CREDIT RISK ANALYSIS:

AN ASSESSMENT OF THE PERFORMANCE OF SIX MACHINE LEARNING TECHNIQUES IN CREDIT SCORING MODELLING

ABBA BELLO MUHAMMAD

BSc (KUST, Wudil)

MSC/STAT/018/0023

A DISSERTATION SUBMITTED TO THE DEPARTMENT OF STATISTICS, ALIKO DANGOTE UNIVERSITY OF SCIENCE AND TECHNOLOGY, WUDIL IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF MASTER OF SCIENCE IN STATISTICS

(MSc STATISTICS)

January, 2025

DECLARATION

I hereby declare that this work is the product of my research effort; undertaken under the supervision of Prof. Abubakar Yahaya and Dr. Olawoyin O. Ishaq and has not been presented by and will not be presented elsewhere for the award of a degree or certificate. All sources have been duly acknowledged.

Abba Bello Muhammad
(MSC/STAT/018/0023)

____
Signature/Date

CERTIFICATION

This is to certify that the research work for this dissertation and subsequent preparation of this dissertation by Abba Bello Muhammad with registration number MSC/STAT/018/0023 was carried out under our supervision.

Dr. Olawoyin Olatunji Ishaq	______________________
(Chairman, Supervisory Committee)	Signature/Date

Prof. Abubakar Yahaya	_____________________
(Member, Supervisory Committee)	Signature/Date

Abdulhameed Ado Osi	_____________________
(Head of Department)	Signature/Date

APPROVAL

This research work entitled “Credit Risk Analysis: An Assessment of the Performance of Six Machine Learning Techniques in Credit Scoring Modelling” has been examined and approved for the award of the degree of MASTER OF SCIENCE in Statistics.

Dr. Yusuf Bello ______________________
(External Examiner) Signature/Date

Abdulhameed Ado Osi ______________________
(Internal Examiner) Signature/Date

Dr. Olatunji Olawoyin Ishaq ______________________
(Chairman, Supervisory Committee) Signature/Date

Prof. Abubakar Yahaya ______________________
(Member, Supervisory Committee) Signature/Date

Abdulhameed Ado Osi ______________________
(Head of Department) Signature/Date

____
(SPGS Representative) Signature/Date

ACKNOWLEDGMENT

All praise belongs to Almighty Allah Who taught mankind by the pen, He taught him what he knows not. Peace and blessing of Allah be upon his exalted servant, the messenger, our beloved and mediator Muhammad (SAW), peace and blessing of Allah be to his offspring, his devoted companions, and those that follow their footsteps till the final day.

Special acknowledgement to my Supervisors Dr Olawoyin O. Ishaq and Prof. Abubakar Yahaya for being critical, diligent, and focused reviewers. Thank you for the support, guidance, and inspiration you rendered to me. I am motivated by the passion you have for research and academic excellence. I also would like to thank all members of the Department of Statistics, Aliko Dangote University of Science and Technology, Wudil. Particularly, Departmental P.G. Coordinator, Dr. Musa Uba Muhammad, the Head of Department, Abdulhameed Ado Osi and Prof S.U. Gulumbe for their keen interest and motivation for the success of this research dissertation. Your support was indeed not in vain.

In Addition, I owe my deepest gratitude and I am profoundly indebted to my lovely parent, family, friends, and my employer particularly my Area Manager Ismail Abubakar and my Branch Manager Bashir A. Sasa who provided me with data for the cause of this research. I appreciate your support, guidance, and motivation. Thank you, and May Allah bless you, Ameen.

DEDICATION

I dedicate this Dissertation to my parents and all those who affected my life positively by any means, preferably concerning my studies.

TABLE OF CONTENTS

Pages

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

Acronyms	Description
ANN	Artificial Neural Network
AUC	Area Under Curve
CART	Classification and Regression Tree
FAMD	Factor Analysis of Mixed Data
FN	False Negative
FNR	False Negative Rate
FPR	False Positive Rate
FP	False Positive
GA	Genetic Algorithm
GMM	Gaussian Mixture Model
GLM	Generalize Linear Model
KNN	K-Nearest Neighbour
LDA	Linear Discriminant Analysis
LTD	Limited
LVQ	Learning Vector Quantization
MCA	Multiple Correspondence Analysis
MLP	Multilayer Perceptron
NN	Neural Network
PCA	Principal Component Analysis
PSO	Particle Swarm Optimization
RBF	Radial Basis Function
ROC	Receiver Operative Curve
ROI-PA	Return Over Investment Per Annum
SVM	Support Vector Machine
ST	Special Treatment
TN	True Negative
TNR	True Negative Rate
TP	True Positive
TPR	True Positive Rate
UBM	Universal Background Model
VIF	Variance Inflation Factor
XGBoost	Extreme Gradient Boost

ABSTRACT

Credit scoring models are a scientific methodology adopted by credit providers to assess the creditworthiness of applicants. The primary objective of such models is to predict the potentiality of the loan applicant. For many decades, Banks face some risks in their businesses such as operational, market and credit risks. This research dissertation was designed to address the misclassification problem in microcredit instructions and identify the factors contributing to default in microcredit loans. Focusing on the assessment of six distinct credit scoring methodologies: Linear discriminant analysis, Logistic regression, artificial neural networks, Support vector machine, Decision tree, and *k*-nearest neighbour. The study leverages credit applicant data acquired from Gombe Microfinance Bank Ltd. The core objective of this research is to scrutinise and compare the performance of the aforementioned techniques in credit scoring modelling. To gauge the efficacy of the models, five essential performance metrics are employed: Area under the receiver operative characteristic curve, accuracy, precision, recall, and F1 score. These metrics provide insights into the models' predictive accuracy and their ability to distinguish between good and bad credit applicants. The results obtained from the experimentation phase reveal distinct performance levels for each technique. Specifically, *k*-nearest neighbour and artificial neural networks showcase exceptional prowess, yielding an AUC of 0.9833 and 0.9062, and an impressive precision score of 1 and 0.8065 respectively. In contrast, logistic regression and support vector machine demonstrate a moderate performance with an area under the curve value of 0.8537 and 0.8532 respectively. On other matric, support vector machine showed impressive high performance while the logistic regression performed poorly. Linear discriminant analysis and Decision tree exhibit comparatively good accuracy scores and achieved an AUC of 0.8494318 and 0.7524 respectively. This dissertation underscores the potential of *k*-nearest neighbour and Artificial neural networks as a superior method for credit risk analysis, supported by robust performance metrics. Although, all techniques achieve significantly good discriminative power and accuracy. The findings advocate for the adoption of modern techniques in credit scoring modelling, positioning *k*-nearest neighbour and Artificial neural networks as a valuable tool in financial institutions' risk assessment processes.

CHAPTER ONE

INTRODUCTION

1.1 Background

Microfinance banks and microfinance institutions' activities have grown significantly over time, and they play a crucial role in the national economy. Like other companies, the banks faces some risks in their businesses, such as credit, operational, and market risks. This research looks into the assessment of credit risk models (i.e., credit scoring), which is one of the most vital areas of study in the world. Given its role in the global crisis of 2008 and the subprime mortgage crisis of 2007, credit risk analysis has become more crucial than ever before. Basel III accord of the Basel Committee on Banking Supervision required the establishment of internal rating measures to assess the risk exposure of a financial institution. This results in banks improving their method of credit risk analysis (Bank for International Settlement, 2010). In addition, Hand and Henley (1997) assert that banks and financial institutions should enhance their credit scoring system not only due to the policy but rather the profit that might be obtained due to small improvements in the system.

Microcredit institutions and programs have been developed over the past year to cover a deficit in adequate saving and credit services for poor and small-scale entrepreneurs. Micro credits are recognized as a strategy for resource transfer to promote self-employment, income generation, poverty alleviation or eradication, and reducing the disparity between rich and poor as objective number 10 of the Sustainable Development Goals (United Nations SDGs, 2015).

Microcredit is a small amount of money lent to a person or group of persons (household or micro-enterprises) usually with zero collateral. Credit risk is a potential loss arising from the failure of some client to meet the obligation of the loan (i.e. loan or bonds will not be repaid either fully or partially). Most microlending is unsecured (i.e. the loan usually has zero collateral). Credit risk is the single largest risk most financial institutions battle with and it is as a result of the possibility that the loan or bond will not be repaid either fully or partially. It can be represented by the factor credit/default risk, loss, and exposure risk.

When a financial institution decides whether to issue a loan to a customer, typically, those customers are labelled as either “Good” or “Bad”. Good credit means the one that is likely to meet his financial obligation as and when due. While bad credit refers to the one that has a high likelihood of defaulting. Yap *et al*. (2011). All the pertinent information regarding the applicants such as economic conditions, marital status, and intentions, are considered when making those decisions.

Credit risk assessment is a process that provides a lender with the necessary tools that would help in making a decision on whether to grant credit to a new applicant or not, and how to deal with existing applicants; whether or not to raise their credit limits. Credit risk decisions are a major factor in determining the success of financial institutions due to the enormous losses that result from wrong decisions (Lahsasna *et al*., 2010). In financial institutions, credit risk assessment is the cornerstone of credit risk management and the decision-making process for loans, (Wu *et al*., 2010). One broadly used method for dealing with this classification challenge is the Credit Scoring technique.

Credit scoring is the set of techniques and decision models that help lending institutions or bodies in granting credit to their clients with possible minimum risk. It was first proposed by Fisher (1936) and the only method used was the discriminant and classification method.

