovered patterns within a dataset and can assist in identifying the characteristics necessary for categorization.

According to Kennedy (2013), machine learning is aimed at building techniques / tools that can correctly and effectively automate operations that are usually performed by people. This is accomplished by identifying patterns in a subset of training data, which can then be used to generalize about the entire data set.

Financial institutions typically evaluate a customer's creditworthiness before granting a loan. One common method of evaluation is credit scoring, which uses numeric expressions to estimate the probability of default on a loan (Mester, 1997). This statistical technique is used to assess the likelihood of a current borrower, loan application, or counterparty becoming overdue or failing to repay their financial commitments, particularly for consumer loans, credit cards, and mortgages (Kenton, 2019). A credit score is typically classified as either "good" or "bad". A high credit score indicates a high probability of fully repaying financial commitments, while a low credit score suggests that the counterparty may have a history of struggling to repay debts and may do so again in the future (Kagan, 2019).

Financial organizations formerly offered financial services to consumers, enterprises, and major corporations through credit reporting. Credit reports provided information about the demographics, insurance, and other utilities of the consumer or business (Aire, 2017). The foundation of modern credit scoring was established by Ronald A. Fisher (Fisher, 1936). In 1941, Durand realized that separating good loans from bad loans could be accomplished using the same method. Linear programming was used to create the first credit rating algorithm (myFICO, 2018). Initially, both the variables chosen, and the scores assigned were mainly based on judgment.

The systematic implementation of a scoring methodology signaled the start of applying statistical techniques to assess creditworthiness in a straightforward and structured manner, assisting to assure uniformity in the credit application process. The Fair Credit Reporting Act, passed in 1970, mandated that financial institutions, known as Credit Reporting Services

Providers, make their files accessible to the public. This ensured that credit decisions were not based on biased information such as race, gender, and disability (Federal Register, 2011).

In the past, credit scoring models primarily relied on data related to past payment history to determine a borrower's ability to repay. These models typically employed methods such as regression and decision trees to produce a credit score using limited structured data. However, in a bid to develop a more thorough imagery of a candidate's credit profile and enhance the precision of their models, financial institutions are now employing new, unstructured, and semi-structured data sources, such open banking transactions, and digital sources. In markets that use traditional credit scoring models, borrowers must have enough historical credit information available to be considered for a score. Without this data, a credit score cannot be created, which prevents potentially creditworthy customers from obtaining loans with advantageous terms.

Largely, the application of machine learning techniques in credit scoring and bankruptcy prediction has great potential to improve the efficiency and accuracy of these tasks, and it can benefit both financial institutions and clients. However, there are also potential risks associated with the use of these models, such as the possibility of errors and biases in the data, and the need for proper regulation and ethical considerations (Zhang and Li, 2022).

1.2 Problem Statement

After the 2007 / 2008 global financial crisis, lending institutions faced challenges in accessing credit to the perceived risks associated with borrows (Barboza, Kimura and Altman, 2017). Credit risk is the possibility of suffering financial loss if the applicant is unable to repay the borrowed funds. This prompted the adoption of machine learning in credit scoring to divide applicants into "good" and "bad" risk categories, banks and other financial organizations are attempting to manage the risk associated with the availability of credit

(Desay, Crook, and Overstreet, 1996). Judgmental approach was used prior to the creation of statistical models, causing lending institutions to lose a lot of capital.

Several studies have explored machine learning algorithms for credit scoring, with varying conclusion. For example, Nti, Aning, Ayawli, Frimpong, Appiah, and Boateng (2021) on twenty-one (21) machine learning algorithms, random forest algorithm consistently performed well in classification and regression tasks across all six datasets. Also, in other studies on credit scoring, logistic regression exhibited superior performance, but no universally ideal algorithm for machine learning emerged from these studies (Bhumireddy and Anala, 2022; Abdou and Pointon, 2011; Boughaci and Alkhawaldeh, 2020).

Credit scoring and bankruptcy prediction play pivotal roles in mitigating lending risks, yet the choice of machine learning techniques remains inconclusive. Imbalances in data, where defaulters are significantly fewer than non-defaulters, pose challenges for accuracy. Various techniques, including oversampling, undersampling, and cost-sensitive learning, have been proposed to address this issue (Hou and Li, 2017). However, it is still unclear which technique is the most effective and suitable for credit scoring and bankruptcy prediction in the financial sector.

Additionally, the selection of features and variables significantly impacts model performance. While traditional credit scoring models rely on credit history and financial ratios, exploring alternative data sources, such as social media, mobile phone usage, and online behavior, is necessary to enhance accuracy (Kim and Won, 2012; Yang et al., 2022).

Overall, the problem statement for this topic is to identify the most suitable machine learning techniques and models for credit scoring and bankruptcy prediction in the financial sector, considering the challenges of imbalanced data, interpretability, and feature selection. This

requires further research to evaluate and compare different machine learning techniques, data pre-processing methods, feature selection strategies, and performance evaluation metrics.

1.3 Objective of the study

To reduce the risk associated with making loans to borrowers who are likely to default, the focused objective of this thesis is to identify the ideal machine learning approach for credit ratings and bankruptcy forecasting in the financial sector. These specific objectives guided the study:

a) To evaluate the accuracy of Naives Bayes (NB), Support Vector Machine (SVM), and Gradient Boosting (GB) machine learning models using Japanese, German, and Australian credit data. and compare their performances in credit scoring and bankruptcy prediction.

b) To examine the precision of the selected machine learning models for credit scoring and bankruptcy prediction and provide insights and recommendations for the development and implementation of effective machine learning models for credit scoring and bankruptcy prediction.

1.4 Significance of the Study

The significance of studies on machine learning techniques for credit scoring and bankruptcy prediction in banking and finance is quite significant. This is because banks and financial institutions play a critical role in the global economy, and managing credit risk is essential for their survival. In recent years, there has been a growing interest in the use of machine learning techniques to address these issues.

Firstly, studies in this area aim to develop accurate and efficient models for credit scoring and bankruptcy prediction. Traditional statistical methods have been used in the past, but machine

learning models offer the potential for improved accuracy, especially in complex and large datasets. These models can be trained on historical data and used to predict the creditworthiness of future borrowers, thereby reducing the risk of default, and increasing the profitability of banks and financial institutions.

Secondly, studies in this area aim to identify the most appropriate machine learning algorithms and techniques for credit scoring and bankruptcy prediction. There is a vast array of machine learning algorithms, and choosing the right one for a specific task is essential for achieving accurate and reliable results. Thus, studies in this area help in selecting the most suitable algorithm or combination of algorithms to improve the accuracy and reliability of credit risk assessment.

Thirdly, these studies have practical implications in the banking and finance industry. Banks and other financial institutions can use these models to assess credit risk and make informed lending decisions. This can lead to better risk management, reduced losses due to default, and increased profits. Additionally, these models can also be used to identify customers who are at risk of default, and targeted interventions can be made to prevent default and improve customer outcomes.

Moreover, the use of machine learning algorithms can help financial institutions evaluate a client's credit risk more objectively and unbiasedly. Traditional credit scoring methods often rely on subjective factors, such as personal relationships or intuition, which can introduce bias and inaccuracies into the credit evaluation process. By using machine learning algorithms, financial institutions can analyze a wide range of objective data points, such as credit history and financial behavior, to evaluate credit risk more fairly and accurately.

1.5 Scope and limitation of the study

Due to the difficulty in obtaining primary credit data, secondary data was downloaded from the University of California, Irvine (UCI) Machine Learning Repository. This may bring down the generality of the model on other credit data. The study covers the use of Naïve Bayes, Support Vector Machine, and Gradient Boosting machine learning algorithms. The study also includes the use of different data sources, such as financial statements, credit reports, and demographic information, to train and test the machine learning models.

Furthermore, the study explores the challenges associated with credit scoring and bankruptcy prediction, issues of data quality issues, class imbalance, and model interpretability. It also examines the ethical and legal considerations associated with using machine learning models in credit risk assessment.

Largely, the scope of this study is to provide the most suitable machine learning model for predicting default and limitations of using machine learning techniques in credit scoring and bankruptcy prediction, and to inform the development of more accurate, efficient, and fair credit risk assessment models for the banking and finance industry.

1.6 Organization of the study

This thesis is grouped into five chapters:

The study's background, problem statement, objective, justification, scope, limitations, and organizational structure are all covered in the first chapter, which also provides a basic overview of the study.

The second chapter offers a survey of comparable works that have been done by other researchers on the subject area, both theoretical and empirical, as well as pertinent literature on credit score and bankruptcy.

The third chapter examines the methods, including Naïve Bayes (NB), Support Vector Machine (SVM), and Gradient Booting (GB). Finally, performance indicators for credit scoring and bankruptcy prediction in the financial industry are presented along with the model's parameters and presumptions.

The fourth chapter examines the results of the technique outlined in chapter 3 and performs analyses for credit ratings and bankruptcies in the banking and financial industry.

The summation, conclusion, recommendation, and suggestions for further study in the area are included in the last chapter.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

The idea of machine learning is introduced in this chapter. The chapter also highlights some selected machine learning models, that is, the idea behind the model and their mode of operation. The chapter also considers several topics including, Bankruptcy, Credit scoring, Data quality and classifiers performance evaluation among others.

2.2 Machine Learning Techniques and Classifiers

In the current day, a wide variety of machine learning methods have been created to assist in coping with challenging real-world scenarios, including financial institutions. Automated and self-correcting, machine learning algorithms are designed to get better over time (Tavasoli, 2022). This section of the study expands on three algorithms (techniques), that is, Naïve Bayes, Support Vector Machine and Gradient Boosting captured in various studies by authors such as Van Gestel and Baesens (2009), and Vidal and Barbon (2019).

2.2.1 Naïve Bayes (NB)

A forecasting model called Naïve Bayes determines the likelihood of an event based on the frequency with which it occurs in a certain population. The Naïve Bayes algorithm was first developed by Thomas Bayes, who was also known as "Father of Probability and Statistics". It is among the most popular prediction models for credit scoring systems in use today.

Researchers have recently integrated Naïve Bayes Classifier techniques into user-friendly toolkits, warranting the creation of decision support systems in a variety of application domains. As a result of technological development, end users — not just researchers — can now create and implement their own Naïve Bayes Classifier solutions. As a result, these tactics are starting to be accepted in standard business practice, and recent studies have

shown that they offer an exceptional return on investment. End-users need systems that respond to their questions almost immediately to handle procedures in daily operations effectively. One of the main sectors with decision issues that require effective techniques of handling the processes is credit evaluation by lending institutions.

2.2.2 Support Vector Machines (SVM)

Vapnik and Cortes (1995) developed the support vector machine as an alternative to two-group classifiers. The machine theoretically embodies the idea that data points are translated non-linearly into a huge subspace. A piece-wise linear field is built in this feature space. Strong generalization skills of the learning machine are guaranteed by the distinctive qualities of the decision surface.

In support vector machine, a soft-margin is employed to permit some misclassification using a slack variable, preventing over-fitting. The "kernel technique" can be used to transfer the data to a higher dimensional feature space (Mercer, 1909), which makes it possible for the data to be linearly separable. Credit risk assessment is one of the areas where SVMs have reportedly performed well (Van Gestel and Baesens, 2009). However, it is well known that SVMs have a high computational cost for large-scale classification problems, are sensitive to parameter choice, and lack transparency, making them difficult to grasp.

A variety of classification techniques, such as word categorization, image identification, and analyzing gene expression, have been successfully completed using support vector machines (SVMs) (Cristianini and Shawe-Taylor, 2000). On the other hand, SVM credit scoring experiments are still in their infancy. Many studies evaluating the efficiency of SVM for credit scoring have just been published. SVMs and other classifiers were used by Baesens et al. (2003) to analyze various data sets. They contend that even if SVMs perform better than other algorithms, they are not necessarily the most effective.

2.2.3 Gradient Boosting (GB)

Gradient Boosting (GB) is a method of amplifying the importance of an attribute in a model. Attributes that have high values are boosted more than attributes with low values. This allows us to create models which can be used for credit scoring and risk assessment purposes. Many forms of data, including text, photos, DNA sequences, and social network analysis, have been subjected to gradient boosting.

Gradient boosting machines (GBMs) are trained by fitting new models to new data, leading to more accurate estimates of the predictor variables. The method entails creating brand-new ground with the highest correlation to the gradient of the ensemble's overall loss function. The researcher finally chooses which loss function to utilize, but when the traditional squared-error loss function is applied, this approach might lead to consecutive error-fitting. The loss function used may vary depending on the specific research goals.

GBMs are highly adaptable and can be customized for specific data-driven activities, which adds flexibility to the model design process. Choosing the best loss function may require some trial and error, but boosting methods are generally user-friendly and allow for testing of various model architectures. Studies including those by Bissacco et al. (2007), Hutchinson et al. (2011), have shown that GBMs are efficient in a variety of machine learning and data mining tasks as well as practical applications.

2.3 Classifier Performance Evaluation

Evaluating the performance of classifiers is crucial, both in terms of absolute performance and in comparison, to other classifiers. Poor classifier performance can lead to significant costs. Japkowicz and Shah (2011) have identified four key factors used in assessing classifier performance: performance measurements, bias prediction, test reference assignment, and predictive validity testing. The performance measurements element deals with the choice of

metrics for comparing projected classifier results to actual results. After choosing an appropriate performance metric, the bias prediction element is utilized to determine the best method for evaluating the performance of the classifier. These methods try to make sure a sample from a population sample is chosen to evaluate classifier performance. As a result, the chosen performance metric is estimated in a neutral manner. The processes for determining the significance of the findings while evaluating the effectiveness of a classifier are covered by the statistical significance testing component. One of the most important areas of machine learning research is the application of statistical techniques to aid in the selection of a certain classifier (Dietterich, 1998). The last element, test benchmark selection, considers the suitability of the datasets and domains used to assess a classifier's performance. When evaluating a learning algorithm's effectiveness across many domains, this is very crucial. The complexity and amount of the datasets from each domain may vary significantly, which may affect how well a given learning method fits a given domain. I do not examine the test benchmark selection component because this thesis exclusively addresses the credit scoring domain.

2.3.1 Performance Measures

In this section, several metrics that are commonly used will be reviewed to assess the effectiveness of a classification model. As part of this study, a credit application is accepted (or rated as good or non-defaulter) if the binary classifier output is 1; otherwise, it is rejected (0) (or rated as bad or defaulter). A threshold can be used to binarize the scores that are produced by many ranking classifiers.

2.3.2 Error Estimation

Testing the classification algorithm comes next after choosing an appropriate performance measure. If the data utilized to evaluate a classification system does not precisely reflect the

underlying distribution, incorrect inferences may be drawn from experimental results. The generation of an impartial approximation of the selected performance metric is one goal of error estimating approaches. According to Japkowicz and Shah (2011), error estimating methods can be broadly divided into three categories: resubstituting, hold-out, and resampling.

By building a classifier and evaluating its performance using the same dataset, the resubstituting error estimate is obtained (Kim, 2009). When the full population or a sizable sample thereof is accessible, this strategy is practical. As more cases are utilized to build and test the classifier, the error rate will converge towards the true error rate.

The dataset is randomly categorized into a training set and a test set using the hold-out method. The training set's labeled examples are used as the learning algorithm's input, and it produces a classifier as its output. The unlabeled examples from the withheld test set are subsequently shown to the classifier. The algorithm assumes tags for every occurrence and generates an estimated error rate when using the hold-out approach. The average estimated error rate, or repetitive hold-out prediction, is the result of several iterations of this method. A section of the training set is frequently put aside to adjust the classifier's parameters. This method guarantees that the test set is independent of the training set, making it an effective tool for classification tasks.

When a big dataset is required, resampling error estimation techniques are utilized to generate precise error estimates. K-fold cross-validation (CV), which divides the data into k subgroups of same size, is one frequently used technique. The learning algorithm is trained using the hold-out technique using the remaining subsets after evaluating one subset in each round. The average error estimation is calculated by averaging the partitions formed.

2.3.3 Statistical Significance Testing

Scientific research relies heavily on statistical significance testing to determine if performance variations between categorization models are linked to real causes or to obvious factors that result from uncontrolled variability. To provide a scenario, Brown, and Mues (2012) performed a set of specific significance tests to assess how well classification algorithms performed on datasets with uneven credit scoring. The use of non-parametric statistical tests to compare the effectiveness of two or more classifiers across several data was also advised by Demsar (2006).

To assess the effectiveness of classification methods across various datasets, the Friedman test (Friedman, 1937) should be utilized. On the other hand, if you are comparing just two classifiers, you should use the Wilcoxon signed-rank test (Wilcoxon, 1945). The Friedman associated ranks test and the Kruskal-Wallis one-way variance analysis by rank order test are sophisticated alternatives to the Friedman test that compare classifier performance across all relevant datasets rather than ranking them separately for each dataset, and they are in accordance with Demsar's (2006) guidelines and recommendations. (Kruskal and Wallis, 1952), which compare classification models effectiveness across all relevant data instead of ranking them separately for each dataset.

2.4 Credit Scoring

Before the advent of automated credit scoring systems, retail banking underwriters would subjectively assess an applicant's creditworthiness based on their prior experiences. Since information about the customer was typically gained through personal interactions between the customer and staff at the lender, it was normal for customers to stick with a single lender (Anderson, 2007). An underwriter, usually a member of the bank's management, would

evaluate applications based on the following factors: character, capital, collateral, capacity, and current economic conditions.

The credit issuing procedure had significant flaws, particularly in terms of its dependability and regularity, which impacted the quality of credit issuing judgments. Hand (2001) identified several primary flaws, including limited handling of applications, resulting in lost revenue, the influence of the bank manager's mood on decisions, inconsistency in decision-making among different managers, a lack of universal formalization of the decision-making process, and the inability of the human-based judgment approach to handle all applications.

The lending industry experienced significant growth during the latter half of the 20th century, as consumers increasingly sought credit products such as credit cards. This led to a need for objective lending decision-making methods, which were developed and expanded upon in response to consumer demand (Thomas et al., 2002; Thomas, 2009). In the US, where programs like the Federal National Mortgage Association (Fannie Mae) were established to encourage home ownership in the wake of the Great Depression of the 1930s, the need for mortgage loans also increased during this time (Romasco, 1983). According to Wright (1983) in Shay (2006), "secure housing was inextricably tied to the maintenance of a loyal people" because of fears of both communism and labor unrest. Indeed, by the 1950s, home ownership had become so symbolic of the American Dream that it stood in for citizenship (Shlay, 2006).

Together with the rise in consumer request for credit products, the development of rules aided to promote the use of credit ranking. The United States Congress underlined the necessity for transparency in the loan approval process in the desire of fairness and equality. Due to the significance of credit scoring, the Equal Credit Opportunity Acts (ECOA) were enacted in the US in 1974 and later amended. The ECOA states that lenders are forbidden from treating loan

applicants differently based on certain traits that are regarded as objectionable, such as ethnicity, color, or religion.