1.2 Statement of the Problem

Misclassifying credit applicants has been a significant issue in credit risk management. Many techniques that have been employed by lenders to identify whom to consider as a good or bad client have been centred on an individual assessment of the analysis and the risk tolerance. This has left most financial institutions with difficulty in: formulating and selecting a statistical model that best minimizes the misclassification risk and, identifying the determinant factor in default/delinquency. There have been many researchers reported in the literature on predicting misclassification of credit applicant using machine learning algorithms. Tekić *et al*. (2021) evaluated the credit risk of agricultural enterprises in the Republic of Serbia using logistic regression and discriminant analysis models, Khemakhem and Boujelbène (2015) evaluated the predictive ability of linear discriminant analysis and artificial neural network, Ala’raj and Abbod (2015) checked the performance of hetrogenous and homogenous ensembled classifiers based on three classifiers logistic regression, support vector machines and artificial neural network, Zhao *et al*. (2015) assessed the accuracy of the decision tree and backpropagation algorithm on credit datasets, and Zhang *et al*. (2018) review weighed k -nearest neighbour for credit assessment.

However, the majority of these researches are not on microcredit data, and did not look into the factor/variable contributing to defaulting in microcredit loan. Thus, the following questions arises:

- Which statistical classification model will optimally minimise the misclassification risk in microcredit bank and institution?
- What are the determinant factors/covariates that contribute in default/delinquency?

1.3 Aim and Objectives

This research aims at assessing the performance of six machine learning techniques in credit scoring: linear discriminant analysis, logistic regression, artificial neural network, support vector machines, Decision Tree and *k*-nearest neighbours.

Objectives

I. To fit a credit scoring predicting model.
II. To identify the most efficient scoring model that will help in minimizing the misclassification risk in microcredit banks and institutions.
III. To identify the variable that contributes significantly to the default/delinquency in loans.

1.4 Significance of the Study:

As a business institution, the main goal of every financial institution is to maximize profit for its stakeholders. This work employed modern credit assessment techniques on risk modelling and how they influence the level of defaulting loans and outline the guide to the management on the way to reduced losses arising from loan defaults.

This study will add to the literature on supervised machine learning techniques and modern multivariate analysis and its application in finance. The result obtained can also serve as a basis and insight into classifying new and future individuals into their correct groups.

The results of this study may be of interest to other researchers and may open up new avenues for research in a related field. Will also serve as a resource for future studies on banking and customer satisfaction.

1.5 Scope and Limitation of the Study:

The research is delimited only to the assessment of the performance of six supervised learning models (i.e. linear discriminant analysis, logistic regression, artificial neural network, support vector machines, Decision Tree and *k*-nearest neighbours) in credit scoring in the Gombe Microfinance Bank. The study seeks to analyse the credit applicant classification problem in the Bank and determine the level of its risk exposure.

CHAPTER TWO

REVIEW OF RELATED LITERATURE

2.1 Introduction

A well-designed credit scoring methodology should have a good classification accuracy to discriminate new applicants and the existing clients as either good or bad. This is the fundamental purpose of credit scoring modelling. Traditional statistical methodologies like logistic regression, discriminant analysis, decision trees, and factor analysis, and the advanced statistical methods like Random forest, Artificial neural network, Mixture of experts, Generic algorithm, Learning vector quantization, Fuzzy adaptive resonance, Support vector machine, and Radial basis function, are the most prominent methods used for the classification of credit applicants.

2.2 Discriminant Analysis

This is a parametric technique developed by Fisher (1936), to discriminate among a priori known groups. Many researches have proved the discriminant analysis as one of the most broadly established techniques in the classification of credit client applications as either "good" or "bad" borrowers. It has two fundamental objectives viz: firstly, to discriminate among priori known groups and secondly, to predict new instances into their correct group (Fuentes, 2011; Mendoza *et al*, 2017).

The suitability of linear discriminant analysis in credit scoring has been in question due to the categorical nature of the credit data which is not normally distributed. Moreover, the variance-covariance matrices are not equal, these deficiencies can be solved by a more advanced statistical model (West, 2000). Whereas, Reichert *et al*. (1983) said this might not be a crucial limitation of linear discriminant analysis.

Eisenbeis (1978) noted some statistical difficulties in applying discriminant analysis in credit scoring in his earlier work in 1977. Complications such as the non-linear nature of the data, the categorical nature of the data (group definition), classification error prediction and prior probabilities inappropriateness among others should be considered when applying linear discriminant analysis. Despite these imperfections, (Greene, 1998; Abdou *et al*. 2009) concluded that discriminant analysis is nevertheless the most acceptable traditional technique in credit scoring.

Grablowsky (1975) used financial, demographic, and behavioral information of credit customers to conduct a binary stepwise discriminant analysis. The data was obtained from 200 borrowers through their loan application forms and a questionnaire. The estimated model predicts 94% of the validation sample accurately despite the data violation of the equal variance-covariance assumption. Tekić *et al*. (2021) evaluated the credit risk of agricultural enterprises in the Republic of Serbia. Comprehensive analysis was conducted using logistic regression and discriminant analysis models. The assessment included several key metrics such as specificity, sensitivity, overall classification rate, and area under the ROC curve. These quantitative measures allowed for a thorough evaluation of the creditworthiness of these enterprises, providing valuable insights into their financial stability and risk potential. The results indicate the superiority of the logistic regression method over the discriminant analysis in all the observed parameters.

The impact of five common feature selection techniques: *t*-test, logistic regression (LR), Particle Swarm Optimization (PSO), linear discriminant analysis (LDA), and genetic algorithm (GA), on the corporate credit risk index was compared by Liang et al. (2015). They also contrasted their results with those of the Bayesian, linear support vector machines (SVM), multi-layer perceptron (MLP), radial basis function (RBF), classification and regression tree (CART), support vector machine (SVM), and *K*-nearest neighbour (KNN) classifiers. They discovered that the Linear Discriminant Analysis (LDA) and t-test datasets from Australia, Japan and Germany, respectively, were the best overall using experimental data from those three countries. There is no ideal combination of the feature selection method and classification technology, and the t-test is the best method for choosing features. On average, however, logistic regression outperforms other methods in predicting bankruptcy datasets while the evolutionary algorithm outperforms other methods in credit score datasets.

Khemakhem and Boujelbène (2015) evaluated the predictive ability of two credit scoring methodologies: Linear Discriminant Analysis and Artificial Neural Network on 86 Tunisian companies from 2005 to 2007. The result indicates an outstanding predictive performance in Artificial Neural Networks. Antonio *et al*. (2013) compared multilayer perceptron Artificial Neural Network with traditional credit scoring methodologies: Quadratic Discriminant Analysis, Logistic Regression Model and Linear Discriminant Analysis on 5500 Peruvian Microfinance institutions' credit client sample data. The result identifies Artificial Neural Networks as superior. Finlay *et al*. (2012) assert that Logistic Regression and Linear Discriminant Analysis are the most prominent methodologies in credit risk analysis given their accuracy, interpretability and easy implementation.

2.3 Logistic Regression:

A logistic regression methodology is used to predict a probability of dichotomous outcomes with respect to the predictive variables. Thus, it can be used for credit risk analysis (Henly 1995).

Martin (1977) evaluates the predictive power of the distinct credit scoring models which include: the logistic model, Z-score and, zeta model on twenty-three bankruptcy banks using twenty-four financial indexes. The finding shows an outstanding performance in logistic regression.

Wu (2003) studied the financial difficulties faced by seventy companies under special treatment in China's A-share market from 1998 to 2000 using some statistical methods including univariate analysis of linear probability model, binary linear discriminant analysis, cross-sectional analysis and logit models. The paper asserts that the logit model has the highest predictive accuracy of 93.53%. Liang (2005) used the principal component as a dimensionality reduction technique to address the high correlation and high dimension in the listed companies' data and used the component as a covariate variable to modify the logistic regression model. The output indicated that the modified logistic regression model is more accurate in terms of prediction and classification than the simple regression. West (2000) assessed the performance of some traditional scoring methods that include: logistic regression, linear, kernel density estimation, discriminant analysis, decision trees, and *K*-nearest neighbour, and benchmarked with five Artificial neural networks models: mixture of experts, fuzzy adaptive resonance, multilayer perceptron, learning vector quantization, and radial basis function. The results showed logistics as the most accurate of the traditional methods.

Pławiak *et al*. (2019) used a probabilistic rough set model of credit scoring based on the two-step method of three-party decision-making. The logistic regression was used to find the likelihood of the sample belonging to the positive part and negative part. And established the relationship between the error decision cost and error tolerated by incorporating the hypothesis testing method and Bayesian decision theory together.

Lessman *et al*. (2015) hold a similar opinion to Finlay *et al*. (2012) asserting that Linear Discriminant Analysis and Logistic Regression are the most prominent methodologies in credit risk analysis given their accuracy, interpretability and easy implementation.

Guo (2020) investigated artificial neural networks and the logistic regression method through comparison. The result shows that artificial neural networks performed better than Logistic Regression in predicting new instances and could reduce investor's risk effectively.

Zhu *et al*. (2016) studied the quarterly report data of 77 small and medium-scale enterprises and 11 core enterprises between 2012 and 2013 in China's financial institutions and established a new credit risk assessment index for small and medium-scale enterprises. The result indicates a better performance in hybrid model III over the other model in predicting negative instances.

Gonçalves and Gouvêa (2021) developed a credit-scoring methodology based on the data from a large financial institution. The results show that the three models produced results that were appropriate for the dataset under consideration, which was provided by a sizable Brazilian retail bank. The findings from the logistic regression model were marginally superior to those from the artificial neural network model, and both performed better than the genetic algorithm model.

Hand and Henley (1997) review some credit scoring methodologies including some quantitative techniques such as mathematical programming, regression, expert systems, smoothing nonparametric methods, logistic regression, discriminant analysis, recursive partitioning, artificial neural networks and other different models were put in view. They conclude that the best model depends mainly on the data characteristics and its structure. The variables that typically distinguish between "good" and "bad" loans are the applicant's annual income, age, sex, number of dependents, marital status, home status, telephone, type of occupation, type of bank account, purpose of loan, and credit card. On a theoretical basis, logistic regression is considered a suitable statistical method, given that the two categories "good" credit and "bad" credit have been precisely explained. Suleiman *et al*. (2014) evaluated the prediction accuracy of logistic regression and linear discriminant models by employing principal components as independent variables for classifying credit applicant status using a credit applicant's data set. The result indicates that the principal component independent variable enhanced the predictive ability of logistic regression and linear discriminant analysis by reducing the dimension and multicollinearity in the dataset. The predictive performance of Logistic regression 91% is slightly higher than Linear Discriminant Analysis 80%.

2.4 Artificial Neural Network

An artificial neural network is the mathematical representation of the human brain system that acquires knowledge by experience. It's used in forecasting, classification, multifactorial analysis, and pattern recognition. The neural network structure mimics the structure of the human brain to solve complex data-driven problems, (Tucker, 1996). Edmond and Abba (2020) created an Artificial neural network using an ensemble method to test whether the model accuracy will be increased on the modified method. Real-world data Collected directly from a financial company was used to compare the models performances in solving customer classification problems. The test results show that the Ensemble Neural Network with 3 numbers of bootstrap can boost the accuracy up to 3% from the single Neural Network classifier. Teles *et al*.(2020). Used Artificial Neural Networks algorithms based on backpropagation and the naive Bayes algorithms to determine the best combination of parameters to use with Artificial Neural Networks and naive Bayes algorithms approaches. It was concluded that both the Artificial Neural Networks and naive Bayes algorithms model provide reliable results, but the former is more effective in predicting credit risk with an average score of 82%.

West (2000) checked the potentiality of five neural network models; a mixture of experts, learning vector quantization, multilayer perceptron, fuzzy adaptive resonance and radial basis function and bench-marked their result with some traditional statistical method; logistic regression, kernel density estimation, Linear discriminant analysis, nearest neighbour and decision trees. The result obtained showed logistic regression as the most accurate of traditional methods and artificial neural network as the most accurate of advanced methods which improve the credit scoring accuracy by 3% over the traditional methods. Jagric *et al*. (2011) noted that developing new credit risk models with improved forecast accuracy continues to be a major issue for banks. The use of artificial neural networks (ANNs), in dealing with the non-linear nature of financial data was emphasized, in building a credit scoring model. They used a logistic regression model for benchmarking, a neural network for retail loans, and a learning vector quantization (LVQ) to create a credit decision-making model. The dataset used was collected from the banks in Slovenia. The findings demonstrated that the LVQ model outperformed the logistic model and produced results with higher accuracy in the validation set. However, the type of the data structure also plays a vital role in model accuracy rating. Neural networks were employed as a credit scoring method by Zhirov *et al* (2021) to address the credit rating issue. Six neurons make up the input layer, one neuron makes up the output layer, and two hidden neurons are present in one hidden layer of the network. For a problem of this type, the backpropagation method of learning has been proven to be the most effective. Testing the system on various statistical sample sizes, with various numbers of epochs, and with various learning rates revealed that, in 95% of cases, the system's suggestions were accurate.

In another study, Malhotra and Malhotra, (2003) evaluated the ability of the artificial neural networks to classify loan applications into "good" or "bad" on a collective data set of twelve credit unions. The effectiveness of artificial neural networks and multiple discriminant analysis models in classifying credit applications was assessed. The result shows that the artificial neural network has an outstanding performance than the multiple discriminant analysis model in classifying credit applicants. However, by using the principal component analysis of the data as the independent variable, the performance of the multiple discriminant analysis model will be also good. Recently, Multilayer perceptron neural network (MLP) was employed by Blanco *et al*. (2013) to create a particular microfinance credit scoring model. They evaluated the effectiveness of the multilayer perceptron model in comparison to three other statistical methods: logistic regression, linear discriminant analysis, and quadratic discriminant analysis. The multilayer perceptron model achieved more accuracy at a reduced cost of misclassification, confirming its superiority to parametric statistical methods. However, the type of the data also affects how well these statistical models function. Some crumbs of proof endorse the superiority of neural network methodologies, showcasing that artificial intelligence techniques have enhanced the outcomes of credit risk analysis when compared with those provided by classical statistical approaches. Neural networks have drawn a lot of attention in credit risk evaluation recently, but it remains unclear whether they are superior to traditional statistical algorithms like logistic regression, (Abellan and Castellano, 2017). Baesens *et al*. (2003) assessed the discriminative ability of logistic regression classifier with the popular C 4.5 algorithm and artificial neural network rule extraction techniques such as trepan and neuro rule, which prove the superiority in classification accuracy.

Lee *et al*. (2002) Backpropagation neural networks were used with the conventional discriminant analysis method to examine their effectiveness in credit scoring. In comparison to the traditional neural network model, the developed hybrid technique converged significantly faster.

Additionally, Conventional logistic regression and discriminant analysis were outperformed by the hybrid method, and the accuracy of the developed methodology improved.

Bensic *et al*. (2005) investigated the characteristics of small scales business credit scoring and evaluated the effectiveness of logistic regression (LR), classification and regression trees (CART) and, artificial neural networks (ANN). The results show that the probabilistic artificial neural network model performed exceptionally well. Moreover, Koh *et al*. (2006) declared that artificial neural networks, decision trees and logistic regression are the best-performing credit-scoring methodologies.

Angelini *et al*. (2008) developed two neural networks credit scoring models using Italian data from small businesses. Due to their capacity to simulate the non-linear structure of the credit data, the models have achieved remarkable success in credit scoring. Their overall performance gives confidence in their ability to be effectively employed in credit risk assessment. The artificial neural network is regarded as black box technology because it's difficult to extort any symbolic information from its internal configuration.

Using the Germany Bank dataset, Khashman (2010) applied neural networks to evaluate credit risk. Nine learning strategies were created for three neural network models, and the various implementation results were contrasted. One of the learning methods performed well, with an average accuracy rate of 83.6%, according to the results. In reality, each applicant's credit score is necessary for the outcome. Therefore, the correctness of the group differentiation is our main focus. As a result, the credit scoring issue can be defined as assigning a given consumer a grade of excellent or bad based on the attribute qualities of other past customers. In numerous business applications, artificial neural networks (ANNs) have been utilised to solve issues with pattern recognition, optimisation, clustering, classification and forecasting.

2.5 Support Vector Machine

Li *et al* (2004) studied the support vector machine performance of 1000 credit clients of Chinese commercial banks, they asserted that the Support Vector Machine performed significantly better than the method used by the bank then, however, they did not in any way disclose the method used by the banks. This reduced the significance of the whole research.

Ala'raj and Abbod (2015) proposed a homogenous ensemble algorithms credit scoring classifier and heterogeneous ensemble algorithms credit scoring classifier based on three (3) classifiers: logistic regression, support vector machine and artificial neural networks. The result demonstrated that heterogeneous ensemble classifiers are the best in term of accurate prediction and enhanced performance in comparison with homogenous ensembled classifiers.

Schebesch and Stecking, (2005) assessed the performance of support vector machines and logistic regression on building and credit applicants data, and they found that support vector machine performed slightly better than logistic regression but not significantly

Huang *et al*. (2007) assessed the performance of support vector machines in comparison with backpropagation artificial neural networks, genetic algorithms and decision trees on the German and Australian credit datasets. They concluded that the support vector machines are competitive credit scoring methods when compared to the other method, but it's not significantly more accurate than the other method.

Bellotti *et al*. (2009) compared the performance of the support vector machine using logistic regression, linear discriminant analysis and *k*-nearest neighbour as a benchmark. They concluded that logistic regression and support vector machine tend to select the same variable as the most important one. Overall, the support vector machine performed slightly better than logistic regression.

Ghodselahi (2011) used 10 ensemble support vector machine classifier models on a standardized German credit dataset and found that it performed significantly better than all the individual support vector machine algorithms or logistic regression. However, logistic regression reported a worse area under curve (AUC) value than other classifiers.

2.4 Decision Tree

The decision tree is a single classifier machine learning classification algorithm. It divides large data into smaller homogeneous groups based on a set of rules and a particular aim. (Yap *et al*., 2011)

Davis *et al*. (1992) used a decision tree and multilayer perceptron neural network. Based on the artificial neural networks and single data partition, the outcomes show a comparable level of accuracy for decision trees and multilayer perceptron neural networks.

Yap, *et al*. (2011) conducted an assessment of the performance of decision tree, logistic regression and own credit scorecard model in credit risk analysis and concluded that no model outperforms the other.

Zhao *et al*. (2015) assessed the accuracy of the decision tree and backpropagation algorithm on credit datasets of German, Australian and Japanese banks, the result suggested that the decision tree performed slightly better than backpropagation algorithms. Although, both achieved high accuracy.