It was suggested that creditors could follow their legal obligations by using statistically derived credit assessment systems. According to Regulation B (Section 202.2) of the US Federal Reserve, which implements ECOA, such credit scoring systems must, among other things, be (i) "based on data that are inferred from an empirical contrast of sample groups or the demography of creditworthy and non-creditworthy candidates who decided to apply for loans within a rational preceding period of time"; and (ii) "Developed and validated using accepted statistical principles and methodology". Even while lenders must give up some preference in their loan selections, credit scoring systems are an obvious fix that adheres to the rules established by the ECOA. These structures accomplish this by clearly communicating to credit applicants the reasons for loan denial. The ability to implement these automated processes was made possible by advancements in computer technology (Hand, 2001). This led to a decrease in the price of credit assessment and higher loan recovery caused by customer defaults when compared to traditional, judgment-based systems (Greenspan, 2002 in Mays, 2004). Retail banks began using statistical approaches to understand various aspects of consumer behavior in the 1980s as the capabilities of computers improved (Hand, 2001). This development resulted in the creation of methods that estimate the risk of default, detect fraud, predict customer response to advertising, gauge customer retention, measure customer attrition, track product usage, and determine customer profitability (Hand, 2001; Thomas, 2009a). According to Thomas (2009), the initial use of credit scoring was for evaluating product default scores.

A credit scorecard is typically used to implement a credit scoring system. The scorecard automatically assigns points to significant consumer qualities and transactional components

to evaluate a customer's proportionate risk of failing on their financial responsibility to other customers.

Systems for estimating credit risk can only be based on historical performance, not future predictions, making them imperfect. Due to the inability of credit rating algorithms to identify debtors who later fail on their loan obligations, a sizable fraction of customer debt stays repaid each year (Finlay, 2011). Unexpected events like fraud, divorce, financial naivete or a lack of financial knowledge are frequently to blame for this, as well as debt incurred due to lost income. The percentage of clients who are delinquent on their loan's repayments for mortgages on homes in the country. The creation of credit assessment methods that can discriminate between profitable and ineffective clients depending on how they will handle their future payback responsibilities is therefore of great importance (Finlay, 2011). Practitioners and academics alike acknowledge, according to Hand and Henley (1997), that even a small improvement in the evaluation of customers' default risk can lead to substantial savings in costs.

2.5 Credit Scorecards

According to Siddiqi (2005), a credit card is a tool that financial institutions use to determine a customer's creditworthiness based on a set of statistically significant characteristics. This tool is designed to be systematic, clear, and easy to understand. The scorecard includes features and their properties, which can be chosen from any of the available data sources and describe specific traits of the borrower or loan.

Each trait is a portion of a range of different integers that a feature can have, or a set of values that are mutually incompatible. Each feature is frequently characterized by a collection of one or even more properties. Each characteristic that affects the calculation of the overall credit score is given a certain amount of points on the scorecard.

The process of developing a credit scorecard is iterative and must meet certain performance requirements, such as stability, discriminative ability, interpretability, complexity, conservatism, and robustness. Regulatory agencies recommend overestimating the chance of default of a loan portfolio to account for economic downturns. A higher credit score generally indicates a lower likelihood of defaulting on a financial commitment.

The conditions are not guaranteed to be satisfied by any concrete quantitative framework. Each financial organization has a different decision-making process due to a variety of reasons, including the resources available and corporate / regional cultures. A combination of statistics, law, information technology, consumer, commercial, and management knowledge is ultimately necessary, even while regulatory measures like the Basel Accords provide recommendations.

2.5.1 Dataset Construction

Practitioners frequently point out that the actions taken during this phase take up the most time while building credit scorecards. The following sections outline the key actions involved in building a dataset that will be used to build a scorecard.

2.5.1.1 Data Quality

To ensure the effectiveness of a scorecard in distinguishing between good and bad applicants, the quality of the dataset employed in the cause of the scorecard building phase is crucial. To achieve high-quality data, accuracy, completeness, consistency, and precision are essential properties that need to be considered (Baesens et al., 2009; Lindsay et al., 2010). Data precision refers to how well the measurements of a feature match its actual value, while user input errors and software faults can affect data accuracy. Data completeness pertains to the number of missing values in the data, and data consistency refers to potential conflicts that may arise when multiple data sources are utilized without proper standardization (Parker et

al., 2006). Removing the problematic entries from the dataset is one technique of dealing with incomplete or missing data, but there are other options as well (Florez-Lopez, 2009).

2.5.1.2 Data Quantity

A sizable amount of client data is required to guarantee the production of a trustworthy and good-quality scorecard. In this procedure, the data sources are identified, and the criteria for data gathering are set. The quantity of data needed is based on the scorecard's aim and the type of information pertinent to it. While assessing applications in the past, authorities in the field advised using 1,500 cases of each class (Anderson, 2007; Lewis, 1992; Siddiqi, 2005). The figures, according to Anderson (2007), are from the 1960s, a time when computer power was constrained, and data collection was more expensive. Nonetheless, these sample sizes continue to be extensively utilized today (Crone and Finlay, 2012), with the assumption that credit score databases are comparable across lenders and geographic regions.

Credit rating data can be obtained from a variety of sources, including internal data such as customer account history, external data like financial statements and application forms, and bureau data from credit bureaus and court records. To gather this information, most lenders use the same inquiries and standardized industry data sources provided by credit bureaus (Crone and Finlay, 2012). These suggested sample sizes are deemed suitable since they are sufficient to reflect the population of interest's properties, and there are enough occurrences to prevent linked variables from overfitting the scoring model.

2.5.1.3 Sampling Period

To create credit scorecards, historical data is used, as discussed earlier. Although past performance is not a guarantee of future performance, historical data is a reliable indicator for credit assessment. For the purpose, of creating training datasets for application scoring, two distinct snapshots of each customer are taken (Martens et al., 2010). The customer's

characteristics are documented in the first snapshot, which is taken at the beginning of the loan. The borrower is classified as either good or bad in the second photo, which is taken later at the standard observation point. The period between the two photographs is described as the "result window". Based on the scorecard's business objectives, the size of the result window is chosen. For instance, a short result window (such as 8 months) may be utilized if the aim is to identify debtors as soon as they have fallen behind on their payments without taking the chance of recovery into account. Lenders must choose an adequate duration of the result window to prevent underestimating the default rate and limiting the possibility of misclassifying a customer. There is a chance that the sample used to build the scorecard and yet-unseen future samples will diverge if the result window is too long. These variations could result from modifications in macroeconomic conditions, business strategies, and individual circumstances (Hoadley, 2001). Important information could be missed if the outcome window is too small; for instance, some default events might not have taken place. The time frame for mortgage outcomes is frequently established by creating a graph of the cumulative default rate on monthly basis. By dividing the total number of defaults by the total number of borrowers, the total default rate is determined.

A level in the monthly cumulative risk of default indicates that the data set is mature. After three to five years, one would anticipate the default rate to plateau (Siddiqi, 2005). However, choosing a result window before this time frame is not commonplace if the credit scorecard's discriminatory and intuitive properties are unaffected.

2.5.1.4 Class Label Definition

The goals of the scoring system and how the financial organization evaluates what constitutes success or failure affect the criteria used to label a loan account as "poor" (McNab and Wynn, 2000). In a circumstance of default, the account's credit score frequently declines. A default, according to the Basel II definition (paragraph 452), occurs when either of the following two

things happens: either the borrower is over 90 days late on a significant credit commitment to the lender or the creditor believes the borrower is highly unlikely to repay its credit obligations and may need to retake collateral, such as the borrower's home in the case of a default.

According to Anderson (2007), banking industry have two options for categorizing clients based on their account status: (i) a present state label definition approach, which assesses clients as either positive or negative at the conclusion of the outcome window; and (ii) a worst status label definition approach, which assesses clients as good or bad based on their account status throughout the outcome window. According to Basel II (BCBS, 2005a), the benchmark for poor accounts should be a customer's 90-day worst status over the course of a year. When handling early-stage loan defaults, financial institutions, however, typically use the status label definition approach (Anderson, 2007).

2.5.1.5 Dataset Completion

The data is often split into two sets during the dataset development phase: the training sample and the testing sample. The testing sample is used to assess performance, whereas the training sample is utilized to construct the scorecard. The datasets can be divided in a variety of ways, but the hold-out strategy, which divides the data into training and testing samples in proportions of 70:30, is one that is frequently employed. To fine-tune the parameters of the scoring model, a validation sample is separated from the training sample in the hold-out technique. Cross-validation and bootstrapping statistical approaches can be used to estimate the model parameters if there is insufficient data without losing significant data (Siddiqi, 2005; Bishop, 2006; Japkowicz and Shah, 2011; Thomas, 2009).

2.5.2 Modeling

In the modeling stage, the scorecard can be created when the training and testing datasets have been obtained. The following sections outline each of the steps that must be taken during the modelling stage. The premise that logistic regression is likely the widely utilized model in the consumer credit rating sector underlies most of the discussion that follows.

2.5.2.1 Feature selection

Eliminating duplicate or ineffective features is a step in the process of choosing a subset of features from a larger set to be used in a scorecard. In-depth discussions of this procedure, also known as feature selection, have been found in the literature (Guyon and Elisseeff, 2003). The three primary kinds of feature selection procedures are (i) filter techniques, (ii) wrapper techniques, and (iii) embedding strategies. Filter approaches assess the value of features entirely based on the inherent characteristics of the data, regardless of the classification method.

To evaluate feature, subsets with a specific classification algorithm, wrapper approaches are employed. These approaches involve exploring the feature space using a particular classification algorithm. Embedded techniques, which incorporate the feature selection method into the classification algorithm, represent the third category of feature selection techniques. The discussion in this thesis is focused on frequently used feature selection strategies in credit scoring.

A sizable number of candidate traits that come from the various sources utilized to capture information about the macroeconomic environment and the customer base are typically used in credit scoring. Although Mays (2004) suggests 8 to 15, a strong scorecard normally employs between 10 and 20 features (Thomas, 2009). When building scorecards, feature selection should be done for a variety of good reasons. First, practically speaking, it is crucial

to eliminate as much superfluous and repetitive information as possible to cut costs. Otherwise, employees are compensated for analyzing and comprehending extra features that are unnecessary when determining a customer's creditworthiness. The second benefit of selecting predictive features is that it helps to clarify and improve understanding of the scorecard. Lastly, according to the Occam's razor principle, it is preferable to use a simple scorecard with the best prediction accuracy rather than a more complicated one with extraneous characteristics. Overfitting can occur when the target population is described by an excessive number of duplicated and irrelevant features, which is referred to as the "curse of dimensionality" (Loughrey and Cunningham, 2005). In such cases, the induced model may correctly categorize all the occurrences in the training sample, including the noisy ones, but perform poorly when applied to an unseen sample.

According to Mays (2004) and Siddiqi (2005), feature selection is influenced by four different variables: cost, legality, business logic, and statistical analysis. The computational and monetary costs associated with getting the input are included in the cost components. The usage of elements that raise legal, regulatory, or moral concerns is related to the legal aspects. Builders of credit scorecards must make sure that the features they utilize follow any of these issues.

The inclusion and exclusion of features can be defended by scorecard builders using business reasoning. Practitioners choose specific features for inclusion based on their projected predictive power using expert knowledge gained from previous scorecard building efforts. For instance, a certain trait may highlight quirks of a subpopulation. It is also possible to assess the dependability of features using business logic. For instance, certain commission-based salespeople may use unverified information to boost a credit applicant's chances. Feature values that are deemed anomalous for an instance belonging to a specific

sub-population might be flagged by business logic. Since it's crucial that the characteristics employed in the first dataset be accessible to subsequent samples, business logic can also be used to assess the stability and availability of features in the future. Finally, to avoid the misuse of ratios, it is important to employ business logic to justify their use. Combining existing features to form ratios may result in a higher occurrence of connected features, as noted by Anderson (2007). However, high feature intercorrelations can create multicollinearity problems, which may lead to poor scorecard performance when analyzing new data, as pointed out by Diamantopoulos and Siguaw (2006). Therefore, careful consideration of business logic and feature intercorrelations is necessary when using ratios in data analysis.

In selecting variables, statistical analysis is the ultimate determining factor. To identify the true contribution of each characteristic to the class label, it is necessary to eliminate strongly correlated features, a process that involves utilizing various statistical analysis techniques. Leung Kan Hing (2008) outlines three commonly employed methods for this purpose: correlation-based feature selection (CFS), stepwise procedures, and factor analysis

These methods are frequently employed in data analysis, to quantify the connection among each feature and the class label as well as between each attribute, CFS provides bivariate and pairwise correlation approaches. To rule out potential characteristics, a pairwise correlation threshold is employed. This approach, however, only looks at one set of elements at a time and does not do statistical relevance testing. To discover the lowest collection of characteristics that accurately describes the class label, stepwise techniques, on the other hand, take a subset of features and iteratively perform linear or logistic regression on the class label. Forward selection, backward elimination, and forward-backward selection are the three commonly used methods in stepwise procedures. Factor analysis, the third statistical analysis method, is used to reduce a big collection of linked features into a more manageable

collection of latent underlying components. Component analysis aims to achieve parsimony by employing the fewest number of non-correlated features to account for the most common variance in a correlation matrix. Two popular techniques for factor analysis are maximum likelihood and principal axis factoring.

By merging several traits into one factor, factor analysis has the obvious benefit of reducing the number of features. Park et al. (2005) suggests that the resulting components from factor analysis are likely to have uncorrelated characteristics, which could explain a significant amount of the variance in the original data. However, it is important to note that the degree of correlation among the resulting factors may depend on the specific method(s) used for factor rotation and score extraction. Additionally, Ozkaya and Siyabi (2008) and Yanovskiy et al. (2007) note that interpreting the resulting elements can be challenging, and there may be differences in interpretation among analysts.

2.5.2.2 Coarse Classification

To prepare data for scorecard modeling, it is necessary to reduce the quantity of labels to a sizeable point and transform the data into a suitable form. Data transformations can be used to simplify the structure of the data for modeling purposes. Credit scoring commonly uses a method called coarse classification, where values of continuous features are divided into a limited number of groups. Similarly, category and ordinal feature traits are often combined into fewer categories. This approach allows the logistic regression model to treat each category of each variable as a dummy feature with its own weight. On the other hand, a single regression coefficient is calculated for a continuous feature, which may not sufficiently describe the label's nonlinear relation with the class label. (Carroll and Ruppert, 1988; Hand et al., 2005).

Coarse classification is a useful technique for improving the robustness of a scorecard. By dividing data into distinct groups, coarse classification reduces the risk of over-fitting and enables the creation of categories with sufficient positive and negative observations (Baesens et al., 2009). Coarse classification can also account for non-monotonic connections between specific borrower characteristics and the probability of default. For example, anecdotal evidence suggests that borrowers older than 30 who still live with their parents in the USA have a higher chance of default because leaving home at that age is considered the norm (Siddiqi, 2005). However, borrowers who move out of their parents' home too young (example at age 19 or 20) may have a higher risk of default due to no savings resulting from paying rent and other household expenses. In a logistic regression model, coarse classification maybe be adopted to address such non-linear correlations by dividing the dataset into distinct categories. Additionally, coarse classification can account for missing information and partially address the instability caused by outliers and extreme values by grouping such data into a separate category. The terms binning, grouping, and discretization are often used interchangeably with coarse classification in the literature.

When performing coarse categorization, it is common to divide each label into about six groups, as suggested by Hand et al. (2005). Increasing the number of categories can lead to over-parameterization and cumbersome models, while fewer categories can result in rigid models. Categorical features are typically combined into coarse classes based on similar good-to-bad ratios (Thomas, 2009a). Ordinal features are usually banded together based on nearby qualities, while continuous features are initially categorized into 10-20 groups based on the span between minimum and maximum values, as proposed by Lin et al. (2011). These groups are then combined into fewer coarse groups with comparable good-to-bad ratios, like ordinal characteristics. Irrespective of the data type, it is recommended that groups be huge to

include at least 5% of the sample group to avoid incorrect estimates of attributes on the scorecard (Thomas, 2009a).

2.5.2.3 Reject Inference

When developing a scorecard for a credit application, only customers who were approved for credit are considered eligible for the result value. The financial industry's client database may not accurately reflect the population of all applicants who applied for credit due to sampling bias, also known as reject bias. This means that characteristic values are only available for approved customers, and not for those who were denied credit due to default risk. To counteract this bias, reject inference techniques are used to predict how the rejected applicants would have fared if they were approved. This helps to improve the scorecard's discrimination and give a precise assessment of its performance on the actual population of credit applicants to which it will be used. (Chandler and Coffman, 1977; Hand & Henley, 1993; Thomas et al., 2002, 2009).

One effective approach to address rejection bias is to gather data on customer characteristics and outcomes for all applicants, giving each applicant due credit for a set duration. This strategy has been previously employed by mail order companies and retailers as reported by Thomas et al. (2002). Nevertheless, implementing this method may prove financially unfeasible for several banks, considering the anticipated losses.

To counteract this tendency, numerous reject inference strategies have been created. Simply labeling each applicant who is rejected as evil is a crude approach. Being able to designate a certain segment of customers as evil without having the opportunity to prove this assumption false is obviously a disadvantage of this strategy (Thomas et al., 2002). Extrapolation and augmentation are two of the reject inference techniques that are most frequently utilized. Each strategy has multiple variations. One method for estimating default probabilities is the

straightforward extrapolation method, as described by Meester (2000). Using only the approved applications, a preliminary model is estimated using this procedure, and the likelihood of default for the rejected applications is extrapolated using the model. The estimation of a new model is then performed utilizing both the approved and denied applications. The augmentation strategy, also known as re-weighting, is an alternate approach and is detailed by Mays (2004). With this approach, a model is estimated using the accepted candidates, but each candidate is given a weight depending on the inverse of their chance of being chosen. The inverse probability is then calculated using a second model that is approximated using both the approved and denied applications. According to Banasik and Crook (2009), to foster the presence of denied applications, this strategy unfairly gives the more marginal clients more weight.

The credit scoring community has not reached a consensus on the benefits or drawbacks of using reject inference approaches, nor is there a clear agreement on which approach is best suited for reject inference. One of the reasons for this lack of agreement is the dearth of empirical studies on datasets that contain data on applicants who have been granted and refused, which makes it challenging to assess the importance of reject bias. Reject inference can only be used to boost scorecard performance, according to Crook and Banasik (2004), when a significant portion of applications are turned down.