Galindo and Tamayo (2000) evaluate the efficacy of Decision trees, Artificial neural networks, *k*-nearest neighbour and Probit algorithms. The result obtained suggests the decision tree is a best-in default prediction with 8.31 average error rates.

2.4 *k*-Nearest Neighbor

Lubis *et al* (2021) studied the feature selection method with Binary particle swarm optimization (BPSO) on the *k*-nearest neighbour credit classification method, the result shows that *k*-nearest neighbour accuracy of 76.40% can be improved to 88.70% by Binary particle swarm optimization (BPSO) feature selection method.

Zhang *et al.* (2018) review weighed *k* -nearest neighbour for credit assessment by considering using kernel functions including Rectangular, Gauss, Tri-weight and, inversion on credit consumer data. The result reveals rectangular and Gauss as the best-performing kernel and their optimal performance reaches *k* =1. Zhang *et al.* (2018) studied a novel *k* -k-nearest neighbour algorithm with data-driven *k* -k-parameter computation. The approach was evaluated with 20 real datasets and the result suggests that the algorithms are much better than previous *k*-nearest neighbour algorithms in terms of data mining tasks (classification, regression and missing value imputation).

Zhou *et al* (2013) apply the hybrid SVM- *k*NN model in credit scoring to improve the prediction accuracy of the support vector machine. The result implied that the SVM-KNN hybrid model is a promising approach for credit scoring.

Henley and Hand (1996) assessed the credit consumer risk with the *k*-nearest neighbour by comparing the performance of the *k*-nearest neighbour and the range of other classification techniques and decision graphs. It was found that the *k*-nearest neighbour performed well achieving the lowest expected bad risk, hence concluded that it's practical to implement the *k*-nearest neighbour classification rule for credit scoring new applicants.

Abdelmoula (2015) analyse the bank credit risk on short-term loans of Tunisian commercial banks from 2003 to 2006. The *k*-nearest neighbour classifier algorithm was used on 924 credit records and the result suggests that the *k*-nearest neighbour shows a good classification rate of 88.63% and AUC of 95.6%.

Xia *et al*. (2017) Prior to creating a sequence centred on the gradient-boosted machine-integrated credit scoring model, data pretreatment was done, redundant variables were removed, and Bayesian parameter optimization for Extreme gradient boosting (XGBoost) was employed. The Bayesian parameter optimization model was superior to those employed in a web search, manual search and random search models, as evidenced by the error rate results and the accuracy of the confusion matrix.

When predicting bank failure, artificial neural networks are the most reliable, followed by logistic regression, *k*-nearest neighbour, linear discriminant analysis, and decision trees. Artificial neural network models are more robust, adaptive, and precise when compared to traditional methods. (Oreski *et al*., 2012; Khashman, 2010). On the contrary, a thorough analysis of 214 papers, books, and thesis that discuss the use of credit scoring in various contexts, particularly in finance and banking showed that there were no ideal statistical method for creating scoring models that can be applied in all situations (Abdou and Pointon, 2011).

CHAPTER THREE

MATERIALS AND METHODS

3.1 Introduction

This study applied six single classifier algorithms; linear discriminant analysis, logistic regression, artificial neural network, Support vector machine, Decision tree and *k*-nearest neighbour on some financial and nonfinancial factors to identify the best credit scoring model that aids in investment decisions and predicting the creditworthiness of a new client. This research used secondary data obtained from Gombe Microfinance Bank Limited (2020 to 2021). The data analysis was carriedout using R-statistical software.

3.2 Description of Variables

The variables under consideration in this study are extracted from the client loan application form, creditworthiness appraisal form, and client credit bureau report. The variables are: age, marital status, Gender, experience in present business, Amount applied, Number of households, Number of dependents, Amount approved, return over investment per annum (ROI PA), Capital assessed, Status of previous loan and current loan status as shown in Table 3.1 below.

Table 3.1: Description of variables

S/No.	Variables	Variable Description	Types of Variable
1	Age	Numeric	Input
2	Marital Status	Categorical (Single, Married, Widow and Divoced)	Input
3	Gender	Categorical (Male and Female)	Input
4	Experience in Business	Numeric	Input
5	Amount Applied	Numeric	Input
6	Number of Children	Numeric	Input
7	Number of dependents	Numeric	Input
8	Amount Approved	Numeric	Input
9	ROI PA	Numeric	Input
10	Capital Assessed	Numeric	Input
11	Credit History	Categorical (Good Loan, Bad Loan)	Input
12	Credit Worthiness	Categorical (Good Loan, Bad Loan)	Response(Output)

3.3 Preliminary Data Analysis

Preliminary data analysis has been employed to get the data ready for further analysis; this includes data cleaning, exploratory data analysis, feature selection, multicollinearity testing, and dimensionality reduction Han and Kamber (2006).

- **Data Cleaning:** The data used in this research had gone through a rigorous data cleaning process to address and resolve unwanted observations, outliers, structural errors, missing values, or incomplete data within the dataset. This is essential to assure the reliability and accuracy of the data and to guarantee that it is suitable for further analysis.
- **Exploratory Data Analysis:** This research employed univariate, bivariate, and multivariate exploratory data analysis to examine the pattern of the data and summarize its main characteristics.

- **Data Partition**

The data used in this research was partitioned into training and testing subsets. The training set is used to learn the pattern in the data, while the test set is used to evaluate the trained network's performance and gauge the model's generalizability. As recommended by (Jha, 2007), the training and testing were carried out using a similar dataset across all the techniques used (linear discriminant analysis, logistic regression, artificial neural network, support vector machine, decision tree and *k*- k-nearest neighbour).

3.4 Linear Discriminant Analysis Methodology

The linear score function of linear discriminant analysis according to Mendoza *et al*, (2017)

$$d_i^{'}(\boldsymbol{X}) =- \frac{1}{2}\boldsymbol{\mu_i^1}\Sigma^{-\mathbf{1}}\boldsymbol{\mu_i}+\boldsymbol{\mu_i^1}\Sigma^{-\mathbf{1}}\boldsymbol{X} + logp_i \tag{3.2}$$

$$d_i^{'}(\boldsymbol{X}) = d_{i0} + \sum_{j=1}^{p} d_{ij}x_j + logp_i \tag{3.3}$$

$where\ d_{i0} =- \frac{1}{2}\mu_i^1\Sigma^{-1}\mu_i \quad and \quad p_i = p_r(\pi_i); i = 1,2, \ldots., k$ is a Prior probabilities, $\mu_i = E(X/\pi_i);\ i = 1,2, \ldots., k$, $d_{ij} = \boldsymbol{\mu_i^1}\Sigma^{-\mathbf{1}}$ is a Population Means which is estimated by the sample mean vector, and $\Sigma = Var\left(\frac{X}{\pi_i}\right);\ i = 1,2, \ldots., k$ is Variance-covariance matrix which is estimated by pooled variance-covariance matrix.

Given a sample unit of credit client features $x_1, x_2, ... x_p$. The sample unit would be classified into the group that has the highest Linear Score Function. This is similar to classifying the group where the posterior probability of membership is the highest. The linear score function for each group will be computed, and the unit to the population with the highest score.

These parameters will be estimated from training data, in which the group membership is priori known. MASS package of R statistical software will be employed in a linear discriminant analysis.

3.5 Logistic Regression Methodology

The Binary logistic regression model will be done using the generalized linear model function “glm()” of R statistical software.

The logistic regression model according to Park (2013) is given by:

$$y_{ij} = logit\left(\frac{\pi}{1-\pi}\right) = \beta_o + \sum_{ij}^{pk} \beta_i x_{ij} + \varepsilon_{ij} \tag{3.4}$$

Where y_{ij} is a binary variable that takes a value of one if the criterion is satisfied, else it takes zero. It represents the loan status of the client i of category j. $i = 1,2, ... p$ $j = 1,2, ... k$

$$y_{ij} = \begin{cases} 1\ (Good/Active\ loan).\ if\ \pi \leq 0.5 \\ 0\ (bad/Default\ Loan)\ elsewhere. \end{cases} \tag{3.5}$$

$$where\ \pi \epsilon (0,1)$$

ε_{ij} is the random error term, β_o is an interception term and β_i is the coefficient of explanatory variable x_{ij}. $\pi = p(y_{ij} = 1)$ is the probability of a Bad/default loan and the term $\left(\frac{\pi}{1-\pi}\right)$ is defined as odds and has formula $\pi = \frac{odds}{1+odds}$ where

$$\pi = e^{\beta X} - \pi e^{\beta X} \tag{3.6}$$

$$\pi = \frac{e^{\beta X}}{1 + e^{\beta X}} \tag{3.7}$$

And $X = (1, x_1, x_2 ...... x_n)$ and $\beta = (\beta_0, \beta_1 ...... \beta_p)$.

The probability that client i of category j will default on the loan is (Tabachnick, 1996):

$$p(y_{ij} = 1/XY) = \frac{e^{\beta X}}{1+e^{\beta X}} \tag{3.8}$$

3.5.1 Logistic Regression R^2

The R^2 statistic is the percentage of the variance in the dependent variable that is accounted for by the independent variable. The larger the R^2 values, the larger the explained variation in the model. In logistic regression R^2 is estimated by Cox and Snell's R^2, Cohen's Nagelkerke's R^2, or Mcfdden's R^2 (Veall and Zimmermann, 1996).

This work used all three methods: cox and Snell's R^2, Cohen's R^2 and, and Nagelkerke's R^2 to check the goodness of fit of the logistic regression model. It was described as a good analogy to the R^2 in linear regression (Hosmer and Lemeshow, 1989)

R^2 Is given by

$$R_L^2 = \frac{D_{null} - D_{fitted}}{D_{null}} \tag{3.9}$$

where $D_{null} =- 2LL(null\, model)\ and\ D_{fitted} =- 2LL(fitted\, model)$ are the likelihoods for the model being fitted and null model (the deviance of the model before any predictors were entered) respectively. R_L^2 is the measure of how much the goodness of fit increases as a result of the inclusion of the independent variables. It is the proportional reduction in the absolute value of the log-likelihood measure. It ranges from 0 (the predictors are ineffective at predicting the result variable) to 1 (the model accurately predicts the outcome variable).

3.6 Artificial Neural Network Methodology

Artificial neural networks are non-parametric techniques with applications in classification, forecasting, pattern recognition, and multi-factorial analysis. The network's structure was inspired by the human brain (as shown in the diagram below) and it is adaptive to different environments by learning from experience.

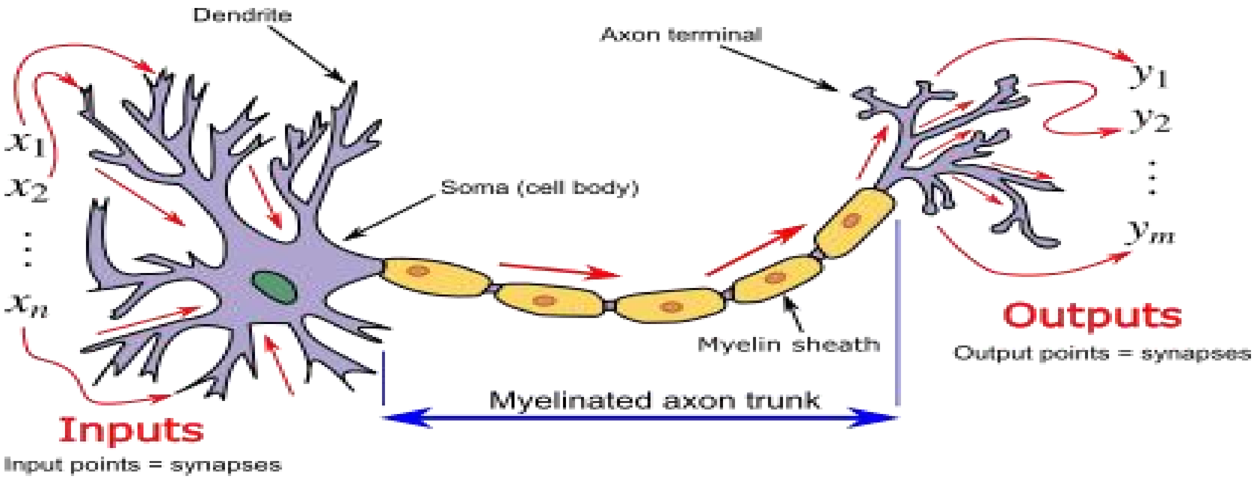


Figure 3.1: Analogy of Artificial neural networks structure to biological neurons.

3.6.1 Preliminary Data Analysis

A preliminary data analysis has been carried out to prepare the data for further assessment. The pre-process methods used in this work include variable selection, variable transformation, and data splitting into Training and Testing.

- **Variable Selection**

  May et al, (2011) assert that no widely accepted procedure defines the number of input variables to be selected in artificial neural networks. Redundancy caused by high correlation in the input variable makes variable selection in artificial neural networks very difficult.

  The variable selection method developed by Garson (1991) which is known as the relative importance of the variable was used. It selects the predictor variable by considering its relative contribution to a specific response variable in an artificial neural network. It is calculated by breaking down the model weights and selecting the most significant variable.

- **Variable Transformation**

  Variable transformation in artificial neural networks is the method of data preparation that aims at facilitating the network optimization process and maximizing the probability of obtaining a good result. It is also used to increase the rate of convergence of the training phase of artificial neural network modelling. The transformation techniques are:

  - Normalization (min-max normalisation)

- Standardization
- Batch normalization
- Rescaling, etc

In this research, variable transformation for the artificial neural network will be done using the normalization technique as recommended by (Rojas 1996), He stated that when using the backpropagation algorithm, an artificial neural network converges to output at a very slow rate and higher number of iterations if the variable in the input layer is not transformed by normalization.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}(U - L) \tag{3.10}$$

where,

- $x =$ original data
- $x' =$ normalized data, and $x' \in (U, L)$
- U, L are the upper and the lower values of the new range of the normalised data.
- x_{min} *and* x_{max}, are original data minimum and maximum values respectively.

Its linear transformation maintains all the distance ratios of the original vector after normalization, (Han and Kamber, 2006).

3.6.2 Determining the Artificial Neural Network Architecture

This research used a feed-forward Artificial neural network that consists of multiple inputs $(x_1, x_2, \ldots, x_n)$ and a binary output $y = 0 \; or \; 1$(McCulloch-Pitts neuron of McCulloch *et al.,* 1943). One hidden layer and the number of nodes are determined by the network with a set of nodes that gives the highest AUC value (discriminative ability). The sigmoid activation function and uniform random weight initialization method were also used, the weights are initialized by drawing random values from a uniform distribution within a specified range.

Hence, the network error was calculated until a specified minimum value of error was achieved.

3.6.3 Artificial Neural Network Model Building

Feed-forward artificial neural network model development was done using the *neuralnet* package of R-statistical software and a backpropagation algorithm was employed to adjust the weight of the network also a 0.01 threshold was set to minimize the error in the training phase.

The connectivity pattern of the network is assumed by a weight ω, which determines the network's structure of the network.

3.6.4 Back-Propagation

Back-propagation algorithms compute the first derivatives of an error function with respect to the network weights. These derivatives are then used to estimate the weights by minimizing the error function through an iterative gradient descent method.

A schematic diagram also displayed the backpropagation steps below in figure 3.2.

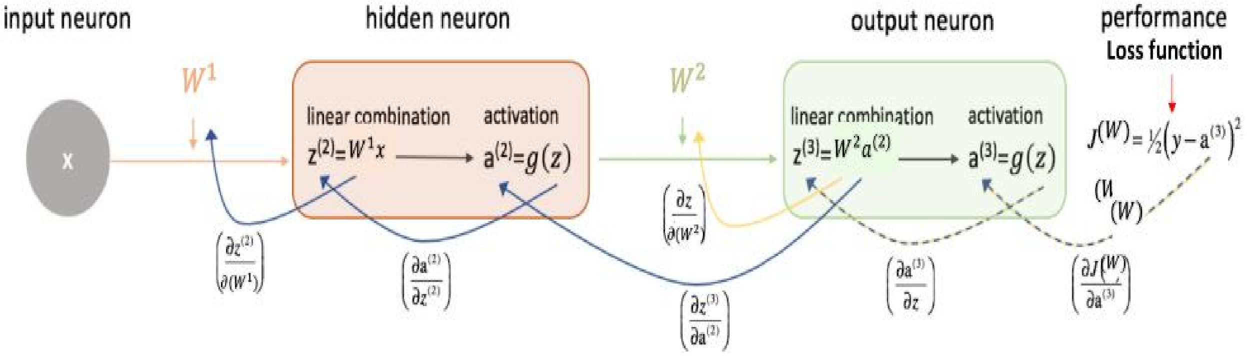

input neuron
hidden neuron
output neuron
performance
Loss function
W^1
W^2
x
linear combination
activation
$z^{(2)}=W^1x$
$a^{(2)}=g(z)$
linear combination
activation
$z^{(3)}=W^2a^{(2)}$
$a^{(3)}=g(z)$
$J(W)=\frac{1}{2}(y-a^{(3)})^2$
(W)
$\frac{\partial z^{(2)}}{\partial(W^1)}$
$\frac{\partial a^{(2)}}{\partial z^{(2)}}$
$\frac{\partial z}{\partial(W^2)}$
$\frac{\partial z^{(3)}}{\partial a^{(2)}}$
$\frac{\partial a^{(3)}}{\partial z}$
$\frac{\partial J(W)}{\partial a^{(3)}}$

Figure 3.2: Backpropagation Algorithm

3.7 Support Vector Machine

Support vector machine is a machine learning model used for classification, was first developed by Vapnik (1995). this work applies Support vector machine on credit data as a single classifier classification model. The Support Vector Machine algorithm is based on the idea of finding the optimal separating hyperplane between classes by maximising the class margin for a given data $[x_i, y_i]_{i=1}^{n}$ where the input is x_i and y_i is the corresponding observed binary class (client creditworthiness).

The maximum margin is a linear classifier that aims at finding the optimal separating hyperplane that divides the data points with the best possible margin, and the margin is the distance between the support vectors (nearest data points) of each class and the hyperplane. The high margin means a low misclassification probability on new data points as shown in figure 3.3 below.