2.5.2.4 Segmentation

The decision to segment the population and create unique scorecards for each category is typically made early in the scorecard modeling process. Segmentation involves categorizing the population based on factors such as buying habits, income, and age. This approach is commonly used in marketing (Wedel and Kamakura, 2000), as well as credit scoring, to increase the accuracy of the scorecard and provide lenders with more options in terms of

interest rates, repayment structures, and other product settings. However, it is important to minimize the number of segments, as creating additional scorecards requires additional resources to build and maintain. The decision to segment the data is influenced by operational, statistical, and strategic factors (Thomas, 2009a). Operational factors, such as differences in data availability and acquisition, can result in variances between segments (Anderson, 2007). Bank mergers can also lead to large variances in customer information and may necessitate separate scorecards for each customer group.

Anderson (2007) asserts that statistical considerations apply to highly predictive indicators that show a close link to one another. The two features interact when the value of one characteristic affects the predictability of another feature. For instance, the risk related to a person's marital status may change depending on how many children they have, as a single parent is frequently thought to be riskier than a couple with children. Scorecard builders frequently establish distinct scorecards for every characteristic of an accurate estimate feature to avoid the inclusion of too many interacting features.

The bank may apply specific policies related to strategic concerns. For example, clients who are financially stable may be offered loans at lower interest rates. The bank can better manage its clients by segmenting the scorecard population according to the strategies that are most effective for each group.

To segment the population, scorecard builders use statistical and experience-based methods (Siddiqi, 2005). Experience-based methods utilize industry standards and business expertise to identify homogeneous subpopulations based on specific characteristics. Statistical methods, on the other hand, employ statistical tools and learning methods to find relevant population group. For instance, K-means clustering, and self-organizing maps can be adopted

for cluster analysis to create various customer groups based on specific attributes. After segmentation, logistic regression is typically used to create a scorecard for each group.

Due to the lack of use of the customers' class label throughout the segmentation process, the discovered groups may not have distinct risk profiles, which is a drawback of this approach. Moreover, initial segmentation using random vectors may provide local optimal rather than global optimal outcomes (Sherlock et al., 2000). Tree-structured classification, such as CART, which separates segments based on the provided client class label, offers a solution to this issue. Using three real-world datasets, CHAID trees and LOTUS, Bijak & Thomas (2012) examined the applicability of CART with two other segmentation techniques. The suite of segmented scorecards did not, however, significantly outperform the specific methodology, according to the authors.

The data must be "sufficiently diverse" and significant in size to justify the additional expenses incurred in the creation, implementation, maintenance, and oversight of a set of scorecards (Banasik et al., 1996). Additional difficulties develop when each segment has too few errors to allow for accurate scoring validation (Mays, 2004). If these requirements are satisfied, the increased productivity should offset the additional expenses related to using numerous scorecards. Although model performance is a critical factor, segmentation can also be affected by operational and strategic factors that have already been discussed.

2.5.2.5 Model Training

The training and testing datasets must be produced and processed before starting the training of the predictive model. In the consumer credit scoring industry, logistic regression is a widely used algorithm at this stage of scorecard generation (Hand and Zhou, 2009). The training dataset is used to create the prediction model, while the testing dataset is used to assess the model's accuracy. It is crucial to make sure the model appropriately distinguishes

between good and poor risks and avoids overfitting problems by properly fitting the data. The datasets might need to be modified by going back through the dataset creation procedure depending on how well the model performs. The Kolmogorov-Smirnov (KS) statistic and Gini coefficient are commonly used to evaluate performance (Hand, 2012).

2.5.2.6 Scaling

A predictive classifier's output is scaled using a scorecard to get a score that corresponds to a specific good-to-bad ratio. The predictive power of the scorecard is unaffected by scaling (Siddiqi, 2005). Scaling is instead mostly used as a cosmetic exercise to make scorecards easier for non-expert users to read and analyze. Thomas et al. (2001a) conducted a survey and identified several desirable properties for scorecards. These include: (i) ensuring that the total score is positive; (ii) assigning positive points to each scorecard attribute; (iii) providing reference scores with specific good-to-bad odds; and (iv) ensuring that score differences have a consistent meaning across the entire scale.

2.5.2.7 Validation

The credit scorecard needs to be checked for accuracy and consistency before use. This procedure, known as validation, is typically carried out by a faction separate from the building activity employing data that was not included during the modeling step. Validation is a continuous process that is carried out not just after the scorecard has been produced but also on an ongoing basis, particularly whenever there have been any noticeable structural changes to the scorecard. A proper system to substantiate the estimates of Probability of default (PDs), Loss given default (LGDs), and Exposure at default (EADs) is a requirement of Basel II. Discriminatory power and calibration are two major factors involved in PD validation (Stein, 2002).

The ability to distinguish between the distributions of scores for good and bad credit is known as discriminating power (Crook et al., 2007). To measure this power, confusion matrices and associated performance metrics are often employed. In addition, various graphical methods, including Receiver Operating Characteristic (ROC) curves, Kolmogorov-Smirnov statistic, and Cumulative Accuracy Profile (CAP) plots, are commonly used to assess credit scoring models (Stein, 2002).

Calibrating the accuracy of probability of default (PD) is crucial in assessing credit portfolios. PD bands or rating grades are utilized to categorize loans or credits in a portfolio. According to Basel II guidelines (BCBS, 2005), lenders are required to calibrate the PD for each rating grade to ensure that actual default rates fall within the predicted range. To assess whether the estimated PDs match the observed default rates, PD calibration (Crook et al., 2007) is used. This process is commonly known as back testing.

2.6 Credit Score Models

Credit Score Models have been defined in several ways. Vidal and Barbon (2019) assert that the Credit Score Model is a risk management instrument that assesses a loan applicant's credit worthiness by forecasting their possibility of default depending on previous data and employing numerical methodologies. Similarly, the Federal Deposits Insurance Corporation (2007) defines the Credit Score Model as a method used to assist management in making decisions and to offer forecasts of the likelihood of delinquency or default that may be used in risk assessment and loan approval.

Typically, a credit score is used in Credit Score Models to represent the likelihood of default. Higher scores indicate a lesser likelihood of default, whereas lower scores indicate a higher likelihood of default.

The two main categories of credit scoring are generally dependent on the function and the data used (Bijak and Thomas, 2012). The first, known as application scoring, determines an applicant's chance of defaulting within a certain time frame now a credit application is made. The monetary and population details of a portion of prior applicants, coupled with their excellent or poor status later, are often the data utilized for model fitting for this activity.

The next form of credit scoring, known as behavioral scoring, is applied after credit has been issued and determines how likely it is that an existing client will default within a specific time frame. Lenders can keep a close eye on their clients thanks to behavioral scoring, which also facilitates decision-making at the consumer level. Depending on how well customer return their loans as well as their good or bad status later, the data used for model fitting for this task. A bank must correctly forecast the probability of client default over various time horizons to remain profitable. Customers who pose a significant default risk can then be identified, enabling the financial institutions to take the necessary precautions to restrict or protect itself from losses.

2.7 Credit Scoring Challenges

This section's purpose is to provide a thorough explanation of a selection of unique difficulties and issues that scorecard developers face. This section gives an overview of the literature, with a focus on the issues that this thesis addresses: low-default portfolios (LDPs), behavioral scoring, and artificial data.

2.7.1 The Low-Default Portfolio

At certain points of an economic cycle, when the number of defaulters is low, modeling can become more difficult. It is significant to note that when working with imbalanced data, traditional supervised classification approaches may not perform as expected. A situation where one class is disproportionately underrepresented in comparison to the other class is

referred to as imbalanced data. Due to the dearth of defaulters, the credit assessment domain frequently experiences data imbalance, which causes what is known as the low-default portfolio problem. In the framework of class inequality, such portfolios are regarded as being extremely rare.

To utilize the Basel II internal ratings-based (IRB) approach for regulatory capital, lenders are required to develop and validate models with consistent and accurate forecasting capabilities (BCBS, 2005). However, concerns have been raised in the financial sector regarding the difficulty of creating and validating precise models, which could lead to the exclusion of lenders with low-default portfolios from the IRB method (BBA, 2004). As a result, such institutions may need to rely on simpler strategies that require higher regulatory capital.

2.7.1.1 Calibration of Low-Default Portfolios

Out-of-sample testing is a topic that is not frequently investigated in literature related to the LDP problem. Instead, much attention is given to the utilization of different statistical methods to validate LDP models based on the portfolio's monotonic ordering or its sparse number of defaults. The primary focus of these studies is on the accurate validation of LDP models.

A parametric bootstrap was used by Christensen et al. (2004) to generate confidence intervals for PD estimates. Similarly, Hanson and Schuermann (2006) used bootstrap techniques to produce confidence ranges for estimates of the frequency of defaults, albeit some grading grades needed a minimum number of defaults (Pluto and Tasche, 2011). Benjamin et al. (2006) described the procedure of comparing the look-up PD to the weighted average PD of a firm's portfolio. LDPs were cautiously evaluated using a look-up table.

The use of Bayesian approaches to calculate the probability of default (PD) for low default portfolios (LDPs) has been previously researched by other authors. When there is inadequate information on the parameter of interest in tiny data samples, it might be very helpful to include past knowledge (Orth, 2011). A previous distribution for the pertinent parameters based on prior knowledge is one approach to accomplish this (i.e. PD). A probability distribution expressing expert opinion was integrated with a Bayesian approach by Kiefer (2009).

However, because of the incentive to produce unduly optimistic estimates of the priors, this strategy can be time-consuming and may not satisfy regulators (Orth, 2011). Dwyer (2007) additionally employed a modified Bayesian method that disallows the use of expert information to verify the precision of projected PD values. Tasche (2012) also employed comparable uninformed priors.

In order to evaluate model calibration using historical rating transition data, Stefanescu et al. (2009) employed a Bayesian hierarchical framework, while Van Der Burgt (2008) proposed a method based on fitting the cumulative accuracy profile (CAP or Lorentz curve) to a concave function. An empirical Bayes approach was developed by Orth (2011) (Carlin & Louis, 2008), which extracts prior information from other datasets that supplement the initial dataset. For instance, a bank might employ a range of retail loan portfolios to estimate the PD for each individual portfolio using the empirical Bayes approach. This strategy may, however, be more appropriate for sovereign bonds given the availability of additional data from rating organizations like Standard and Poor's.

2.7.1.2 Modelling Low-Default Portfolios

The choice of an acceptable classification approach for credit scoring is still a complex and difficult one. The ambiguity brought on by contradicting studies and research in this area has

been noted by Baesens et al. (2003). Some studies might favor one categorization algorithm over another, while other studies might suggest the exact reverse. Additionally, many of these only examine a few classification algorithms, limited to a few datasets for credit scoring. The fact that many of the datasets are not accessible to the public further hinders replication and verifiability. One problem in the field is that authors often focus on their own approach without making a comparable effort to explore other approaches (Michie et al., 1994). Thomas (2009b) notes that studies aimed at addressing this issue (Xiao et al., 2006) have found that distinctions in the effectiveness of categorization algorithms are usually not statistically significant and are generally modest.

When working with Lack of Default Probability (LDPs), accurately determining the probability of default in the absence of previous defaults, as well as evaluating the predictive performance of a model, are two significant technological challenges (Stefanescu et al., 2009). These challenges arise not only during model validation but also during model creation. To construct a model that addresses the LDP problem, researchers have used various approaches such as making assumptions about the data's ordering, incorporating expert opinion, or using a specific number of historical defaults that have been artificially or naturally generated.

When creating credit scoring models, two common approaches are: (i) statistical models based on a representative sample of data and (ii) expert systems where financial experts choose the parameters. Research conducted in other fields, such as that by Van Gestel and Baesens (2009), suggests that quantitative statistical models outperform human experts, as demonstrated in earlier work by Meehl (1955). However, knowledge-based implementations, like the BVR-I rating system used by the Federal Association of German Cooperative Banks (OeNB / FMA, 2004), can also effectively predict loan applicants' creditworthiness, as shown by Tang and Chi (2005). Explanatory models generated by expert systems, such as those

discussed by Hoffman et al. (2007), can provide the expert with a rationale for approving or denying a particular credit applicant, but our focus is on quantitative methods.

Credit scoring models have been developed through various studies that employ classification techniques such as logistic regression (Westgaard and Van der Wijst, 2001) and neural networks (West, 2000). Brown (2012) provides a non-exhaustive collection of such investigations. These classification methods can be combined to form an ensemble classification method. However, most of these research assumes that the built credit scoring models utilize datasets with a representative number of historical defaults.

There is a lack of research that evaluates the LDP issue. Brown and Mues (2012) conducted a study comparing various categorization techniques on a variety of credit score datasets with different degrees of class imbalance. Eight additional datasets with good:bad ratios ranging from 70:30 to 99:1 was constructed for each of the five real-life credit datasets utilized in the study. The classification methods used were decision trees, the k-nearest neighbor algorithm (k-NN), random forests (Friedman, 2002), logistic regression, least square support vector machines (linear kernel), neural networks (multi-layer perceptron), linear discriminant analysis, quadratic discriminant analysis, and gradient boosting (Breiman, 2001). The study used Friedman's test and Nemenyi's post hoc tests to evaluate the performance of these strategies using the AUC. The results showed that gradient boosting and random forest classifiers, which are more sophisticated algorithms, produced "very good performance" at severe degrees of class imbalance. However, even at levels of extreme class imbalance, the classification approaches of logistic regression and linear discriminant analysis produced findings that were reasonably competitive.

One-class classification algorithms are designed to use only labeled examples from the target class as training data, as obtaining samples from other classes can be too expensive or rare.

Such approaches have been successfully applied to real-world problems like fault detection. However, there is a lack of literature that examines the effectiveness of one-class classification approaches on low default portfolios. A previous study by Juszczak et al. (2008) compared one-class and two-class classification algorithms for identifying fraudulent plastic card transactions and found that two-class classifiers perform better than one-class classifiers when training and test objects are drawn from the same distribution. Another study by Krivko (2010) proposes a method for combining one-class and two-class classifiers to identify suspicious activity in debit card transaction data.

The LDP issue has been identified by Basel II legislation as a continuing problem in credit scoring. As a result of the training data being unrepresentative of the notion that must be taught, supervised classifiers designed to tackle such challenges generally perform poorly. To accurately determine the potential and restrictions of applying OCC approaches to the LDP problem, more research is required.

2.7.2 Behavioural Scoring

Behavioral scoring can be categorized into two main approaches: those that use static information on a customer's past behavior, and those that incorporate dynamic elements. Thomas et al. (2001) provides an overview of the methods and goals of behavioral scoring, focusing on techniques that consider dynamic elements of client behavior, such as Markov models (Malik & Thomas, 2012). The use of behavioural scoring methods that consider changing client behavior is not covered in this thesis. Financial organizations must consider how specific operational policies, or policy bias, may affect client behavior. Thomas (2009a). For instance, a quick response program lowers the likelihood that consumers will continue to miss loan payments and later be labeled as bad. Last but not least, the outcome period needs to be large enough to include a meaningful sample of errors from which to create a reliable behavioral scoring model.

According to Thomas et al. (2001), customers are often categorized as “good” or “bad” using methods similar to those used in application scoring. Creating and implementing behavioral scorecards, like application scoring, pose various challenges, such as identifying and accounting for different population segments (Bijak and Thomas, 2012), ensuring feature correlation (Tsai, 2009), handling class imbalance (Burez and Van den Poel, 2009), and determining the appropriate sample size (Crone and Finlay, 2012).

Constructing behavioral scoring models requires practitioners to make critical decisions regarding several key characteristics. These include determining the scope of the historical dataset used to estimate client performance, the time frame for making accurate predictions, and the criteria for identifying loan defaulters. However, the credit score literature provides limited guidance on how to address these issues definitively.

Recent literature has only started to publish empirical studies on the effects of different temporal horizon sizes on classifier performance. In the context of identifying loan defaulters using survival analysis (Andreeva, 2005), much of the current research on selecting relevant time periods in behavioral scoring is being conducted. Duration models are primarily concerned with predicting when a default will occur, rather than if it will occur (Banasik et al., 1999).

2.7.3 Artificial Data

The field of credit scoring has seen numerous studies over the past decade assessing the effectiveness of different models used to create credit scorecards (Baesens et al., 2003; Chen et al., 2011; West, 2000). These studies have used data from two sources: publicly accessible datasets from the University of California Irvine (UCI) Machine Learning Repository, including Australian, German, and Japanese datasets from financial institutions, which raises some questions (Asuncion and Newman, 2007).

According to Salzberg (1997), the UCI repository serves several important purposes. It allows researchers to verify published results and assess the effectiveness of new algorithms by comparing them to prior results. However, while the repository is a valuable resource, some researchers caution against relying too heavily on it (Martens et al., 2011; Salzberg, 1997; Soares, 2003). One potential issue with the repository is that it can lead to over-fitting, as researchers may be tempted to create algorithms that are tailored to the datasets in the repository due to their familiarity with them (Salzberg, 1997). Consequently, researchers may overlook the importance of understanding the specific conditions in which an algorithm performs optimally (Soares, 2003). To avoid this issue, it is beneficial for researchers to incorporate a variety of data sources and not solely rely on the UCI repository.

Datasets available in the UCI repository have been criticized for their limited representation of real-world scenarios, as well as for capturing only a subset of the circumstances that can occur (Drummond and Holte, 2005; Saitta and Neri, 1998). For instance, the credit application datasets from Australia and Germany have dissimilar class distributions, and their sizes do not reflect those commonly found in contemporary practice. Such differences may stem from the way the datasets were constructed, which raises concerns about the data collection process (Drummond and Japkowicz, 2010). It is necessary to exercise caution when assuming that the sampling of these datasets is random, given the various class distributions. Moreover, the applicability of UCI data can be called into question based on the inclusion of certain attributes. For example, it is uncertain whether a telephone component was used in the German dataset during the era of the widespread use of cell phones. Therefore, it is important to avoid drawing excessive conclusions from experiments that solely rely on these datasets.

It is preferred for researchers to use information obtained from various sources. This can be achieved by combining different real-world datasets collected from financial institutions,

whether directly or indirectly. However, acquiring real-world data can be challenging without proper tools. Fischer and Zigmond (2010) listed several criteria that prevent academics from sharing data, including negative career impact, limited resources, and property rights and legal issues.

Negative career impact can be a concern for researchers as their career often depends on their ability to publish. Sharing data may result in their dataset being used in long-term projects resulting in several publications. However, if a researcher is forced to share data after their first publication, the likelihood of producing additional publications may be reduced, particularly if the data is acquired by a research group with more funding and resources.

Limited resources can also hinder data sharing as providing data in a format that other researchers can use may require additional resources, leaving less time and money for the originator's own research projects. Additionally, some datasets may require upkeep and updating, which may not be practical once a researcher has completed their work.

Property rights and legal issues can also prevent data sharing. For instance, maintaining customer anonymity is crucial for financial firms, and the aggregation of numerous datasets may compromise anonymity. The authors believe that these restrictions will continue to exist due to the absence of incentives for data originators to share their data and the stringent requirements of data protection regulations (Bergkamp, 2002).

Without publicly accessible datasets, credit scoring will not be open to the larger data mining community.