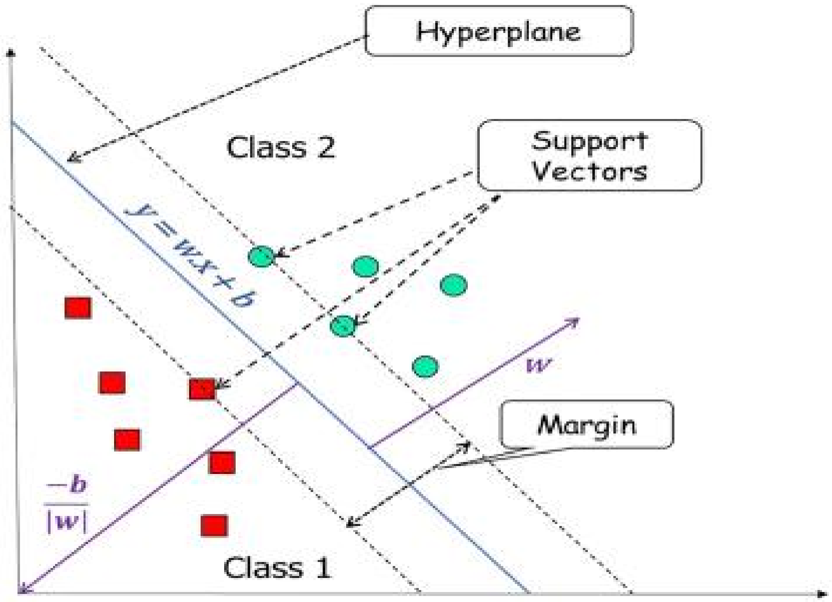


Figure 3.3: Support Vector Machine Single Classifier illustration

Boser et al. (1992) suggested a way to create nonlinear classifiers by applying the kernel trick in a nonlinearly separable scenario.

Suppose ψ(·) is a nonlinear function that maps the input space into a higher dimensional feature space. The separating hyperplane can be represented as:

$$g(x) = \omega^T \psi(x_i) + b = 0 \quad (3.11)$$

where $\omega \, and \, b$ are the normal vector of the hyperplane and the bias (which is scalar) respectively. The classifier for a linearly separable set in the feature space is as follows.

$$\omega^T \psi(x_i) + b \geq 1 \qquad if \;\; y_i = 1 \quad (3.12)$$

$$\omega^T \psi(x_i) + b \leq -1 \qquad if \;\; y_i = -1 \quad (3.13)$$

$$y_i(\omega^T \psi(x_i) + b) \geq -1 \qquad for \; i = 1, \dots., N \quad (3.14)$$

To deal with data that are not linearly separable, the equation can be generalised by putting a nonnegative variable $\xi_i \geq 0$.

$$y_i(\omega^T \psi(x_i) + b) \geq 1 - \xi_i \quad (3.15)$$

where the sum of ξ_i can be considered as the misclassification measurement.

According to the structural risk minimization principle, the minimization can be done by the optimization problem below.

$$\text{minimize } \psi(\omega, b, \xi_i) = \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{N}\xi_i \qquad (3.16)$$

C is a free regularization parameter controlling the trade-off between margin maximisation and tolerable classification error.

Subject to

$$y_i(\omega^T\psi(x_i) + b) \geq 1 - \xi_i \qquad for\ i = 1, \ldots, N; \qquad \xi_i \geq 0 \qquad (3.17)$$

According to Aizerman *et al.* (1964), the support vector machine decision function can be written as:

$$g(x) = sign(\sum_{i=1}^{m}\alpha_i y_i K(x_i, x_j\,) + b) \qquad (3.18)$$

Where: $\alpha_i\ and\ \beta_i$ are a set of Lagrangian multipliers and also the primal function can be written as

$$\mathrm{L}(\omega, b, \xi_i, \alpha_i, \beta_i) = \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}[y_i\alpha_i(\omega^T\psi(x_i) + b) - 1 - \xi_i] - \sum_{i=1}^{N}\xi_i\,\beta_i$$

(3.19)

The weight vector ω optimal solution is

$$\omega = \sum_{i=1}^{m}\alpha_i y_i \psi(x_i) \qquad (3.20)$$

$K(x_i, x_j)$ is the kernel function in the input space that satisfy $K(x_i, x_j) = \psi(x_i) \cdot \psi\big(x_j\big)$.

3.8 Decision Tree

The decision tree is the recursive partition of the feature space into regions, each region corresponding to a distinct class label using a split criterion. It is commonly used in credit scoring to fit the data and predict default. There are various decision tree algorithms used to establish classification rules, the most popular of them is the Classification and regression tree (CART). CART partition the tree branches according to the splitting criteria as shown in the figure 3.4 below.

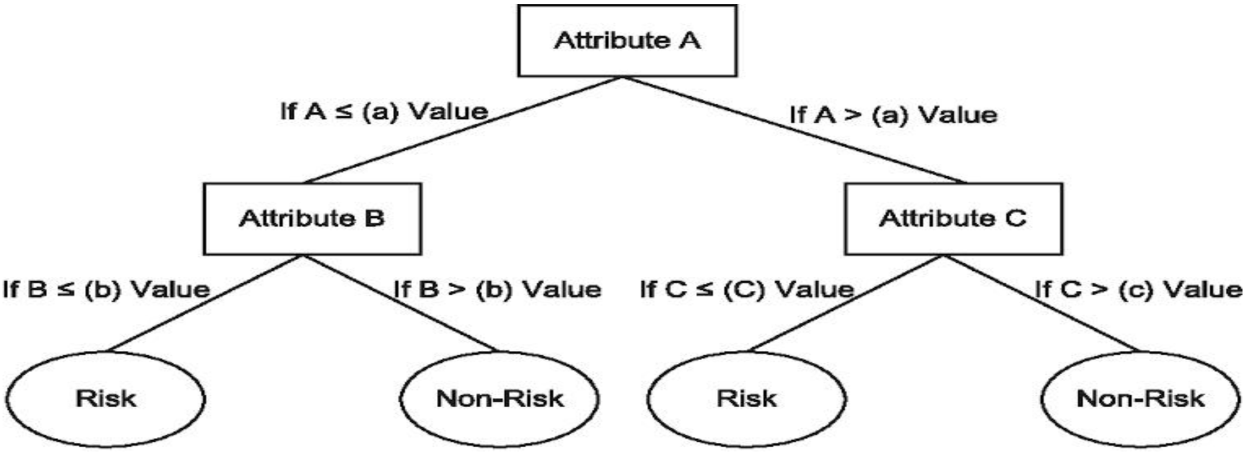


Figure 3.4: Decision Tree single classifier illustration

3.8.1 Structure of Decision Tree

The decision tree has nodes, branches and leaves which represent the feature test, the decision tree outcomes and the class label respectively. At each node, the algorithm selects the feature that best splits the data based on a given criteria such as Gini impurity, Entropy or information gain. The process continues until the stopping (Halt) criterion is achieved.

3.8.2 Splitting Criteria

Splitting criteria are criteria which determine how to partition the feature space at each node of the tree. Below are some commonly used splitting criteria for regression and decision trees.

Gini Impurity

Entropy

Information Gain

This work used the Gini purity criterion as a splitting criterion. The main aim of this splitting criterion is to maximize the Gini index which is the measure of the node purity. (Xu *et al.,* 2014), A low Gini index value indicates that a node contains predominantly observations from a single class. The Gini purity index can be written as

$$G = \sum_{k=1}^{p} \hat{p}_{mk}(1 - \hat{p}_{mk}) \qquad (3.21)$$

where, $0 \leq G \leq 1,$ (0 cleanest, all the instances are of the same class, 1 messiest, instances slits evenly across all classes)

$k = cases, \hat{p}_{mk}$ is the proportion of the training observations,

$0 \leq \hat{p}_{mk} \leq 1,$

3.8.3 Tree Pruning

Perhaps a better way to grow a very large tree is by recursive binary splitting and then pruning it back to obtain a subtree as the final model that gives the lowest test error. Typically, test errors can be examined via validation (K-fold cross-validation). However, it would be computationally expensive to calculate the test errors for each possible subtree. Hastie *et al.,* (2001)

3.8.4 Cost Complexity Pruning

Provides another alternative to control the size of the tree. For a given tree T with |T| number of nodes and a turning parameter (α) the cost function $C_\alpha(T)$ is given by

$$C_\alpha(T) = \sum_{M=1}^{|T|} \sum_{I:x_i \in R_m} (y_i - \hat{y}_{Rm})^2 + \alpha|T| \qquad (3.22)$$

3.9 *k*-Nearest Neighbour

k-nearest neighbour is a nonparametric supervised learning classification with application in both regression and classification. It was first developed by (Fix and Hodges, 1951) and later expanded by (Thomas and Hart, 1967) It remains the simplest and most widely used family of lazy learning algorithms. It operates based on the similarity principle (principal of proximity), and it classifies new instances by a majority vote of its *k*-nearest neighbour in the training test. The figure 3.5 below illustrate *k*-nearest Neighbor Single Classifier