Illustrating the advantages of employing artificial data to get over the aforementioned challenges. The ability to manipulate the many characteristics utilized in the evaluation process makes generated data superior to real-world data. According to Malin and Schlapp (1980), the researcher can include individuals having the freedom to determine the quantity

of data samples to include, as well as the specific distribution of data. Additionally, individuals may introduce noise into their data with a known standard deviation and can examine the effects of alternative variables by utilizing synthetic data. By employing this strategy, the researcher can create exact experiments that evaluate the effectiveness of algorithms under certain conditions of interest (Scott and Wilkins, 1999).

It is crucial to understand that manufactured data cannot duplicate the inherent unpredictable nature of real-world data (such as unforeseeable changes in personal circumstances or natural disasters). The unpredictability of artificial data is due in part to the structural complexity resulting from unaccounted-for external factors (Scott and Wilkins, 1999). To impose structural regularity on fabricated data, a fixed distributional model must be used, as stated by Japkowicz and Shah (2011). However, generating artificial data can introduce unintentional bias towards a specific classification technique, which may not accurately reflect real-world circumstances. Therefore, researchers should exercise caution when analyzing results derived from fabricated data, as they may not be comparable to real-world scenarios (Japkowicz & Shah, 2011). Despite these challenges, synthetic data can help researchers to conceptualize a problem more precisely and evaluate existing and novel methodologies within the established limits and assumptions of the data (Scott and Wilkins, 1999).

2.7.3.1 Studies on Artificial Data

Researchers frequently evaluate the effectiveness of suggested credit scoring categorization techniques using false data. Creating data from a p-dimensional multivariate normal distribution with a predetermined mean vector and covariance matrix for each class is one typical method. This method was used by Hand and Adams (2000). Two univariate Gaussian distributions were employed in an even simpler manner by Hoffmann et al. (2007) and Kelly et al. (1999), which facilitated model visualization. Moreover, investigations, including one

by Martens et al. 2010, have used publicly accessible artificial datasets like Ripley's dataset (Ripley, 1994).

While artificially created data can be helpful for demonstrations, it is important to note that the findings drawn from them may not necessarily apply to real-world situations. To address the shortage of real-world data, there are two general approaches. The first involves using existing real-world data as a "seed" to create artificial data, while the second strategy involves generating artificial data without relying on any real-world data. An example of the first approach is the Synthetic Minority Oversampling Technique (SMOTE), which produces synthetic cases to oversample the minority class, rather than simply oversampling with replacement. This technique was introduced by Chawla et al. in 2002.

Andersson et al. (2011) note the use of fraudulently manufactured credit ratings by US authorities to validate credit scoring algorithms as an example of the second strategy. These ratings are associated with systematic characteristics, such as the unemployment rate. The literature describes several specialized dataset generators beyond credit scoring, including the work of Scott and Wilkins (1999), who discuss two artificial data generators. The first is based on the multivariate normal distribution, whereas the second is inspired by fractal techniques used to create artificial landscapes. Moreover, Myers (1999) created celsim, which is applied in the genome assembly procedure to produce a DNA sequence with repeat structures and polymorphic variants in accordance with user specifications. The IBM dataset generator (Srikant, 1994), which mimics a retail setting and generates market baskets of commodities, is another illustration.

Alaiz-Rodrguez and Japkowicz (2008) developed a medical simulation tool that can anticipate the prognosis of patients one month after being diagnosed with influenza. The simulation employs several variables to describe each patient, including their age, the severity

of their influenza, their overall health, and their social standing. While one variable, overall health, is dependent on age and social status, the remaining three variables are independent. The tool generates data based on the user's input of prior probabilities for each variable, and the prognosis is determined based on all four variables. Users can modify the prior probabilities for each variable to simulate various scenarios, such as an outbreak of a more virulent strain of influenza, changes in the population demographics, or a less affluent population.

While there are some basic frameworks available for generating data (Atzmueller et al., 2006; Melli, 2007), these tools are limited in their ability to capture the complex nuances of a specific subject, such as credit scoring. As far as I am aware, there is currently no established method for fabricating realistic credit scoring information.

2.8 Bankruptcy Prediction

2.8.1 Bankruptcy

In today's economy, bankruptcy is a natural occurrence and a component of the economic base. One of the reasons limited liability firms, which allowed equity investors to solely be accountable for their own shares and thereby reduce personal risk, first came into being in England during the seventeenth century was bankruptcy. Due to incentives that were misaligned, this development opened the door to the prospect of separating owners from management, which created additional issues including principal-agent dilemmas where management and stockholders had conflicting interests. The issue also involves adverse selection, where stock investors and management have differing information as a result of information asymmetry. As a result, the investors are forced to rely on management information and run the danger of being duped. The information asymmetry is reduced by developing reliable models that assist investors in determining the risk.

By eliminating businesses that lack significant competitive advantages, such as those who offer dated services or goods or other drawbacks, bankruptcy strengthens the market. In such cases, the estate invests the leftover values at the expense of the equity holders in more productive enterprises. Therefore, the bankruptcy incident is illustrative proof for creative destruction, which was first put out by Joseph Schumpeter (Reinert and Reinert, 2006). The growing accountability that corporations confront from their stakeholders is the main driver behind improving bankruptcy predictions. Misclassifying businesses, particularly bankrupt ones, could have very expensive consequences. As a result, considering the development of effective bankruptcy prediction models is justified because doing so could help to cut down on the expenses associated with misclassification (Chen et al., 2011).

2.8.2 Early Application of Bankruptcy Prediction

Credit analysis and bankruptcy predictions have been practiced for a very long time. The first proof dates to likelihood estimations in the 1890s (Correia, 2018). Private banks used the research largely to offer lending services to organizations based on their creditworthiness, which helped to popularize the idea of ratio analysis. In the early 1900s, the framework became more standardized., which helped credit men become more prevalent (Correia, 2018). To generate public interest and launch a public dialogue on credit risk, the Federal Reserve in the United States issued its first ratio study of the federal bank in 1919. (Wall, 1919; Correia, 2018).

Beaver (1966) was one of the first pioneers in bankruptcy prediction. He used a univariate analysis to discover substantial differences between two categorical groups, bankrupt and non-bankrupt enterprises, in a range of variables. He conducted his investigation over a five-year period on a sample of 706 businesses. The sample was chosen to exclude specific industries, and for all the years, there was roughly a 50 percent split between these two categories. The 30 variables that were chosen were split up into five distinct subgroups and

arranged according to qualities. These subgroups were tied to several aspects of the financial structures of the companies, including cash flows, net income ratios, turnover, and acid tests. He developed four hypotheses based on this methodology to pinpoint troubled businesses and suitable cutoff points for each of these ratios. These limits are now accepted as general guidelines for the aforementioned ratios. Beaver also pioneered the practice of systematically rating businesses' creditworthiness using financial data. Later, he also presented alternate ratios, which changed the way investors perceive distress (Beaver, 1968).

2.9 Theoretical Framework

This section provides a theoretical framework on the models the machine learning algorithms for credit scoring and bankruptcy prediction are built on.

2.9.1 Support Vector Machine

The approach's fundamental idea is to place observations on a p-1-dimensional hyperplane and utilize those observations to steer vectors across the hyperplane according to the various responses. The goal is to divide the feature space between the responses by doing this. This can be seen by contrasting it with the two-dimensional predictor space in classification trees. According to James et al. (2017) and Hastie et al. (2009), the hyperplane is for p dimensions by definition.

$$\beta_0 + \beta_1 X_1 + \cdots + \beta_\rho X_\rho = 0 \ldots\ldots\ldots (1)$$

where β_i's are coefficients and X_i's are hyperplane points. Theoretically, the hyperplane can be divided into areas based on the response value if the data is separable. Inherently producing perfect predictions, this would result in linear vectors throughout the hyperplane that were tailored to the training set of data. If this is accurate, it means that there are an endless number of sites on the hyperplane where the support vector might be drawn because there are an infinite number of deviations that will not cause this separation to fail. Now, the

issue is which of these infinite vectors represents the real vector. The answer is to draw the support vector at the same distance from the two classes on the hyperplane, leaving an equal margin on either side of the vector.

However, data rarely can be separated. By permitting a soft margin of error, the support vector classifier resolves this. This suggests that certain misclassification or margin violations on the hyperplane are possible (James et al., 2017). By assigning a cost parameter, c, as a result of this simplification, it is possible to compensate the model for going over the margin. To provide the best bias-variance trade-off, this hyper parameter is essential.

By extending the feature space with kernels, the support vector machines improve on the solutions the support vector classifier provides. The stretched feature space can be compared to the larger feature space to make things simpler and ensure that the vectors are linear. However, in practice, the vectors would not be linear without this stretch. The approach is to employ kernels to reduce the complexity of the vectors' curvature because this process requires a lot of computation.

Irrespective of where you are on the hyperplane, the algorithm will calculate the inner products. It uses a term known as the linear kernel (James et al., 2017). In general, the kernel can be seen of as a vessel that expresses the similarity between two observations on the hyperplane. The chosen distribution of the decision border is related to the kernel, which is a function. There are numerous types of kernels; however, polynomial, and radial kernels are the most popular. Since it is uncommon to have a clear understanding of the distribution in practice, one of these two is frequently chosen. The following is a description of how these two kernels work (James et al., 2017; Hastie et al., 2009):

$$K(x_{\$}, x'_{\$}) = \text{1222223222224}^{polynom\$al}_{(1 + \sum_{y)1} x_{7y}, x'_{7y})^{d}} \quad \text{or}$$

$$K(x_{\$}, x'_{\$}) = exp\left(-\gamma \sum_{y)1}^{rad\$al} (x_{7y}, x'_{7y})\right) \text{ ..... (2)}$$

where γ is a constant that is positive. Calculating the observation's closeness is crucial for the radial kernel. As a result, the accepted practice is to first determine the Euclidean distance between the training and testing locations before ranking them. If d = 1, the polynomial kernel behaves differently and becomes linear, resulting in the support vector classifier approach. In order to calculate the support vector machine (James et al., 2017), we use the radial kernel:

$$f(x) = \beta_0 + \sum_{\$\in 3} \alpha_{\$} K(x, x_{\$}) \text{ ..... (3)}$$

where ς denotes a vector of indices depending on the coordinates of the points, and α can be thought of as a weight, and / or each pair of training observations, α and β are determined by the inner products (James et al., 2017). Using kernels has the advantage of providing superiority in comparison to just expanding the feature space of the original variables, computation complexity, and time.

Vanilla SVM has been the focus of this section thus far, however training with large amounts of data makes this method computationally expensive. Therefore, a less computationally intensive SVM variant termed liquidSVM is adopted and implemented as suggested by Steinwart and Thomann (2017). This model is based on a similar theoretical foundation as the one described above. However, this approach introduces certain additional qualities that are quite beneficial. The algorithm controls each chunk separately before they are ensembled, which is the key advantage. The algorithm de-aggregates the feature space into many chunks. This makes computations less difficult and speeds up model training. The chunks are already processed, which brings us to the second advantage, speeding up cross-validation while also performing hyper parameter tuning. The tuning is carried out over an adaptable grid of

various parameter values, and the algorithm selects the ideal tuning settings from a list of available choices.

2.10 Empirical Literature Review

The significance of machine learning approaches for credit scoring and bankruptcy prediction in banking and finance is empirically examined in this work. Case-based reasoning, decision trees, artificial neural networks, and support vector machines have all been demonstrated to be effective techniques for credit rating in several studies. These techniques include case-based reasoning (Buta 1994; Shin and Han 2001), artificial neural networks (Desai et al. 1996; West 2000), decision trees (Hung and Chen 2009; Makowski 1985), and SVM (Baesens et al. 2003; Huang et al. 2007; Schebesch and Stecking 2005). Artificial intelligence systems, in contrast to statistical approaches, automatically extract knowledge from training samples without assuming the distribution of the data. Previous studies indicate that when it comes to credit scoring issues, Machine learning frequently outperforms statistical techniques, especially for nonlinear categorization patterns (Huang et al. 2004; Van Gestel and Baesens 2009). A study by Yu et al. (2008) suggests that there is no single best machine learning technique because the best approach depends on the problem's specifics, the data structure, the predictors used, how well the classes can be divided using those predictors, and the classification analysis's intended use. This shows that even though machine learning solves a bigger problem of reducing human error, there is no perfect system for credit scoring and bankruptcy prediction.

Comparing the efficiency of various machine learning techniques for credit scoring and bankruptcy prediction, Abdykalykova (2020) in her study concluded that Gradient Boosting, Support Vector Machine and Naïve Bayes were the models that can be adapted by financial institutions. On the percentage of accuracy of the models, Gradient Boosting produce an

accuracy of 81.2 percent, followed by Support Vector Machine at 77 percent and Naïve Bayes at 69 percent.

Credit scoring is an essential task in banking and finance that helps lenders assess the risk associated with a loan applicant. Traditional credit scoring methods rely on financial ratios and historical data. However, the emergence of big data has provided an opportunity to enhance credit scoring accuracy using machine learning techniques.

According to a study by Zulfiqar et al. (2022), machine learning models have been used to predict credit risk in various financial institutions. The study evaluated different machine learning algorithms, including decision trees, support vector machines (SVM), and logistic regression. The study found that SVM was the most accurate algorithm for credit scoring, with an accuracy of 91.56%.

Another study by Elhoseny et al. (2019) evaluated deep learning-based models for financial distress prediction. The study used a long short-term memory (LSTM) model to predict financial distress for firms listed in the S&P 500 index. The study found that, the LSTM model outperformed traditional models such as logistic regression and decision trees, achieving an accuracy of 87.05%.

In another study, Boughaci and Alkhawaldeh (2018) compared the performance of logistic regression, decision trees, random forests, and SVM for credit scoring. The study used a dataset of loan applicants from a Jordanian bank. The study found that SVM outperformed the other algorithms, achieving an accuracy of 85.2%.

Chen and Li (2019) proposed a novel approach to credit scoring using an ensemble of random forest and extreme gradient boosting (XGBoost) algorithms. The study evaluated the proposed model on a dataset of Chinese loan applicants. The study found that the proposed

model outperformed traditional models such as logistic regression, decision trees, and random forests, achieving an accuracy of 78.25%.

In a similar study, Tahir et al. (2021) proposed an ensemble of XGBoost and SVM algorithms for credit scoring. The study used a dataset of Pakistani loan applicants. The study found that the proposed model outperformed traditional models such as logistic regression, decision trees, random forests, achieving an accuracy of 82.1%.

In another study, Liu et al. (2018) proposed a credit risk assessment model based on deep belief networks (DBN). The study evaluated the proposed model on a dataset of Chinese loan applicants. The study found that, the proposed model outperformed traditional models such as logistic regression, decision trees, random forests, achieving an accuracy of 83.5%.

Furthermore, Togbenu et al. (2022) proposed a hybrid machine learning model that combines the k-nearest neighbor algorithm (k-NN) with the particle swarm optimization (PSO) algorithm. The study evaluated the proposed model on a dataset of Nigerian loan applicants. The study found that the proposed model outperformed traditional models such as decision trees, and random forests, achieving an accuracy of 81.8%.

Naïve Bayes is a probabilistic algorithm that assumes independence among the features in the input data. This algorithm is widely used in text classification, sentiment analysis, and spam filtering, among others. In a study conducted by Y. Tang et al. (2018), the Naïve Bayes algorithm was compared with other machine learning algorithms in predicting stock prices. The study showed that the Naïve Bayes algorithm outperformed other algorithms in terms of accuracy, precision, and recall.

Similarly, in a study conducted by R. A. Mohammed et al. (2021), the Naïve Bayes algorithm was used to predict the creditworthiness of borrowers. The study showed that the Naïve

Bayes algorithm outperformed other algorithms such as decision tree and k-nearest neighbor (KNN) in terms of accuracy and speed.

SVM is a popular machine learning algorithm used in various applications such as text classification, image recognition, and bioinformatics. SVM works by finding the optimal hyperplane that separates the data into different classes. In a study conducted by A. Alam and R. Kumar (2018), SVM was compared with other machine learning algorithms in predicting stock prices. The study showed that SVM outperformed other algorithms such as decision tree and KNN in terms of accuracy.

Similarly, in a study conducted by Y. Li et al. (2017), SVM was used to predict the probability of loan default. The study showed that SVM outperformed other algorithms such as logistic regression and decision tree in terms of accuracy and AUC.

Gradient Boosting is an ensemble learning algorithm that combines multiple weak learners to form a strong learner. Gradient Boosting has been widely used in various applications such as credit risk analysis, image classification, and anomaly detection. In a study conducted by J. Z. Gao et al. (2021), Gradient Boosting was compared with other machine learning algorithms in predicting credit risk. The study showed that Gradient Boosting outperformed other algorithms such as Random Forest and SVM in terms of accuracy, precision, recall, and F1-score.

Similarly, in a study conducted by Y. Liu et al. (2019), Gradient Boosting was used to predict the credit risk of peer-to-peer lending. The study showed that Gradient Boosting outperformed other algorithms such as SVM and KNN in terms of accuracy, AUC, and Gini coefficient.

2.11 Gap in Research

Based on the studies mentioned above, it appears that there is a significant amount of research on the use of various machine learning algorithms for credit scoring and credit risk analysis. However, there may be a gap in the literature regarding the comparison of these algorithms in different contexts. For example, according to Zhao et al. (2020), while there have been numerous studies evaluating different algorithms for credit risk analysis, there is a lack of research on the use of these algorithms in the context of emerging markets. Similarly, Zhang and Wang (2020) suggest that there is a need for more research on the use of machine learning algorithms in credit risk analysis for small and medium-sized enterprises (SMEs).

There is limited research on the impact of interpretability and explainability of machine learning models on credit scoring and bankruptcy prediction: While machine learning models have shown promising results in predicting credit risk and bankruptcy, they are often considered as black boxes due to their complex nature. Therefore, it is important to investigate the impact of interpretability and explainability of machine learning models on their performance and adoption in the banking and finance sector.

Limited research on the scalability and generalizability of machine learning models for credit scoring and bankruptcy prediction: Machine learning models for credit scoring and bankruptcy prediction often require large amounts of data and computational resources, which can limit their scalability and generalizability in practice. Therefore, it is important to investigate the scalability and generalizability of machine learning models for credit scoring and bankruptcy prediction and to develop efficient and robust algorithms that can handle large-scale and diverse datasets.

Furthermore, some scholars have pointed out the need for research on the interpretability and transparency of machine learning models for credit scoring and credit risk analysis. For

example, Soltani et al. (2020) argue that there is a lack of transparency in the decision-making process of black-box models such as deep neural networks, which can lead to ethical concerns and regulatory challenges. Similarly, Chen et al. (2021) suggest that the interpretability of machine learning models is crucial for ensuring the fairness and accountability of credit decisions.