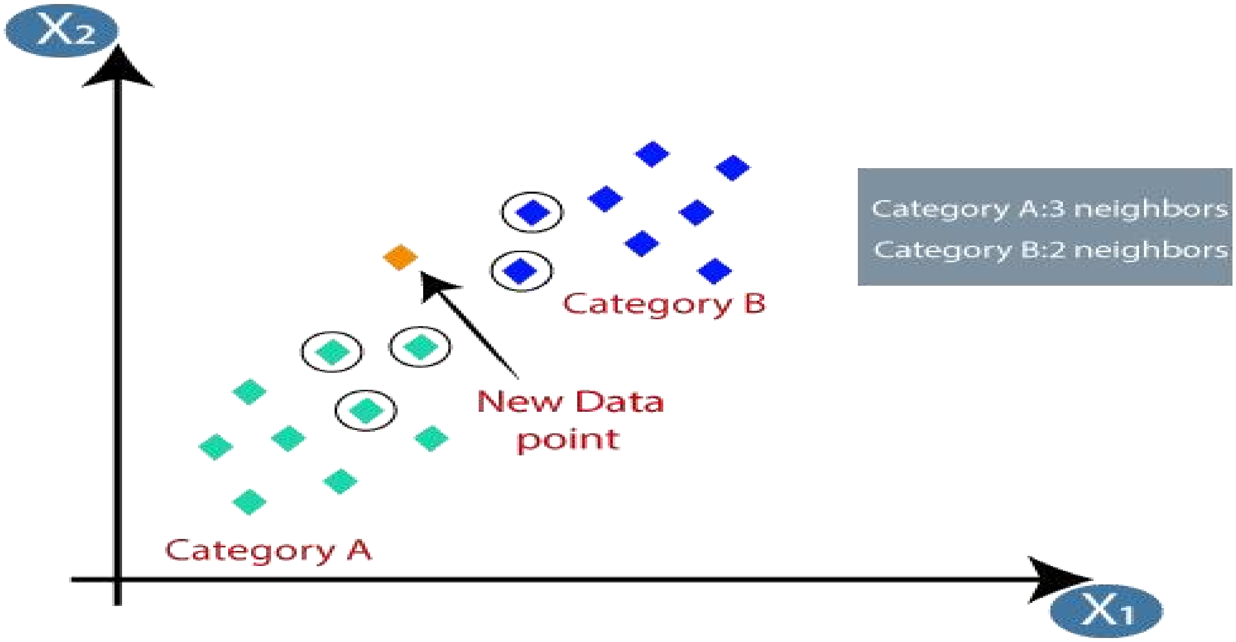


Figure 3.5: *k*-nearest Neighbor Single Classifier Illustration

3.9.1 *k*-Hyperparameter

The *k*-hyperparameter defines the number of nearest neighbours considered when predicting the class of a given instance. The small *k* makes the model more vulnerable to noise from nearby points as a result of high variance and overfitting. If the *k* is large the model becomes less sensitive to local variations in the data which leads to potential misclassification of instances as a result of high bias and underfitting.

Choosing the best method to determine the optimal *k* in *k*-nearest neighbour depends on several factors including the nature of the dataset and the desired level of model performance. In this work, the value of *k* can be determined by the *k*-fold cross-validation method thus, the *k* with a good classification accuracy for the test data is chosen.

3.9.2 Distance Metric

In the case of credit data, which by nature is the combination of both numerical and categorical data, it is quite essential to choose a distance metric that appropriately captures the similarities or otherwise of the data points. This work employed the Euclidean distance metric to assess the similarities or dissimilarities between the data points in the feature space.

3.10 Model Performance

The linear discriminant analysis, logistic regression, artificial neural network, support vector machine, decision tree and *k*-nearest neighbour models performances were assessed using the area under the receiver operative characteristics curve to determine their discriminative ability as recommended by Jaimes *et al.* (2005). Due to the imbalanced nature of our dataset, this work employed precision, accuracy, recall and F1 score in evaluating the performance of the model.

The area under the receiver operative characteristics curve, precision, accuracy, recall and F1 score performance metrics can be obtained through the following steps:

- Construct the confusion matrix and set a threshold to classify the probabilities as “Good client” or “Bad client”. All predictions at or above the threshold are classified as “Good Client” otherwise “Bad Client”. The number of Good clients that are classified as good are called True Positive (TP) and those classified as bad while they were good are called False Negative (FN), and the number of bad clients classified as bad are called True Negative (TN) while those classified as good while they were really bad are called False Positive (FP).

Table 3.2: Confusion Matrix

	Actual	
Predicted	Positive	Negative
Positive	**True Positive(TP)** Predicted as "Good Client" and they were "Good Client"	**False Positive(FP)** Predicted as "Good Client" but they were "bad Client"
Negative	**False Negative(FN)** Predicted as "Bad Client" but they were "Good Client"	**True Negative(TN)** Predicted as "bad Client" and they were "bad Client"

- The mathematics of sensitivity (True Positive Rate) and specificity (True Negative Rate) in model performance are shown below:

  $Sensitivity\ (\boldsymbol{TPR}) = \frac{TP}{TP+FN}$ *(3.23)*

  $False\ positive\ rate\ (\boldsymbol{FPR}) = \frac{FP}{TN+FP}$ *(3.24)*

  $Specificity\ (\boldsymbol{TNR}) = \frac{TN}{TN+FP}$ *(3.25)*

  $False\ negative\ rate\ (\boldsymbol{FNR}) = \frac{FN}{TP+FN} = (1 - Specificity)$ *(3.26)*

- Precision: $Precision = \frac{TP}{TP+FP}$ *(3.27)*
- Accuracy: $Accuracy = \frac{TP+TN}{N}$ (3.28)
- Recall: $Recall = \frac{TP}{TP+FN}$ (3.29)
- F1 Score: $F1 = 2\left(\frac{Precision\ x\ Recall}{Precision+Recall}\right)$ (3.30)
- Area Under the Receiver Operative Characteristics Curve: Plot TPR against FPR at different thresholds, then join the dots with the line. The area covered below the line is called AUC. The higher the AUC the better the discriminative ability of the model. Hence, the model with the highest AUC will be considered the best. The variable with the highest coefficient (weight) will be considered an important factor in investment and loan decisions.

CHAPTER FOUR

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter assesses the performance of some classification algorithms, including linear discriminant analysis, logistic regression, artificial neural networks, Support vector machine, Decision tree and K-nearest neighbour and identifies the classification technique that best classifies credit applicants using ROC and the variables that contribute significantly to default and delinquency.

4.2 Preliminary Data Analysis

4.2.1 Data Structure

The data consists of eleven (11) independent variables and one (1) dependent variable of six hundred and sixty (660) client credit records obtained from Gombe Microfinance Bank Limited (2020 to 2021) which contain both numeric and non-numeric types. All the columns have the correct data types.

4.2.2 Data Cleaning and Manipulation

The data undergoes a cleaning process to get rid of duplicates, incorrect formats, and outliers from the dataset to ensure the conclusions drawn from the analysis are based on accurate and high-quality information.

4.2.3 Exploratory Data Analysis

a. Univariate Exploratory Data Analysis

i. Graphical Representation of Variable

The display of the frequency distribution of the numeric and non-numeric variables in the data set is presented in **Appendix A1** using histograms and bar charts, respectively.

ii. Descriptive Statistics:

Table 4.1: Descriptive Statistics of the Dataset for Continuous Variable

	Age	Experience in Business	Amount Applied	No. of Children	No. of Dependent	Amount Approved	ROI PA	Capital Assesed
Missing Value	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Minimum	19	12	10,000	0	0	10,000	24.82	5,500
Maximum	80	492	150,000	30	20	200,000	6,036	500,000
Range	61	480	140,000	30	20	190,000	6,011.18	494,500
Median	37	120	250,000	5	1	20,000	642.45	35,000
Mean	37.81	131.84	2,7946.97	4.72	2.13	23,322.73	764.02	42615.15
S.E. Mean	0.37	3.50	455.00	0.12	0.13	383.78	21.55	1,166.29
Variance	88.27	8063.57	136636030.26	9.05	10.54	97209042.63	306582.8	897752273.88
Std. Deviation	9.4	89.80	11,689.14	3.01	3.25	9859.46	553.70	29962.51
Coef. of Var.	0.25	0.68	0.42	0.64	1.52	0.42	0.72	0.70
Skewness	0.48	0.95	3.00	1.58	2.60	9.96	3.56	6.43
Skewness S.E	2.55	5.01	15.76	8.33	13.66	52.34	18.73	33.82
Kurtosis	0.22	0.56	20.34	8.61	8.92	161.14	22.18	83.53
Kurtosis S.E	0.59	1.47	53.52	22.65	23.47	424.11	58.38	219.83
Normal Test W	0.98	0.91	0.77	0.90	0.68	0.48	0.73	0.64
Normal Test P-Value	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

From the above Table 4.1, there is a wide merging between the minimum and maximum values of the variables, indicating that the data needs to be scaled prior to the main data analysis. The values of the kurtosis and skewness show that the data is largely skewed. Furthermore, the Shapiro Wilk's Test P-value indicates the non-normality of the variables in the dataset.

b. Bivariate Exploratory Data Analysis:

Bivariate tools are used to explore the relationship between two variables and identify the direction, potential, the strength of their relationship. From the correlation matrix in **Appendix A2**, the bivariate correlation is weak and the direction of the relationship is not specific (the relationship between some variables moves in a positive direction and some in a negative direction)

4.3 Linear Discriminant Analysis

4.3.1 Model Building

The linear discriminant analysis models were trained to classify instances of loan applicants into two groups: "good" and "bad" credit using the *MASS* package of R statistical software, it was used in the training and test phase of the linear discriminant analysis models. The *ROCR* package was used in assessing model discriminative ability (ROC plot). The data used was partitioned into training and testing. (**Appendix B8** Presents the R code for Linear Discriminant Analysis).

4.3.2 Results

The prior probabilities indicate the proportion of instances belonging to each class in the test dataset. In the case of this research, the prior probability of the "good" class is 0.7407407, suggesting that approximately 74.07% of the instances are labelled as creditworthy, while the prior probability of the "bad" class is 0.2592593, indicating that around 25.93% of the instances are labelled as noncredit worthy. Other measures of the discriminative ability and accuracy of the model are summarised in the tables below:

Table 4.2: Linear Discriminant Analysis Confusion Matrix:

	Bad	Good
Bad	25	06
Good	13	90

Table 4.3: Linear Discriminant Analysis Measures of Discrimination and Precision

AUC	0.8490
Precision	0.8065
Accuracy	0.8582
Recall	0.6579
F1 Score	0.7247

From the Table 4.3 above, a precision of 0.8065 indicates that the linear discriminant analysis model correctly identified approximately 80.65% of the applicants who are actually creditworthy as creditworthy.