Lack of interpretability: Along with transparency, the interpretability of machine learning models is essential for credit scoring and bankruptcy prediction. It is important to be able to explain the factors that contribute to a particular score or prediction.

Lack of real-time implementation: Many of the existing studies are based on historical data and are not designed to be implemented in real-time. It is essential to develop models that can be used in real-time to improve the efficiency and effectiveness of credit scoring and bankruptcy prediction.

Generally, while there have been many studies on the use of machine learning algorithms for credit scoring and credit risk analysis, there may be a need for more research on the application of these algorithms in different contexts and on the interpretability and transparency of these models.

2.12 Summary

This chapter detailed the concept of machine learning and machine learning techniques, credit scoring and bankruptcy predication. The chapter covered various literature on machine learning technique for which authors such Yu et al. (2008) suggests that there is no single best machine learning technique. However, Baesens et al. (2003) on the other hand suggests that the SVM is one of the most effective amongst other techniques. The various opposing views of the authors captured in this literature have informed the purpose of this study. The objective is this study is geared towards selecting the best machine learning technique and

identifying their strength and weakness. To achieve this, the researcher measured the accuracy of the three machine learning techniques; Naïve Bayes, Support Vector Machine, and Gradient Boosting and identify the weakness and strength of the techniques.

In this empirical review, comparing Naïve Bayes, Support Vector Machine, and Gradient Boosting was based on their performance in various applications. The studies reviewed showed that these algorithms have been successful in predicting stock prices, creditworthiness, loan default, and credit risk. While each algorithm has its strengths and weaknesses, it is important to choose the algorithm that is best suited for the specific problem at hand.

Naïve Bayes, SVM, and Gradient Boosting are powerful machine learning algorithms that can be used for classification tasks. Each algorithm has its strengths and weaknesses, and the choice of algorithm depends on the specific problem at hand. Naïve Bayes is a simple algorithm that is easy to implement, while SVM is a powerful algorithm that can handle non-linearly separable data. Gradient Boosting is a powerful algorithm that can handle complex non-linear relationships between variables. By understanding the strengths and weaknesses of these algorithms, machine learning practitioners can make informed decisions when choosing an algorithm for a specific problem.

CHAPTER THREE

RESEARCH METHODOLOGY

3.1 Introduction

This chapter presents a detailed and systematic process that was adopted to achieve the objective of the study. The main topics discussed in this chapter are the description of the models and parameters, assumptions of the models, the study dataset, the data analysis strategy, and the determinant for selecting the best technique.

3.2 Selected Models

For the purpose of this study, three (3) models, that is, Naïve Bayes, Support Vector Machine and Gradient Boosting were used for the analysis. The selected models for this study are briefly described in 3.2.1 below.

3.2.1 Description of the Models

3.2.1.1 Naïve Bayes

The use of Bayes' theorem with strong (naive) independence assumptions between the features forms the basis of the family of simple probabilistic classifiers known as Naïve Bayes classifiers. These classifier types are incredibly scalable and require a set of parameters proportional to the number of features or predictors in a learning problem. For particular types of probability models, supervised learning environments can be used to teach Naïve Bayes classifiers very successfully. While parameter estimation for Naïve Bayes models often uses the maximum likelihood method, it is possible to use the model without adopting Bayesian probability or using any Bayesian procedures.

Joyce, James (2003) described Bayes theorem as the probability of an event based on prior knowledge of conditions that might be related to an event. The theorem states that:

$$P(y|X) = \frac{P6X7y8*P(y)}{P(<)} \text{ ..... (4)}$$

where $P(y|X)$ means Posterior probability – probability of hypothesis y on the observed event X.

$P(X|y)$ means Likelihood probability – probability of the evidence given that the probability of a hypothesis is true.

$P(y)$ means Prior probability – probability of hypothesis before observing the evidence.

$P(X)$ means Marginal probability – Probability of evidence.

To calculate the posterior probability, the likelihood probability is formed using the chain rule:

$$P(X|y) = P(x_1, x_0, \dots, x_n|y) \text{ ..... (5)}$$

$$= P(x_1|x_0, \dots, x_n, y)* P(x_0|x_=, \dots, x_n, y) \dots P(x_n|y)$$

$$P(X|y) = P(x_1|y) * P(x_0|y) \dots P(x_n|y) \text{..... (6)}$$

The assumption of conditional independence holds, making the conditional probabilities independent of each other from equation (6) we have:

$$P(y|X) = \frac{P6x_1 7y8*P6x_0 7y8 \dots P6x_n 7y8*P(y)}{P(x_1)*P(x_2)\dots*P(x_n)} \text{ ..... (7)}$$

Rewriting the posterior since the denominators are constant:

$$P(y|x_1, x_0, \dots, x_n) \text{ a } P(y) \prod_{\$)1}^{n} P(x_\$|y) \text{ ..... (8)}$$

where $P(y|x_1, x_0, \dots, x_n)$ is the posterior probability of class y given the feature $x_1, \dots, x_n$, $P(y)$ is the prior probability of the class y, and $P(x_\$|y)$ is the probability of the feature x given class y.

Now, the maximum posterior or MAP decision rule is:

$$y = argmax_y \, P(y) \prod_{\$)1}^{n} P(x_\$|y) \text{ ..... (9)}$$

3.2.1.2 Support Vector Machine

Support Decision planes, which specify decision boundaries, serve as the foundation for vector machines. A decision plane is a diagram that distinguishes between a collection of objects with various class memberships. SVM is essentially a classier technique that performs classification tasks by dividing instances of different class labels into hyperplanes in a multidimensional space. SVM, which permits both regression and classification tasks, can handle a variety of continuous and categorical data.

The SVM optimization model is based on the transformation of a mathematical function by a different function known as the Kernel, as stated by Noble (2006).

Given a training set $(x_1, y_1), (x_0, y_0), \dots, (x_k, y_k), .., (x_n, y_n)$ where $x_k \in R^m$ and $y_k \in \{-1, +1\}$ and (x_k, y_k) elements correspond to a point in a high-dimensional space. Consider the case where a hyperplane can divide all the points $w^T w + b = 0$; then it is natural to build a linear classifier in the manner shown below (Vapnik 1998).

$$y(x) = sgn(w^T x_k + b) \quad \dots\dots\dots\dots (10)$$

where $y(x)$ - decision function for classifying a new sample x,
w - weight vector perpendicular to the hyperplane,
b - bais term.

When the data of the two classes are separable, then we say that:

$$\begin{cases} w^T x_k + b \geq +1, & if\ y_k = 1 \\ w^T x_k + b \leq -1, & if\ y_k = -1 \end{cases} \quad \dots\dots\dots\dots\dots\dots\dots (11)$$

The following compact form is obtained from equation (11),

$$y_k(w^T x_k + b) \geq 1, k = 1,2, \dots, N \quad \dots\dots\dots\dots (12)$$

Given that all training data points must be accurately classified, SVM determines the best separation hyperplane to maximize the margin and presents the following optimization problem:

$$\min_{C,b} Z = \frac{1}{0} w^T w \quad \ldots\ldots\ldots\ldots (13)$$

$$\text{subject to } y_k(w^T x_k + b) \geq 1, k = 1,2, \ldots, N.$$

Finding a hyperplane that can accurately separate all the data points is often challenging for most real-life scenarios. Therefore, an extension of linear SVM to a non-separable case is made to introduce an additional slack variable $\xi_t \geq 0$, indicating the misclassification error. Two objectives are put together to find an optimal separating hyperplane that is minimizing the classification error and maximizing the margin. This problem is formulated as:

$$\min_{C,b,F} Z = \frac{1}{0} w^T w + C \sum_{k)1}^{N} \xi_k \quad \ldots\ldots\ldots\ldots\ldots (14)$$

$$\text{subject to } y_k(w^T \phi(x_k) + b) \geq 1 - \xi_k , k = 1,2, \ldots, N.$$

where k = 1, …., N, C is known as the classification cost (regularization parameter), $\xi_k \geq 0$ are the margins of error related to classification cost C, y_k are the classification in the training set and $\phi(x_k)$ transforms space R^m.

But the kernel function is introduced when the problem is nonlinear. Flavio Barboza, Herbert Kimura, and Edward Altman (2017) stated that a given a kernel function $K(x) = K(x_k, x_l)$, which satisfies $K(x_k, x_l) = \phi(x_k)^T . \phi(x_k)$. The kernel function is predetermined in the algorithm and a solution to the optimization problem above.

3.2.1.3 Gradient Boosting

To minimize a loss function, the functional gradient method known as Gradient Boosting continually chooses a function that points in the direction of a weak hypothesis or a negative gradient. A powerful predicting model is created using the gradient boosting classifier by combining many weak learning models. According to Jake Hoare on Display R website, the desired results for each case of prediction will vary, depending on how much altering a case's prediction affects the total amount of prediction error. That is:

- the next target outcome of the case is a high value if a little adjustment in the forecast for a given case results in a significant decrease in inaccuracy. The inaccuracy will be decreased with fresh model predictions that are close to their goals.
- the next target outcome of a case is zero if a small change in the prediction makes no difference in the error. This forecast cannot be altered to reduce the error.

The model of Gradient Boosting is built in a stage wise fashion, represented by Candice Bentéjac, Anna Cséögő and Gonzalo Martínez-Muñoz (2020):

Given a training dataset:

$$D = \{X_{\$}, Y_{\$}\}_{1}^{N},$$

where $X_{\$}$ is the feature vector and $Y_{\$}$ is the target variable.

Finding an approximate solution is the goal of gradient boosting, $\hat{F}(x)$, of the function F*(x), which maps x to their outputs y.

Minimizing the expected value of the Loss function

$$L(y, F(x)) = \sum_{\$=1}^{N} L(y_{\$}, \alpha),$$

where N is the number of training samples, $\alpha = F(x_{\$}, \theta)$ is the current model prediction for the ith sample and L is the loss function measuring the difference between the true target $y_{\$}$ and the predicted target

An additional approximation of F*(x) as a weighted sum of functions

$$F_m(x) = F_{m-1}(x) + \rho_m h_m(x) \quad \text{.....(15)}$$

where $F_m(x)$ is the model's prediction at iteration m, $F_{m-1}(x)$ is the model's prediction at iteration $m-1$, ρ_m is the learning rate, and $h_m(x)$ is the weak learner's prediction at iteration m.

The above is the model of ensemble. To construct the iterative approximation, we obtain a first constant approximation of the function F*(x):

$$F_0(x) = \sum_{\$=1}^{N} L(y_{\$}, \alpha) \quad \text{..... (16)}$$

Subsequently the models are expected to minimize:

$$(\rho_m, h_m(x)) = \underset{\rho, \alpha}{argmin} \sum_{\$)1}^{N} L\acute{E}y_\$, F_{m-1}(x_\$) + \rho h_m(x_\$)\tilde{N} \quad \text{.....(17)}$$

Note: Each h_m is perceived as a greedy step in gradient descent optimization for F*.

The value of ρ_m is computed by solving a line search optimization problem.

The model, h_m is trained on a new dataset $D = \{X_\$, r_{m\$}\}_{i=1}^{N}$, where the pseudo-residuals, $r_{m\$}$ (the negative gradient of the loss function with respect to the current model's prediction) is calculated by:

$$r_{m\$} = \grave{a}^{\frac{\mathrm{I}L6y_\$, F(x)8}{\mathrm{I}F(x)}}\hat{a}_{F(x))F_{m\&1}(x)} \quad \text{..... (18)}$$

The iteration process ends prematurely, if the model h_m fits the pseudo-residuals exactly, leaving the pseudo-residuals to be zero.

3.2 Interpretation of Model Parameters

In the analysis of the credit datasets, the chosen machine learning model were Naïve Bayes, Support Vector Machines and Gradient Boosting. The following parameters were used for the models:

- Number of trees (n_estimators): This parameter controls the number of decision trees used in the model. A higher number of trees increases the model's complexity and can lead to overfitting. In this analysis, the number of trees was set to 100.

- Maximum depth of tree (max_depth): This parameter controls the maximum depth of the decision tree. A higher maximum depth will make the model more complex and can lead to overfitting. In this analysis, the maximum depth was set to 10.

- Minimum samples split (min_samples_split): This parameter controls the minimum number of samples required to split an internal node. Higher minimum samples split will

make the model more robust to noise in the data. In this analysis, the minimum samples split was set to 2.

- Minimum samples leaf (min_samples_leaf): This parameter controls the minimum number of samples required to be at a leaf node. A higher minimum samples leaf will make the model more robust to noise in the data. In this analysis, the minimum samples leaf was set to 1.

- Number of features considered at each split(max_features): This parameter controls the number of features to be considered when looking for the best split. In this analysis, the max_features parameter was set to sqrt(n_features) which is considered as a good heuristic.

3.3 Assumptions of the Model

The assumptions vary from model to model. Vishal Mendekar published on KD nuggets site on the assumptions of machine learning. For Naïve Bayes, assumption of conditional independence exist, Support Vector Machines assumes data is independent and identically distributed and Gradient boosting assumes that encoded integer value for each variable has ordinal relation and deals with missing values. These assumptions were also affirmed by Aman Kharwal (2021).

3.4 Study Dataset

The datasets chosen for this study were chosen under the conditions that (i) they had previously been used in credit scoring research, and (ii) other academic researchers could access them to verify the reproducibility of experimental findings. The chosen datasets have been utilized in past research on predicting corporate bankruptcy, screening individual revolving credit product applicants, and screening retail loan applications. This research

includes Hand (2009), Mozina et al. (2007), Tsai (2009), West (2000) and Xie et al. (2009), among others. The credit data sets for Australia, Germany, and Japan that I selected are publicly available through the UCI Machine Learning repository.

The dataset of the selected countries downloaded from the UCI machine learning repository, that is, Australia, Germany, and Japan contain instance of 690, 1000, and 125 respectively. It is important to note that the variables contained in the dataset, such as Age, Job, Housing, Accounts, Credit amount, Duration of credit will be significant for this study.

3.5 Data Description

The data description used in this study follows an order done by Zurada, Jozef; Kunene, Niki; and Guan, Jian (2014):

The Australia dataset describes financial attributes of Australia credit card customers. It has 690 instances with 15 attributes consisting of 6 numerical and 9 categorical attributes. The data is also divided into 2 classes, the first class been good loans and second been bad loans. The good loans consist of 307 instances representing 45% whereas the bad consist of 383 instances representing 55%. The dataset is a mixed up of continuous variables, nominal variables, and missing values.

The Germany dataset has 1000 instances with 20 attributes consisting of 8 numerical and 12 categorical attributes. The dataset has no missing values, the good loans are 700 instances representing 70% and the bad loans are 300 instances representing 30%. The remaining attributes range in size from 2 to 5, while one of the nominal attributes has 10 different values.

The Japanese has 125 instances with 10 attributes consisting of 4 numerical and 6 categorical attributes. The good loans consist of 48 instances representing 38% whereas the bad consist

of 77 instances representing 62%. The dataset is a mixed up of continuous variables, nominal variables, and missing values.

The above is represented in a tabular form:

Table 1: Summary of Dataset

DATASET	DESCRIPTION				
	Instances	Attributes	Numerical	Categorical	Variable
Australia	690	15	6	9	GL: 307 BL: 383
Germany	1000	20	8	12	GL: 700 BL: 300
Japan	125	10	4	6	GL: 48 BL: 77

Source: *UCI Machine Learning Repository Dataset*

where GL – Good Loans BL - Bad Loans

3.6 Data Analysis Strategy

The downloaded credit dataset of Australia, Germany, and Japan will be analyzed using R.

The downloaded dataset will be categorized according to the various countries and converted from the data to CSV file to make working with easy. The conversion will be done using Microsoft Excel.

The first step in the analysis is to describe the data set, that is, list the predictor variables that will help classify a credit applicant as likely to default or not default. These variables were captured in the earlier discussion on the dataset. The variables are Age, Job, Housing, Accounts, Credit amount, Duration of credit.

Next is defining the output variable. The expected output variable determines whether an applicant will default or not, with categorical variable for the output classes as Yes (0) and

No (1). Yes, indicates the likelihood of an applicant to default, and No is the likelihood of an applicant not to default.

The dataset is cleaned by removing missing values from the loaded data. Missing values are not significant to this analysis as they produce zero results. After this, each variable is visualized in order to understand the significance of each predictor variable. This is followed by data modelling. This stage begins with a process called Data Splicing, wherein the data set is split into two parts:

- Training set: This part of the data set is used to build and train the Machine Learning model.
- Testing set: This part of the data set is used to evaluate the efficiency of the model.

Distinct variables are generated to store the value of the response variable and loaded the e1071 package, which contains the model function (Naïve Bayes, Support Vector Machine and Gradient Boosting) an in-built function offered by R, to compare the results of the training and testing phases. The training data set is used to generate the Naïve Bayes, Support Vector Machine, and Gradient Boosting models after loading the e1071 package. Evaluating the model is done to obtain the performance metrics. The testing dataset is used to test the effectiveness of the model, and then a confusion matrix is also used to assess how accurate the dataset is.

3.7 Best Machine Learning Technique Determinant

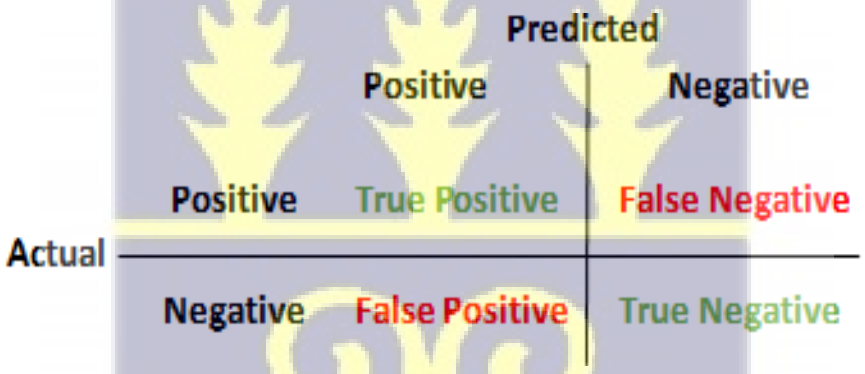
The accuracy result from the data analysis was used to determine which machine learning technique according to this study is best for credit scoring and bankruptcy prediction. This was achieved by running the analysis for each country, that is, Germany, Japan and Australia using the various machine learning techniques, Naïve Bayes, Support Vector Machine and Gradient Boosting.

3.7.1 Model Performance Evaluation

The model performance evaluation is normally generated from a matrix with the numbers of examples correctly and incorrectly classified for each class (Youssef Tounsi, Larbi Hassouni, and Houda Anoun, 2019). The matrix under machine learning is known as confusion matrix.

Confusion matrix: A key idea in classification evaluation is the confusion matrix. The combinations of a dataset's actual and expected observation counts are shown in a matrix. A confusion matrix is a p * p table that lists the predictions made by a classification algorithm given a N number of classes. The axis of the confusion matrix is labeled predicted and actuals. Generally, the confusion matrix is represented in the diagram below:

Figure 1: Confusion Matrix



Source: *Bhandari, 2020*

Predictions lies in one of the outcomes, since it is based on how it matches up with the actual value. True Positive (TP) – predicted true and true in reality, True Negative (TN) – predicted false and false in reality, False Positive (FP) – predicted true and false in reality and False Negative (FN) – predicted false and true in reality.

In evaluating the performance of NB, SVM and GBM, the criteria used are Accuracy, Precision, Recall, and Specificity. D. Boughaci and A. A. K. Alkhawaldeh, 2018, stated that,

one model performance evaluation will not give financial institution, decision makers and others a clear judgment in decision making.

- **Accuracy:** Accuracy measures the correct prediction of the classifier compared to its overall datapoints. It can be calculated as the ratio of the units of correct predictions and the total number of predictions made by the classifier.

$$\text{Accuracy} = \frac{T\ PLTN}{Total(PLN)}.$$

- **Recall:** Recall is also known as Sensitivity or True Discovery. It is the fraction of examples which were predicted to belong to a class with respect to all the examples that truly belong in the class.

$$\text{Recall} = \frac{T\ P}{TPLFN}.$$

If the value obtained after computation is high, it means that the model is better and has correctly identified the positive cases, if it is low, it means the model has identified the positive class in a manner that is not appropriate.

- **Precision:** Precision is defined as the true positive rate and make known of how many actual positives are identified among all the positive predicted by the machine leaning model. When accuracy becomes unreliable by having an imbalance class, precision is used for evaluating the model performance. It is also known as Positive Predictive Value (PPV).

$$\text{Precision} = \frac{T\ P}{TPLFP}.$$

- **Specificity:** It is a measure of how many true negatives that were successfully detected as negative and not true. It is also known as True Negative Rate (TNR)

$$\text{Specificity} = \frac{T\ N}{TNLFP}.$$

The outcome of the analysis is then presented in figures and tables in the next chapter.

CHAPTER FOUR

FINDINGS AND DISCUSSION

4.1 Introduction

This chapter presents and discusses findings based on the objectives of the study. The objectives of this study as captured earlier was to; Determine the most suitable machine learning model for predicting default and identify the strengths and weakness of the models, that is, Naïve Bayes (NB), Support Vector Machine (SVM), and Gradient Boosting (GB).

4.2 Models Evaluation

The analysis is done with respect to the various credit data, following a sequence of Japanese credit data - Naïve Bayes, Support Vector Machine and Gradient Boosting, Germany credit Data - Naïve Bayes, Support Vector Machine and Gradient Boosting, and Australian credit Data - Naïve Bayes, Support Vector Machine and Gradient Boosting.

4.2.2 Japanese Credit Data

4.2.2.1 Naïve Bayes

The analysis of the Japanese dataset represented in the table below:

Table 2: Naïve Bayes Results of the Japanese Data

Statistics	
Accuracy	0.6832
95% CI	(0.5831, 0.7722)
No Information Rate	0.6238
P-Value [Acc > NIR]	0.1287
Kappa	0.4111
Mcnemar's Test P-Value	1.815e-06
Sensitivity	0.5238
Specificity	0.9474
Pos Pred Value	0.9429
Neg Pred Value	0.5455
Prevalence	0.6238
Detection Rate	0.3267
Detection Prevalence	0.3465
Balanced Accuracy	0.7356

Source: *Research Findings (2022).*

The Naïve Bayes model achieved an accuracy of 0.6832, indicating that it correctly predicted the target variable (credit risk) 68.32% of the time. The 95% Confidence Interval (95% CI) for the accuracy estimate was (0.5831, 0.7722), indicating that the true accuracy was expected to lie between 58.31% and 77.22% with 95% probability. The model's kappa value was 0.4111, suggesting moderate agreement between the model's predictions and the actual target variable. The McNemar's test p-value was 1.815e-06, indicating a significant difference between the model's predictions and the actual target variable. The model achieved a sensitivity of 0.5238 and a specificity of 0.9474, correctly identifying 52.38% of the risky credit cases and 94.74% of the safe credit cases. The positive predictive value was 0.9429 and the negative predictive value was 0.5455. The prevalence of the positive class was 0.6238,

and the detection rate was 0.3267. The detection prevalence was 0.3465. The balanced accuracy was 0.7356.

4.2.2.2 Support Vector Machine

The analysis of the credit data presented the result in the table below.

Table 3: Support Vector Machine Results of the Japanese Data

Statistics	
Accuracy	0.5417
95% CI	(0.3282, 0.7445)
No Information Rate	0.625
P-Value [Acc > NIR]	0.8538
Kappa	0
Mcnemar's Test P-Value	1..0000
Sensitivity	0.3333
Specificity	0.6667
Pos Pred Value	0.3750
Neg Pred Value	0.6250
Prevalence	0.3750
Detection Rate	0.1250
Detection Prevalence	0.3333
Balanced Accuracy	0.5000

Source: *Research Findings (2022).*

Based on the results of the SVM model on the Japanese credit data, the accuracy of the model is 0.5417, which means that the model correctly predicted the outcome for 54.17% of the cases. However, it is important to note that the 95% confidence interval (CI) for this estimate is relatively wide, ranging from 0.3282 to 0.7445, indicating that there is a considerable amount of uncertainty associated with the accuracy estimate. The no information rate (NIR) is 0.625, which represents the proportion of the majority class in the dataset. The P-Value [Acc > NIR] is 0.8538, indicating that the model's accuracy is not significantly different from

the NIR. The Kappa statistic measures the agreement between the model's predictions and the actual outcomes, with a value of 1 indicating perfect agreement and 0 indicating agreement due to chance. In this case, the Kappa statistic is 0, which suggests that the model's predictions are no better than random chance. The Sensitivity of the model is 0.3333, which means that the model correctly identified 33.33% of the positive cases. The Specificity is 0.6667, indicating that the model correctly identified 66.67% of the negative cases. The Positive Predictive Value (PPV) of the model is 0.3750, which means that out of all the cases the model predicted as positive, only 37.50% of them were positive.

The Negative Predictive Value (NPV) of the model is 0.6250, indicating that out of all the cases the model predicted as negative, 62.50% of them were negative. The prevalence of the positive class in the dataset is 0.3750, and the Detection Rate of the model is 0.1250, which means that the model correctly identified 12.50% of the positive cases. The Detection Prevalence of the model is 0.3333, indicating that out of all the cases the model predicted as positive, 33.33% of them were positive. Finally, the Balanced Accuracy of the model is 0.5000, which is the average of Sensitivity and Specificity, and represents the accuracy that would be achieved by a random guess.

4.2.2.3 Gradient Boosting

The analysis of the Japanese credit data presented the result:

Table 4: Gradient Boosting Results of the Japanese Data

Statistics	
Accuracy	0.5833
95% CI	(0.3664, 0.7789)
No Information Rate	0.625
P-Value [Acc > NIR]	0.7395
Kappa	0.1489
Mcnemar's Test P-Value	0.7518
Sensitivity	0.5556
Specificity	0.6000
Pos Pred Value	0.4545
Neg Pred Value	0.6923
Prevalence	0.3750
Detection Rate	0.2083
Detection Prevalence	0.4583
Balanced Accuracy	0.5778

Source: *Research Findings (2022).*

The gradient boosting model was analyzed to predict credit outcomes and had an accuracy of 58.33%, indicating that it correctly predicted the outcome of 58.33% of the cases in the test dataset. The 95% confidence interval for the accuracy was found to be between 36.64% and 77.89%, with 95% probability. The model's p-value was greater than 0.05 when compared to the no information rate, suggesting that there was no significant difference between the accuracy of the model and the no information rate. The kappa statistic was found to be 0.1489, indicating low agreement. The sensitivity and specificity of the model were 55.56% and 60.00% respectively, indicating that the model correctly identified 55.56% of the positive cases and 60.00% of the negative cases. The positive predictive value and negative predictive value of the model were found to be 45.45% and 69.23% respectively, indicating that the model correctly predicted 45.45% of the positive cases and 69.23% of the negative cases. The

prevalence of positive cases in the test dataset was 0.3750, and the detection rate and detection prevalence of the model were found to be 0.2083 and 0.4583 respectively. The balanced accuracy of the model was found to be 0.5778, indicating that the model was performing slightly better than random guessing. The 'Positive' class considered by the model was class 0.

4.2.3 German Credit Data

4.2.3.1 Naïve Bayes

The analysis of the German credit data presented the results in a table below:

Table 5: Naïve Bayes Results of the Germany Data

Statistics	
Accuracy	0.645
95% CI	(0.5744, 0.7112)
No Information Rate	0.7
P-Value [Acc > NIR]	0.960372
Kappa	0.2447
Mcnemar's Test P-Value	0.004396
Sensitivity	0.6167
Specificity	0.6571
Pos Pred Value	0.4353
Neg Pred Value	0.8000
Prevalence	0.3000
Detection Rate	0.1850
Detection Prevalence	0.4250
Balanced Accuracy	0.6369

Source: *Research Findings (2022).*

The results show that the accuracy of the model was 0.645, which means that it correctly predicted the outcome of 64.5% of the cases in the dataset. The 95% confidence interval for the accuracy was (0.5744, 0.7112), indicating that the true accuracy of the model lies between

these two values with 95% confidence. The model's accuracy was only slightly better than the no information rate (NIR) of 0.7, which is the accuracy rate of the model that always predicts the most common outcome in the dataset. The p-value (0.960372) for the model's accuracy being greater than the NIR was not significant, indicating that the model's accuracy was not significantly different from the accuracy of the simplest model that always predicts the most common outcome. The Kappa value of the model was 0.2447, indicating a fair agreement between the predicted outcomes and the actual outcomes in the dataset. The sensitivity of the model was 0.6167, meaning that it correctly identified 61.67% of the positive cases in the dataset. The specificity of the model was 0.6571, meaning that it correctly identified 65.71% of the negative cases in the dataset. The positive predictive value (PPV) of the model was 0.4353, meaning that when the model predicts a positive outcome, it is correct 43.53% of the time. The negative predictive value (NPV) of the model was 0.8, meaning that when the model predicts a negative outcome, it is correct 80% of the time. The prevalence of positive cases in the dataset was 0.3, and the detection rate of the model was 0.1850, meaning that the model correctly identified 18.50% of the positive cases in the dataset. The detection prevalence of the model was 0.4250, meaning that the model predicted a positive outcome for 42.50% of the cases in the dataset. The balanced accuracy of the model was 0.6369, which considers both sensitivity and specificity, indicating a moderately good performance of the model.

4.2.3.2 Support Vector Machine

The analysis of the German credit data presented the result below:

Table 6: Support Vector Machine Results of the Germany Data

Statistics	
Accuracy	0.71
95% CI	(0.6418, 0.7718)
No Information Rate	0.7
P-Value [Acc > NIR]	0.4123
Kappa	0.2677
Mcnemar's Test P-Value	0.1486
Sensitivity	0.4167
Specificity	0.8357
Pos Pred Value	0.5208
Neg Pred Value	0.7697
Prevalence	0.3000
Detection Rate	0.1250
Detection Prevalence	0.2400
Balanced Accuracy	0.6262

Source: *Research Findings (2022).*

The SVM model had an overall accuracy of 71%, indicating that it correctly predicted the outcome of 71% of the cases in the test dataset. The 95% confidence interval for the accuracy was found to be between 64.18% and 77.18%, with 95% probability. When compared to the no information rate, which is the accuracy of a model that always predicts the majority class in the test dataset, the SVM model's p-value was found to be greater than 0.05, indicating that there was no significant difference between the accuracy of the model and the no information rate. The kappa statistic, was found to be 0.2677, indicating low agreement. The sensitivity and specificity of the model were found to be 41.67% and 83.57% respectively, indicating that the model correctly identified 41.67% of the positive cases and 83.57% of the negative cases.

The positive predictive value and negative predictive value of the model were found to be 52.08% and 76.97% respectively, indicating that the model correctly predicted 52.08% of the positive cases and 76.97% of the negative cases. The prevalence of positive cases in the test dataset was 0.3, and the detection rate and detection prevalence of the model were found to be 0.1250 and 0.2400 respectively. The balanced accuracy of the model, which is the average of sensitivity and specificity, was found to be 0.6262, indicating that the model was performing slightly better than random guessing. The 'Positive' class considered by the model was class 0.

4.2.3.3 Gradient Boosting

The analysis of the German credit data presented the result:

Table 7: Gradient Boosting Results of the Germany Data

Statistics	
Accuracy	0.75
95% CI	(0.684, 0.8084)
No Information Rate	0.7
P-Value [Acc > NIR]	0.06955
Kappa	0.3557
Mcnemar's Test P-Value	0.03389
Sensitivity	0.4500
Specificity	0.8786
Pos Pred Value	0.6136
Neg Pred Value	0.7885
Prevalence	0.3000
Detection Rate	0.1350
Detection Prevalence	0.2200
Balanced Accuracy	0.6643

Source: *Research Findings (2022).*

The gradient boosting model for German credit data was analyzed and had an accuracy of 75%, indicating that it correctly predicted the outcome of 75% of the cases in the test dataset. The 95% confidence interval for the accuracy was found to be between 68.4% and 80.84%, with 95% probability. The model's p-value was less than 0.05 when compared to the no information rate, suggesting that there was a significant difference between the accuracy of the model and the no information rate. The kappa statistic was found to be 0.3557, indicating moderate agreement. The sensitivity and specificity of the model were 45.00% and 87.86%, respectively, indicating that the model correctly identified 45.00% of the positive cases and 87.86% of the negative cases. The positive predictive value and negative predictive value of the model were found to be 61.36% and 78.85%, respectively, indicating that the model correctly predicted 61.36% of the positive cases and 78.85% of the negative cases. The prevalence of positive cases in the test dataset was 0.3, and the detection rate and detection prevalence of the model were found to be 0.1350 and 0.2200, respectively. The balanced accuracy of the model was found to be 0.6643, indicating that the model was performing better than random guessing. The 'Positive' class considered by the model was class 0.

4.2.4 Australian Credit Data

4.2.4.1 Naïve Bayes

The analysis of the Australian credit data presented the result:

Table 8: Naïve Bayes Results of the Australian Data

Statistics	
Accuracy	0.7664
95% CI	(0.6866, 0.8344)
No Information Rate	0.5547
P-Value [Acc > NIR]	2.139e−07
Kappa	0.5114
Mcnemar's Test P-Value	0.0007829
Sensitivity	0.5738
Specificity	0.9211
Pos Pred Value	0.8537
Neg Pred Value	0.7292
Prevalence	0.4453
Detection Rate	0.2555
Detection Prevalence	0.2993
Balanced Accuracy	0.7474

Source: *Research Findings (2022).*

The Naïve Bayes machine learning model achieved an accuracy of 0.7664 or 76.64%, which means that the model correctly classified 76.64% of the instances in the dataset. The 95% confidence interval for the accuracy was between 0.6866 and 0.8344. The model performed better than the No Information Rate, which was 0.5547. The p-value associated with the difference between the model's accuracy and the No Information Rate was very small, indicating that the difference was statistically significant. The Kappa score was 0.5114, indicating a moderate level of agreement between the model's predictions and the actual labels. The model's sensitivity was 0.5738, and specificity was 0.9211, indicating that the model was better at identifying low-risk borrowers than high-risk borrowers. The positive predictive value was 0.8537, indicating that when the model predicted a borrower as high-risk, it was correct 85.37% of the time. The negative predictive value was 0.7292, indicating

that when the model predicted a borrower as low risk, it was correct 72.92% of the time. The prevalence was 0.4453 or 44.53%, and the detection rate was 0.2555, indicating that the model identified 25.55% of high-risk borrowers correctly. The detection prevalence was 0.2993, indicating that the model predicted 29.93% of borrowers as high-risk. The balanced accuracy was 0.7474, indicating that the model has good discrimination power in distinguishing between high-risk and low-risk borrowers.

4.2.4.2 Support Vector Machine

The analysis of the Australian credit data presented the result:

Table 9: Support Vector Machine Results of the Australian Data

Statistics	
Accuracy	0.4672
95% CI	(0.3815, 0.5543)
No Information Rate	0.5547
P-Value [Acc > NIR]	0.9839
Kappa	- 0.0115
Mcnemar's Test P-Value	2.846e-06
Sensitivity	0.2500
Specificity	0.7377
Pos Pred Value	0.5429
Neg Pred Value	0.4412
Prevalence	0.5547
Detection Rate	0.1387
Detection Prevalence	0.2555
Balanced Accuracy	0.4939

Source: *Research Findings (2022).*

The results of the SVM machine learning model for the credit data show an accuracy of 0.4672, which means that the model correctly predicted the target variable (credit risk) for 46.72% of the instances in the dataset. The 95% confidence interval (CI) for this estimate is

(0.3815, 0.5543), which means that if the model was applied to different samples of data, the accuracy would be expected to fall within this range 95% of the time. The "No Information Rate" (NIR) is the accuracy that would be achieved if the model simply predicted the most common class for all instances. In this case, the NIR is 0.5547, which means that the model is performing only slightly better than chance. The p-value [Acc > NIR] is 0.9839, which indicates that the model's accuracy is not statistically significantly different from the NIR. The kappa coefficient is -0.0115, which suggests poor agreement between the model's predictions and the true values. The sensitivity and specificity measures indicate how well the model is able to correctly identify positive and negative instances, respectively. The model has a sensitivity of 0.25, which means that it correctly identifies 25% of the positive instances, and a specificity of 0.7377, which means that it correctly identifies 73.77% of the negative instances.

The positive predictive value (PPV) is the proportion of instances predicted as positive that are positive, and the negative predictive value (NPV) is the proportion of instances predicted as negative that are actually negative. In this case, the PPV is 0.5429 and the NPV is 0.4412. The prevalence is the proportion of instances that are positive in the dataset, which in this case is 0.5547. The detection rate is the proportion of positive instances that are correctly identified by the model, which in this case is 0.1387. The detection prevalence is the proportion of instances predicted as positive by the model, which in this case is 0.2555. Finally, the balanced accuracy is a measure of the overall accuracy of the model that considers the proportions of positive and negative instances in the dataset. In this case, the balanced accuracy is 0.4939.

4.2.4.3 Gradient Boosting

The analysis of the Australian credit data presented the result:

Table 10: Gradient Boosting Results of the Australian Data

Statistics	
Accuracy	0.8686
95% CI	(0.8003, 0.9202)
No Information Rate	0.5547
P-Value [Acc > NIR]	2.871e-15
Kappa	0.7349
Mcnemar's Test P-Value	0.8137
Sensitivity	0.8684
Specificity	0.8689
Pos Pred Value	0.8919
Neg Pred Value	0.8413
Prevalence	0.5547
Detection Rate	0.4818
Detection Prevalence	0.5401
Balanced Accuracy	0.8686

Source: *Research Findings (2022).*

The gradient boosting model was analyzed to predict credit outcomes of the Australian dataset and had an accuracy of 86.86%, indicating that it correctly predicted the outcome of 86.86% of the cases in the test dataset. The 95% confidence interval for the accuracy was found to be between 80.03% and 92.02%, with 95% probability. When compared to the no information rate, which is the accuracy of a model that always predicts the majority class in the test dataset, the model's p-value was found to be 2.871e-15, indicating a significant difference between the accuracy of the model and the no information rate. The kappa statistic was found to be 0.7349, indicating substantial agreement. The sensitivity and specificity of the model were 86.84% and 86.89% respectively, indicating that the model correctly identified 86.84% of the positive cases and 86.89% of the negative cases. The positive predictive value and negative predictive value of the model were found to be 89.19% and

84.13% respectively, indicating that the model correctly predicted 89.19% of the positive cases and 84.13% of the negative cases. The prevalence of positive cases in the test dataset was 0.5547, and the detection rate and detection prevalence of the model were found to be 0.4818 and 0.5401 respectively. The balanced accuracy of the model, which is the average of sensitivity and specificity, was found to be 0.8686, indicating that the model was performing well. The 'Positive' class considered by the model was class 0.

4.3 Summary of the Analysis

Table 11: Summary of the Results According to the Evaluating Metrics

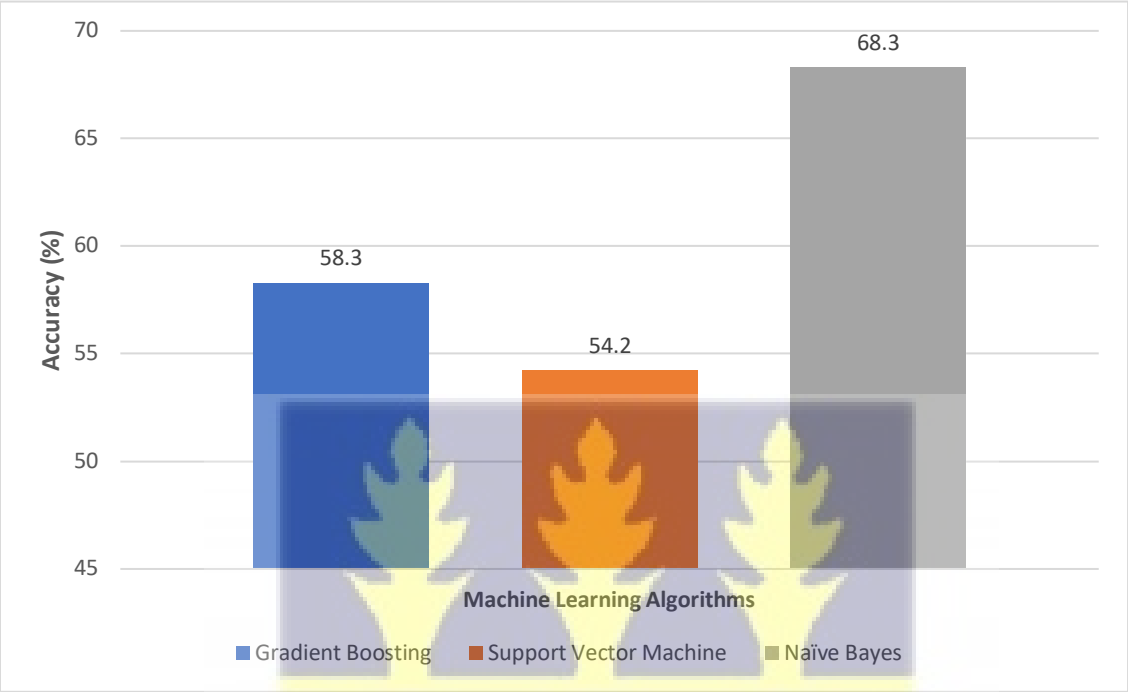
TECHNIQUES USED	CREDIT DATA	EVALUATION METRICS				
		Accuracy	Recall / Sensitivity	Specificity	Precision / Positive Predictive Value	Negative Predictive Value
NAÏVE BAYES	Japanese	0.6832	0.5238	0.9474	0.9429	0.5455
	German	0.645	0.6167	0.6571	0.4353	0.8000
	Australian	0.7664	0.5738	0.9211	0.8537	0.7292
SUPPORT VECTOR MACHINE	Japanese	0.5417	0.3333	0.6667	0.3750	0.6250
	German	0.71	0.4167	0.8357	0.5208	0.7697
	Australian	0.4672	0.2500	0.7377	0.5429	0.4412
GRADIENT BOOSTING	Japanese	0.5833	0.5556	0.6000	0.4545	0.6923
	German	0.75	0.4500	0.8786	0.6136	0.7885
	Australian	0.8686	0.8684	0.8689	0.8919	0.8413

Source: Author, 2022.

4.4 Accuracy Prediction of the Models

4.4.1 Accuracy Prediction of Japanese Dataset

Figure 2: Comparison of machine learning algorithm accuracy on the Japanese credit dataset



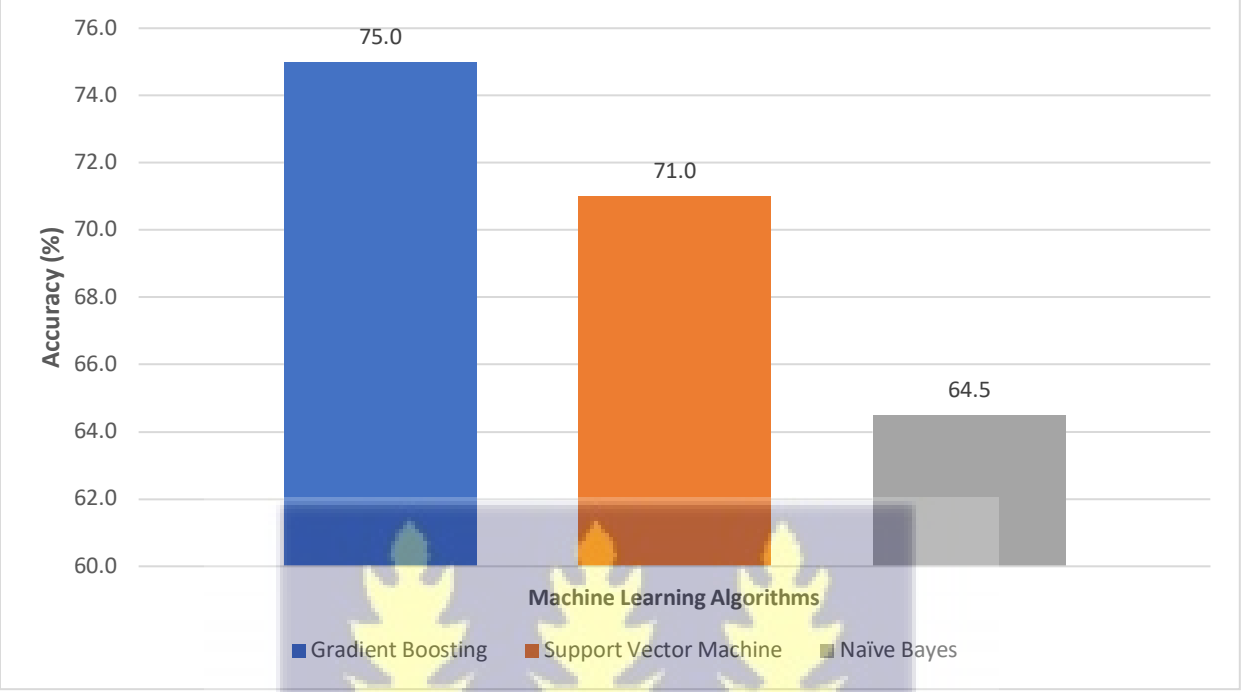
Source: *Author, 2022.*

Figure 2 shows the accuracy values of three machine learning algorithms - Naïve Bayes, SVM, and Gradient Boosting - on the Japanese credit dataset. The y-axis represents the accuracy values, and the x-axis represents the algorithms.

According to the chart, the SVM algorithm had the lowest accuracy value, with an accuracy of 54.2%. The Gradient Boosting algorithm had an accuracy value of 58.3%. The Naïve Bayes algorithm had the highest accuracy value, with an accuracy of 68.3%. Overall, these results suggest that the Naïve Bayes algorithm performs the best on this dataset, followed by the Gradient Boosting algorithm and then the SVM algorithm.

4.4.2 Accuracy Prediction of German Dataset

Figure 3: Comparison of machine learning algorithm accuracy on the German credit dataset



Source: *Author, 2022.*

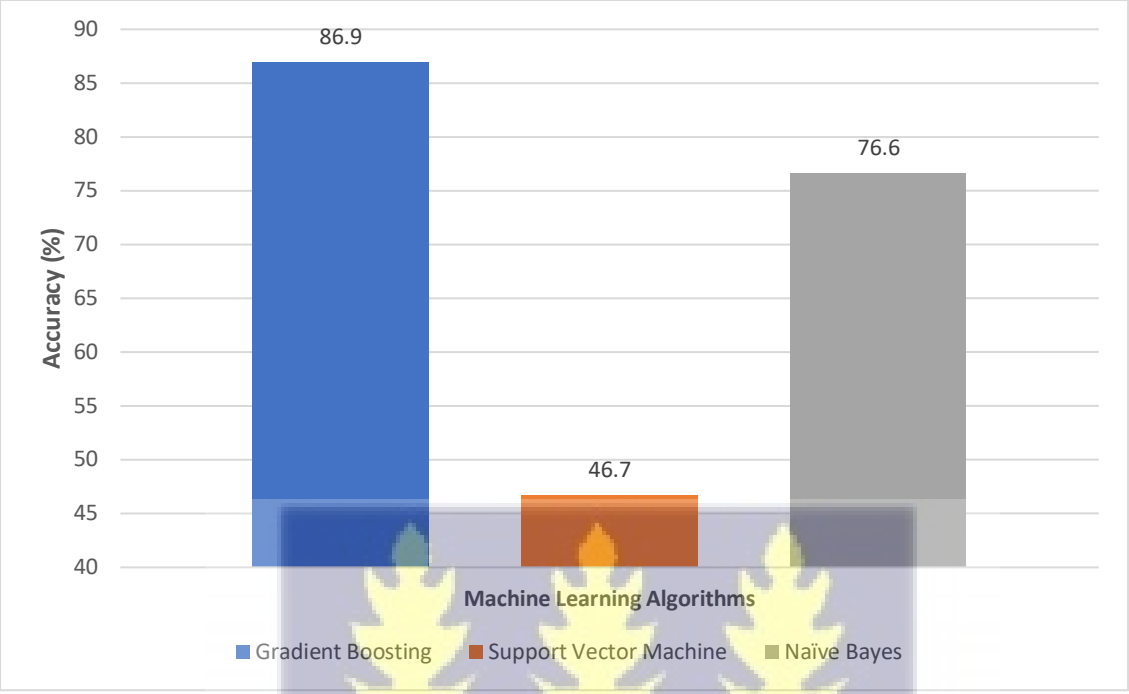
Figure 3 shows the accuracy values of three different machine learning algorithms applied to the German credit dataset. The Naïve Bayes model achieved an accuracy of 64.5%, the Support Vector Machine model achieved an accuracy of 71.0%, and the Gradient Boosting model achieved an accuracy of 75.0%.

Overall, the Gradient Boosting model had the highest accuracy value, followed by the SVM model and then the Naïve Bayes model. This suggests that the gradient boosting model was the most effective at correctly predicting the class labels of the German credit data.

It is worth noting that all three algorithms performed relatively well on this dataset, with accuracy values all above 60%. However, the Gradient Boosting model had a higher accuracy value, which may make it the best choice for this application.

4.4.3 Accuracy Prediction of Australian Dataset

Figure 4: Comparison of machine learning algorithm accuracy on the Australian credit dataset



Source: *Author, 2022.*

According to Figure 4, the SVM algorithm had the lowest accuracy value, with an accuracy of 46.7%. The NB algorithm had the second highest accuracy value, with an accuracy of 76.6%. The GB algorithm had the highest accuracy value, with an accuracy of 86.9%. Overall, these results suggest that the Gradient Boosting algorithm performs the best on this dataset, followed by the Naïve Bayes algorithm and then the Support Vector Machine algorithm.

4.4.4 Accuracy Prediction Plot of the Models

Figure 5: Line Graph of the Accuracy on the three-credit dataset

Source: *Author, 2022.*

Figure 5 shows the percentage accuracy results of three machine learning algorithms, namely Naïve Bayes, Support Vector Machine, and Gradient Boosting, applied to three different credit datasets from Japanese, Germany, and Australia. The chart indicates that on average, Gradient Boosting has the highest accuracy in predicting default compared to the other two algorithms across all three datasets. However, Naïve Bayes, on average, tends to have a higher prediction accuracy than the Support Vector Machine algorithm.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

This chapter summarizes the main findings, draws conclusions from the findings, and makes recommendations as well as suggestions for future research.

5.2 Summary

This study provides an overview of the best machine learning technique for credit scoring and bankruptcy prediction in banking and finance to reduce the risk of giving out loans to likely defaulters. The following is a summary of key findings of the study:

According to the study, the Gradient Boosting model outperformed the Naïve Bayes and Support Vector Machine when applied to the German, Japanese and Australian dataset. The average accuracy prediction of the models is Gradient Boosting (73.4%), Naïve Bayes (69.8%), and Support Vector Machine (57.3%).

Based on the accuracy prediction of the various models, I found out that Support Vector Machine and Naïve Bayes models do not perform well with large data as compared to the Gradient Boosting model.

5.3 Conclusion

In conclusion, the study aimed to access the accuracy of Naïve Bayes, Support Vector Machine, and Gradient Boosting machine learning algorithms determine the using the Japanese, German, and Australian credit datasets. The primary goal was to compare their performance in credit scoring and bankruptcy prediction. The three algorithms, Naïve Bayes, Support Vector Machine, and Gradient Boosting were applied to the respective datasets. The results obtained shows that, the Gradient Boosting algorithm had the highest accuracy values

across all the datasets, followed by the Naïve Bayes and then the Support Vector Machine algorithms. However, the Naïve Bayes algorithm had higher prediction accuracy than the Gradient Boosting algorithm when applied to the Japanese credit dataset which had a lower number of instances. This suggests that Naïve Bayes may be a preferable choice for datasets with fewer instances.

Overall, the Gradient Boosting algorithm can be considered the best choice for credit scoring and bankruptcy prediction in banking and finance.

5.4 Recommendation

5.4.1 Financial Institutions

Based on the results of the analysis, the recommendation is to use Gradient Boosting model for classification tasks on credit data to financial institutions. The Gradient Boosting model achieved an average accuracy of 73.4% on the dataset, which was significantly higher than the average accuracy values of the Naïve Bayes, and Support Vector Machine models.

While all three models performed well, the Gradient Boosting model consistently outperformed the other models in terms of accuracy. Therefore, recommending the use of Gradient Boosting model as it provides the best balance of accuracy and computational efficiency for this type of dataset.

5.4.2 Recommendation for Future Research

Future research should consider applying reinforcement learning, which essentially relies on Markov decision-making, for credit rating and bankruptcy predictions. This would give decision-makers an additional overview of different machine learning techniques that can help detect defaulters, which might or would be helpful than the supervised learning techniques.

REFERENCE

Alpaydin, E. (2004). Introduction to machine learning. The MIT Press, Cambridge, MA., USA, 2nd edn. 2

Altman, E. I. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The Journal of Finance, 23*(4), 589–609.

Anderson, R. (2007). The Credit Scoring Toolkit: Theory and Practice for Re- tail Credit Risk Management and Decision Automation. Oxford University Press, USA. 51, 52, 53, 58, 71, 74, 77, 84, 88, 89, 96, 202

Atiya, A. F. (2005). Bankruptcy Prediction for Credit Risk Using Neural Networks: A Survey and New Results. IEEE Transactions on Neural Networks 12 (4): 929-935.

Baesens, B., Van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J. (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society*, *54*(6), 627–635. https://doi.org/10.1057/palgrave.jors.2601545

Barboza, Kimura and Altman, (2017). "Machine Learning Models and Bankruptcy Prediction". Expert Systems with Applications 83, 405–417.

Basel Committee on Banking Supervision *International Convergence of Capital Measurement and Capital Standards. A Revised Framework*. Bank for International Settlements, June 2004.

Bissacco, A., Yang, M. H., & Soatto, S. (2007). Fast Human Pose Estimation using Appearance and Motion via Multi-Dimensional Boosting Regression. *2007 IEEE Conference on Computer Vision and Pattern Recognition*. https://doi.org/10.1109/cvpr.2007.383129

Brown, I. & Mues, C. (2012). An experimental comparison of classification algorithms for imbalanced credit scoring data sets. Expert Systems with Applications, 39, 3446–3453. 48, 105, 158

Cakmak, G. and Yildiz, C., The prediction of seedy grape drying rate using a neural network method. Computers and Electronics in Agriculture, 75:1(2011), 132-138.

Cook, D. & Holder, L. (2001). A client-server interactive tool for integrated artificial intelligence curriculum. In Proceedings of the Fourteenth International Florida Artificial Intelligence Research Society Conference AAAI Press. 2

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. https://doi.org/10.1007/bf00994018

Cristianini, N. & Shawe-Taylor, J. (2001). An introduction to support vector machines and other kernel-based learning methods. Repr. Introduction to Support Vector Machines and other Kernel-Based Learning Methods. 22. 10.1017/CBO9780511801389.

Demˇsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. The Journal of Machine Learning Research, 7, 1–30. 48, 49, 128

Dietterich, T. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. Neural computation, 10, 1895–1923. 40

Durand, J. (1941). Models and Statistical Inference: The Controversy between Fisher and Neyman-Pearson, Oxford University Press.

Durand, P. (1941). Credit Scoring Using Neural and Evolutionary Techniques. IMA Journal of Management Mathematics, 11(2), pp 111–125.

Friedman, M. (1937). The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance. *Journal of the American Statistical Association*, *32*(200), 675–701. https://doi.org/10.1080/01621459.1937.10503522

Garc´ıa, S., Fern´andez, A., Luengo, J. & Herrera, F. (2010). Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. Information Sciences, 180, 2044–2064. 49

García, S. & Herrera. F. (2008). An Extension on "Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons. *Journal of Machine Learning Research*, *9*(89), 2677–2694. https://jmlr.csail.mit.edu/papers/volume9/garcia08a/garcia08a.pdf

Gestel, T. van, Baesens, B., & Van Gestel, T. (2009). *Credit Risk Management: Basic Concepts: Financial Risk Components, Rating Analysis, Models, Economic and Regulatory Capital*. Oxford University Press.

Hand, D. (2001). Modelling consumer credit risk. IMA Journal of Management Mathematics, 12, 139–155. 53, 55, 56, 177

Hodges, J. & Lehmann, E. (1962). Rank methods for combination of independent experiments in analysis of variance. The Annals of Mathematical Statistics, 33, 482–497. 49

Hutchinson, J. R., Bates, K. T., Molnar, J., Allen, V., & Makovicky, P. J. (2011). A Computational Analysis of Limb and Body Dimensions in Tyrannosaurus rex with Implications for Locomotion, Ontogeny, and Growth. *PLoS ONE*, *6*(10), e26037. https://doi.org/10.1371/journal.pone.0026037

Japkowicz, N. & Shah, M. (2011). Evaluating Learning Algorithms: A Classification Perspective. Cambridge University Press, 1st edn.

Jiang, J., Trundle, P. and Ren, J., Medical image analysis with artificial neural networks. Computerized Medical Imaging and Graphics, 34:8(2010), 617-631

Kennedy, K. (2013). Credit scoring using machine learning. Doctoral thesis. Technological University Dublin. doi:10.21427/D7NC7J.

Kim, J. (2009). Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap. Computational Statistics & Data Analysis, 53, 3735–3745. 47

Kruskal, W. & Wallis, W. (1952). Use of ranks in one-criterion variance analysis. Journal of the American Statistical Association, 583–621. 49

Loretta J. Mester, (1997). "What's the point of credit scoring?," Business Review, Federal Reserve Bank of Philadelphia, issue Sep, pages 3-16.

M. B. Waad, "On Feature Selection Methods for Credit Scoring," no. January 2015.

McCarthy, J., Minsky, M. L., Rochester, N., and Shannon, C. E. (1995). A proposal for the Dartmouth Summer Research Project on Artificial Intelligence.

McCarthy, J., Minsky, M., Rochester, N. & Shannon, C. (1955). A proposal for the Dartmouth summer research project on artificial intelligence. Tech. rep., Dartmouth College, http://www.aaai.org/ojs/index.php/aimagazine/article/viewArticle/1904last accessed 29/01/2013. 2

Mercer. J. (1909). Functions of positive and negative type, and their connection with the theory of integral equations. *Proceedings of the Royal Society of London. Series a, Containing Papers of a Mathematical and Physical Character*, *83*(559), 69–70. https://doi.org/10.1098/rspa.1909.0075

Mira, J. (2008). Symbols versus connections: 50 years of artificial intelligence. Neurocomputing, 71, 671–680. 2

Mitchell, T. M. (1997). *Machine Learning (McGraw-Hill International Editions Computer Science Series)* (1st ed.). McGraw-Hill.

Pittman, S. J., & Brown, K. A. (2011). Multi-Scale Approach for Predicting Fish Species Distributions across Coral Reef Seascapes. *PLoS ONE*, *6*(5), e20583. https://doi.org/10.1371/journal.pone.0020583

Rie Johnson, & Tong Zhang. (2013). Accelerating Stochastic Gradient Descent using Predictive Variance Reduction. *Neural Information Processing Systems*, *26*, 315–323. http://riejohnson.com/rie/stograd_nips.pdf

Tavasoli, S. (2022, November 14). *Top 10 Machine Learning Algorithms for Beginners: Supervised, Unsupervised Learning and More*. Simplilearn.com. https://www.simplilearn.com/10-algorithms-machine-learning-engineers-need-to-know-article

Thomas, L. (2009). Consumer credit models: Pricing, profit, and portfolios. Oxford

Thomas, L. (2009). Operations research in consumer finance: Challenges for operational research. Journal of Operational Research Society, 61, 41–52. 5, 52, 54, 103, 124, 127, 158, 238

Thomas, L.C., Edelman, D.B. & Crook, J.N. (2002). Credit scoring and its applications. Society for Industrial and Applied Mathematics, Philadelphia, USA. 24, 53, 85, 86, 126, 248

Tian, Y., Shi, Y., & Liu, X. (2012). Recent advances on support vector machines research. *Technological and Economic Development of Economy*, *18*(1), 5–33.

Tsai, C.-F., Hsu, Y.-F., & Yen, D. C. (2014). A comparative study of classifier ensembles for bankruptcy prediction. *Applied Soft Computing*, *24*, 977–984. University Press, USA.

V. Desay, J. N. Crook and G. A. Overstreet (1996). A Comparison of neural network and linear scoring models in the credit union environment, European Journal of operational Research 95, 24 – 37.

Van Gestel, T. & Baesens, B. (2009). Credit Risk Management: Basic Concepts. Oxford University Press, USA. xxi, 5, 27, 52, 60, 67, 68, 104

Velioğlu, H. M., Boyacı, I. H. and Kurultay, S., Determination of visual quality of tomato paste using computerized inspection system and artificial neural networks. Computers and Electronics in Agriculture, 77:2 (2011), 147-154

Vidal, M. F., & Barbon, F. (2019). *Credit Scoring in Financial Inclusion, Technical Guide*. Retrieved from Washington DC. Technical Guide Credit Score.

West, D. (2000). Neural network credit scoring models. Computers & Operations Research, 27, 1131–1152. 5, 7, 105, 113, 126

Wilcoxon, F. (1945). Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, *1*(6), 80. https://doi.org/10.2307/3001968

Yu, L., Wang, S. and Lai, K. K., Forecasting crude oil price with an EMD-based neural network ensemble learning paradigm. Energy Economics, 30:5, (2008), 2623-2635

APPENDICES

CODES USED FOR THIS STUDY

Naïve Bayes

```
library(caret), library(naivebayes), library(caTools), library(MASS), library(psych),
library(e1071), library(tidyverse), library(lattice), library(dplyr), library(class), library(tidyr),
```

Japanese Data

```
#Load Data
library(readxl)
Japancrx <- read_excel("C:/Users/F/Desktop/Evans School/Thesis/Data to be
used/Japancrx.xlsx")
#Statistics of the data
head(Japancrx)
str(Japancrx)
summary(Japancrx)

#Convert to factor
Japancrx$`Credit Rating`<-as.factor(Japancrx$`Credit Rating`)
str(Japancrx)

#Partition of data - train(80%) & test(20%)
set.seed(1)
indexj<-createDataPartition(Japancrx$`Credit Rating`,p=0.8, list=FALSE )
trainj<-Japancrx[index1,]
validj<-Japancrx[-index1,]

#Cross validation Technique
cvt<-trainControl(method ="cv", number = 10 )
set.seed(1)

#Fit the Model
names(Japancrx)
nb_fit<- naiveBayes(`Credit Rating` ~ Job+Age+Gender+Address+Purpose+
```

```
        Duration+Risk+Balance+income,data=train1,trcontrol= cvt )
nb_fit
summary(nb_fit)

#Confusion Matrix for the Japanese Data
nb_pred<-predict(nb_fit, newdata=valid1)
nb_pred
confusionMatrix(nb_pred, valid1$`Credit Rating`, positive="+")
```

Australian Data

```
#Load Data
library(readxl)
Australiancrx <- read_excel("C:/Users/F/Desktop/Evans School/Thesis/Data to be
used/Australiancrx.xlsx")

#Statistics of the data
head(Australiancrx)
str(Australiancrx)
summary(Australiancrx)

#Convert to factor
Australiancrx$`Pos/Neg` <-as.factor(Australiancrx$`Pos/Neg`)

#Partition of data - train(80%) & test(20%)
set.seed(1)
indexa<-createDataPartition(Australiancrx$`Pos/Neg`,p=0.8, list=FALSE )
traina<-Australiancrx[indexa,]
valida<-Australiancrx[-indexa,]

#Cross validation Technique
cva<-trainControl(method ="cv", number = 10 )
set.seed(1)

#Fit the Model
```

```
names(Australiancrx)
nba_fit<- naiveBayes(`Pos/Neg` ~ . , data=traina,trcontrol= cva )
nba_fit
summary(nba_fit)

#Confusion Matrix for the Australian Data
nba_class<-predict(nba_fit, newdata=valida)
nba_class
confusionMatrix(nba_class, valida$`Pos/Neg`)
```

Germany Data

```
#Load Data
library(readxl)
germancrx <- read_excel("C:/Users/F/Desktop/Evans School/Thesis/Data to be
used/germancrx.xlsx")

#Statistics of the data
head(germancrx)
str(germancrx)
summary(germancrx)

#Convert to factor
germancrx$Creditability <-as.factor(germancrx$Creditability)

#Partition of data - train(80%) & test(20%)
set.seed(1)
indexg<-createDataPartition(germancrx$Creditability,p=0.8, list=FALSE )
traing<-germancrx[indexg,]
validg<-germancrx[-indexg,]

#Cross validation Technique
cv<-trainControl(method ="cv", number = 10 )
set.seed(1)
#Fit the Model
```

```
names(germancrx)
nbg_fit<- naiveBayes(Creditability ~ . ,data=traing,trcontrol= cv )
nbg_fit
summary(nbg_fit)

#Confusion Matrix for the German Data
nbg_class<-predict(nbg_fit, newdata=validg)
nbg_class
confusionMatrix(nbg_class, validg$Creditability)
```

Support Vector Machine

```
library(caret), library(svm), library(caTools), library(MASS), library(psych) , library(e1071),
library(tidyverse), library(lattice), library(dplyr), library(class), library(tidyr),
library(svmpath),library(SVMMaj)
```

Japanese Data

```
#Load Data
library(readxl)
Japancrx <- read_excel("C:/Users/F/Desktop/Evans School/Thesis/Data to be
used/Japancrx.xlsx")
#Statistics of the data
head(Japancrx)
str(Japancrx)
summary(Japancrx)

#Convert to factor
Japancrx$`Credit Rating`<-as.factor(Japancrx$`Credit Rating`)
str(Japancrx)

#Partition of data - train(80%) & test(20%)
set.seed(1)
indjj<-createDataPartition(Japancrx$`Credit Rating`,p=0.8, list=FALSE )
trainjj<-Japancrx[indjj,]
validjj<-Japancrx[-indjj,]
```

```
#Fit the Model
set.seed(1)
names(Japancrx)
svm_fit<- svm(`Credit Rating` ~ Job+Age+Gender+Address+Purpose+
Duration+Risk+Balance+income,data=trainjj,kernel="linear",cost=10 )
print(svm_fit)
summary(svm_fit)

#Confusion Matrix for the Japanese Data
svm_pred<-predict(svm_fit, newdata=validjj)
svm_pred
confusionMatrix(svm_pred, validjj$`Credit Rating`)
```

Australian Data

```
#Load Data
library(readxl)
Australiancrx<- read_excel("C:/Users/F/Desktop/Evans School/Thesis/Data to be
used/Australiancrx.xlsx")

#Statistics of the data
head(Australiancrx)
str(Australiancrx)
summary(Australiancrx)

#Convert to factor
Australiancrx$`Pos/Neg`<-as.factor(Australiancrx$`Pos/Neg`)
Australiancrx$age<-as.factor(Australiancrx$age)
str(Australiancrx)

#Partition of data - train(80%) & test(20%)
set.seed(1)
indaa<-createDataPartition(Australiancrx$`Pos/Neg`,p=0.8, list=FALSE )
trainaa<-Australiancrx[indaa,]
validaa<-Australiancrx[-indaa,]
```

```
#Fit the Model
set.seed(1)
names(Australiancrx)
svma_fit<svm(`Pos/Neg`~age+credit_screening+problematic_region+discredit_bad_region+
unmarried+married+jobless_mascu+jobless_fem+purchase_item+numb_of_years_in_compa
ny+numb_of_months+monthly_payment,deposit in bank,data=traina,
kernel="linear",cost=10)
print(svma_fit)
summary(svma_fit)

#Confusion Matrix for the Australian Data
svma_pred<-predict(svma_fit, newdata=validaa)
svma_pred
confusionMatrix(svma_pred, validaa$`Pos/Neg`)
```

Germany Data

```
#Load Data
library(readxl)
germancrx<- read_excel("C:/Users/F/Desktop/Evans School/Thesis/Data to be
used/germancrx.xlsx")

#Statistics of the data
head(germancrx)
str(germancrx)
summary(germancrx)

#Convert to factor
germancrx$Creditability<-as.factor(germancrx$Creditability)
str(germancrx)

#Partition of data - train(80%) & test(20%)
set.seed(1)
indgg<-createDataPartition(germancrx$Creditability,p=0.8, list=FALSE )
traingg<-germancrx[indgg,]
validgg<-germancrx[-indgg,]
```

```
#Fit the Model
set.seed(1)
names(germancrx)
svmg_fit<- svm(Creditability~.,data=traing,kernel="linear",cost=10 )

print(svmg_fit)
summary(svmg_fit)

#Confusion Matrix for the German Data
svmg_pred<-predict(svmg_fit, newdata=validgg)
svmg_pred
confusionMatrix(svmg_pred, validgg$Creditability)
```

Gradient Boosting

```
library(caret), library(xgboost), library(caTools), library(MASS), library(psych) ,
library(e1071), library(tidyverse), library(lattice), library(dplyr), library(class), library(tidyr),
library(readxl)
```

Japanese Data

```
#Load Data
library(readxl)
Japancrx<- read_excel("C:/Users/F/Desktop/Evans School/Thesis/Data to be
used/Japancrx.xlsx")

#Statistics of the data
head(Japancrx)
glimpse(Japancrx)
str(Japancrx)
summary(Japancrx)

#Convert to factor
Japancrx$`Credit Rating`<-as.factor(Japancrx$`Credit Rating`)
Japancrx$Address<-as.factor(Japancrx$Address)
Japancrx$Job<-as.factor(Japancrx$Job)
Japancrx$Gender<-as.factor(Japancrx$Gender)
```

```
Japancrx$Risk<-as.factor(Japancrx$Risk)
Japancrx$Purpose<-as.factor(Japancrx$Purpose)

summary(Japancrx)

#Partition of data - train(80%) & test(20%)
set.seed(1)
indexjjj<-createDataPartition(Japancrx$`Credit Rating`,p=0.8, list=FALSE )
trainjjj<-Japancrx[indexjjj,]
validjjj<-Japancrx[-indexjjj,]

#Fit the Model
trainedj<-model.matrix(`Credit Rating`~Job+Age+Gender+Address
                +Purpose+Duration+Risk+Balance+income -1,data=train1)
head(trainedj)

train_label<- trainjjj[,10]
trainlabelj<-as.numeric(unlist(train_label))-1
train_matrix<-xgb.DMatrix(data=as.matrix(trainedj), label=trainlabelj)

testedj<-model.matrix(`Credit Rating`~. -1,data=validjjj)
head(testedj)

valid_label<- validjjj[,10]
validlabelj<-as.numeric(unlist(valid_label))-1
valid_matrix<-xgb.DMatrix(data=as.matrix(testedj), label=validlabelj)

Model<-xgboost(data=trainedjjj, label =trainlabelj,
          objective="binary:logistic", nthread=3, nrounds = 100)

attributes(model)
Model$evaluation_log
xgb.importance(model=Model)
```

```
#Confusion Matrix for the Japanese Data
pred<-predict(Model,tested)
head(pred)

predj<-as.integer(pred>0.5)
confusionMatrix(as.factor(predj), as.factor(validlabelj))
```

Australian Data

```
#Load Data
library(readxl)
Japancrx<- read_excel("C:/Users/F/Desktop/Evans School/Thesis/Data to be
used/Japancrx.xlsx")

#Statistics of the data
head(Australian)
glimpse(Australian)
str(Australian)
summary(Australian)

#Convert to factor
Australian$creditscreening <-as.factor(Australian$creditscreening)
Australian$discreditbadregion <-as.factor(Australian$discreditbadregion)
Australian$unmarried <-as.factor(Australian$unmarried)
Australian$married <-as.factor(Australian$married)
Australian$purchaseitem<-as.factor(Australian$purchaseitem)
Australian$monthlypayment <-as.factor(Australian$monthlypayment)
Australian$`Pos/Neg` <-as.factor(Australian$`Pos/Neg`)

summary(Australian)

#Partition of data - train(80%) & test(20%)
set.seed(1)
indexaaa<-createDataPartition(me$`Pos/Neg`,p=0.8, list = FALSE)
trainaaa<-me[indexaaa,]
```

```
validaaa<-me[-indexaaa,]
#Fit the Model
traineda<-model.matrix(`Pos/Neg`~.-1,data=trainme)
head(traineda)

trainlabel<- trainaaa[,14]
trainlabela<-as.numeric(unlist(trainlabel))-1
train_matrixa<-xgb.DMatrix(data=as.matrix(traineda), label=trainlabela)

testeda<-model.matrix(`Pos/Neg`~. -1,data=validaaa)
head(testeda)

validlabel<- validme[,14]
validlabela<-as.numeric(unlist(validlabel))
valid_matrixa<-xgb.DMatrix(data=as.matrix(testeda), label=validlabela)

Modela<-xgboost(data=traineda, label=trainlabela,
        objective="binary:logistic", nthread=3, nrounds = 100)

attributes(Modela)
Modela$evaluation_log
xgb.importance(model=Modela)

#Confusion Matrix for the Australian Data
preda<-predict(Modela,testeda)
head(preda)
predaa<-as.integer(preda>0.5)
confusionMatrix(as.factor(predaa), as.factor(validlabela))
```

German Data

```
#Load Data
library(readxl)
germancrx<- read_excel("C:/Users/F/Desktop/Evans School/Thesis/Data to be
used/germancrx.xlsx")
```

```
#Statistics of the data
head(germancrx)
glimpse(germancrx)
str(germancrx)
summary(germancrx)

#Convert to factor
germancrx$Creditability<-as.factor(germancrx$Creditability)
summary(germancrx)

#Partition of data - train(80%) & test(20%)
set.seed(1)
indexggg<-createDataPartition(germancrx$Creditability,p=0.8, list=FALSE )
trainggg<-germancrx[indexggg,]
validggg<-germancrx[-indexggg,]

trainedg<-model.matrix(Creditability~.-1,data=trainggg)
head(trained)

trainlabel<- trainggg[,1]
trainlabelg<-as.numeric(unlist(trainlabel))-1
train_matrixg<-xgb.DMatrix(data=as.matrix(trainedg), label=trainlabelg)

testedg<-model.matrix(Creditability~. -1,data=validggg)
head(testedg)

validlabel<- validggg[,1]
validlabelg<-as.numeric(unlist(validlabel))-1
valid_matrixg<-xgb.DMatrix(data=as.matrix(testedg), label=validlabelg)

Modelg<-xgboost(data=trainedg, label =trainlabelg,
        objective="binary:logistic", nthread=3, nrounds = 100)
```

```
attributes(Modelg)
Modelg$evaluation_log
xgb.importance(model=Modelg)

#Confusion Matrix for the German Data
predg<-predict(Modelg,testedg)
head(predg)

predg<-as.integer(predg>0.5)
confusionMatrix(as.factor(predg), as.factor(validlabelg))
```