Accuracy measures the overall correctness of the model across all classes. An accuracy of 0.8582 means that the model classified approximately 85.82% of all credit applicants correctly.

A recall of 0.6579 points out that the model correctly identified approximately 65.79% of credit worthy applicants.

An F1 score of 0.7247 shows that the model achieves a very good balance between precision and recall.

AUC (Area Under the Curve) is a metric commonly used in binary classification tasks to assess the model's ability to distinguish between the two classes. A value of 0.8494318 for AUC indicates that the linear discriminant analysis model has a good level of discrimination, with a high probability of ranking a randomly chosen "Good" instance higher than a randomly chosen "Bad" instance.

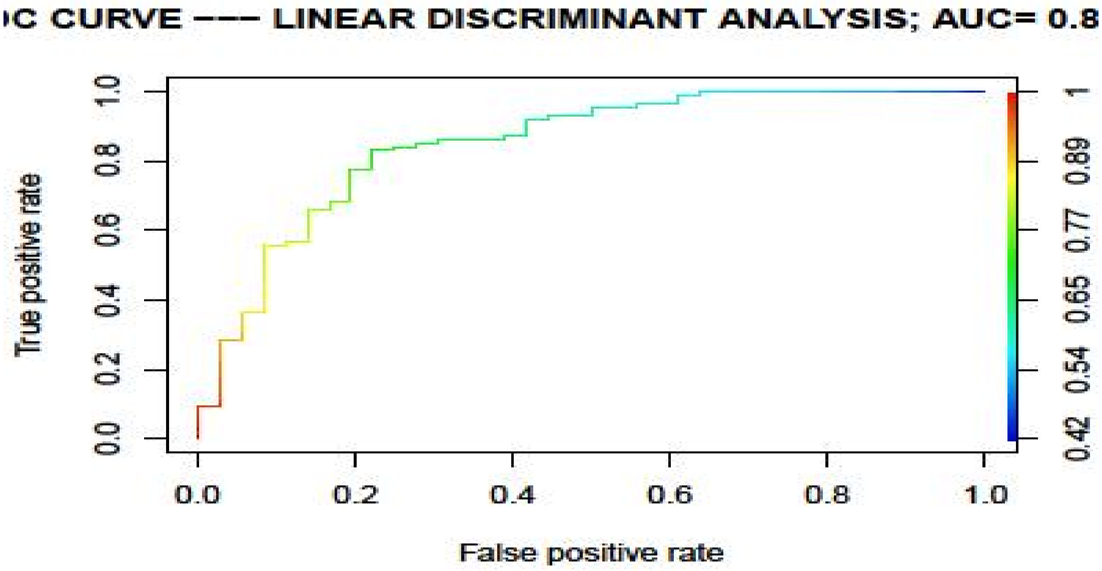


Figure 4.1: Linear Discriminant Analysis ROC

4.4 Logistic Regression

4.4.1 Model Building

The Logistic Regression model was trained using R statistical software to classify instances into two classes, Good and Bad. On the same dataset and the same partition that was used in linear discriminant analysis. *mlogit, caret, leaps, car* and *pROC* software packages were employed. **Appendix B9** Presents the R Code for Logistic Regression.

4.4.2 Results

Table 4.4: Logistic Regression Chi-Square Summary

ModelChiSqr(χ2)	ChiSqrDF	ChiSqrProb
111.2639	5	0.00

The differences between null deviance and residual deviance are the model chi-square. The chi-square value represents the logistic regression model’s goodness of fit. It measures the discrepancy between the expected and observed values based on the model. In the case of this work, a chi-square value of 111.26 and a p-value of 0.00 (assuming it’s very close to zero) indicate extremely strong evidence that the logistics regression model is highly significant. It implies that the explanatory variables included in the model have a significant impact on the predicted outcomes.

Table 4.5: Logistic Regression R^2 Summary

Hosmer-Lemeshow	Cox and Snell	Negelkerke
0.2216975	0.5893899	0.6002199

Hosmer-Lemeshow test: The Hosmer-Lemeshow statistic is used to assess the model's calibration (how well the predicted probabilities match the observed probabilities). A higher value indicates better calibration, suggesting that the model's predicted probabilities align well with the observed outcomes. Thus, the Hosmer-Lemeshow statistic of 0.2216975 means a fairly good fit to the data.

Cox and Snell's R^2: The Cox and Snell's R^2 value of 0.5893899 is a measure of the percentage of the variability in the response variable that is explained by the model. A value closer to 1 indicates a better fit of the model. The Cox and Snell's R^2 value of 0.5893899 suggests an acceptable moderate fit to the data.

Nagelkerke's R^2: The Nagelkerke's R^2 value of 0.6002199 is another measure of the proportion of the variability in the dependent variable explained by the model. Similar to Cox and Snell's R^2, a higher value suggests a better fit of the model. The Nagelkerke's R^2 value of 0.6002199 suggests an acceptable moderate fit to the data.

Table 4.6: Logistic Regression Confusion Matrix

	Bad	Good
Bad	16	91
Good	14	4

Table 4.7: Logistic Regression Performance Metrics Scores

AUC	0.8537
Precision	0.1495
Accuracy	0.1600
Recall	0.5333
F1 Score	0.2335

The AUC (Area Under the ROC curve) value of 0.8537 represents the discriminative power of the logistic regression model. A higher AUC means better discrimination between the positive and negative outcomes.

The precision of 0.1495 suggests that the logistic regression model correctly identified approxim ately 14.95% of the applicants who are actually creditworthy as creditworthy.

Accuracy measures the overall correctness of the model across all classes. An accuracy of 0.1600 means that the model classified approximately 16.00% of all credit applicants correctly.

A recall of 0.9560 points out that the model correctly identified approximately 95.60% of credit worthy applicants. An F1 score of 0.8833 shows that the model achieves a good balance between precision and recall.

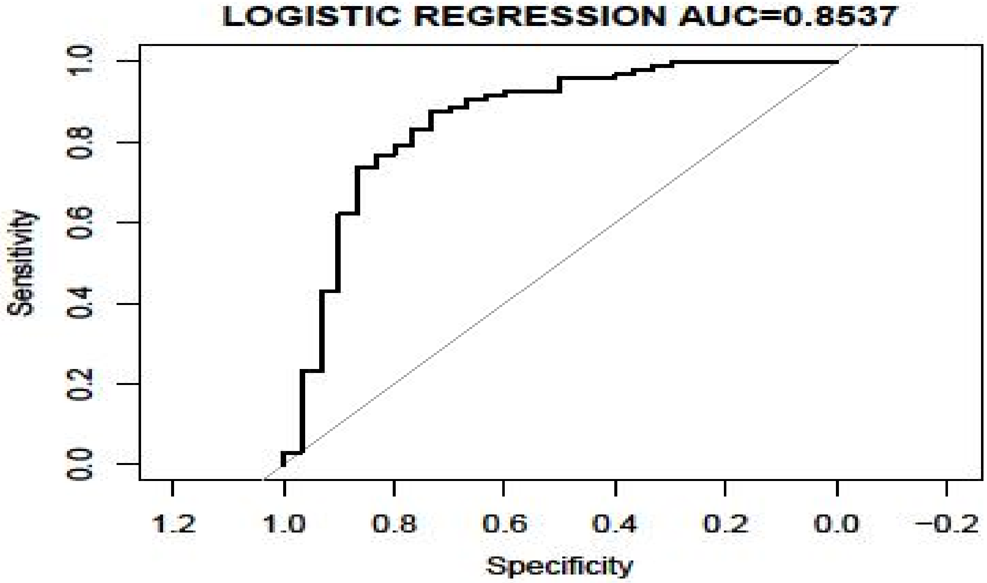


Figure 4.2: Logistic Regression ROC

In summary, the logistic regression model has a statistically significant association with the outc ome variable (as indicated by the chi-square values and probabilities). The model’s goodness of f it (Hosmer and Lemeshow, Cox and Snell, Nagelkerke) suggests an acceptable fit to the data. A UC value of 0.8537 indicates a reasonable level of discrimination power in distinguishing betwee n creditworthy and non-creditworthy applicants.

4.5 Artificial Neural Network

4.5.1 Model Building

An artificial neural network was developed on the same dataset that was used in linear discriminant analysis and logistic regression on the same partition size. *Neuralnet, NeuralNetTool,* and *pROC* packages of R statistical software were employed. The networks have one hidden layer, the number of nodes was determined by the network that has the set of nodes that gives the highest AUC value (discriminative ability). The network was trained using the *Rprop* algorithm. **Appendix B13 to B18** presents the R code for Artificial Neural Network.

4.5.2 Variable Selection:

Variable selection is a preliminary data analysis, usually done prior to training the network to save resources and time May et al. (2011). The method of variable selection proposed by Garson (1991) was employed to select fewer variables to use in the neural network model by discarding the variable with less or zero contribution toward the neural network models' accuracy. The results obtained from the variable selection are shown in figure 4.3.

```
rel.imp              x.names
-0.107057492       Capital Assessed
-0.050645971       Number of Dependents
-0.021225921       Age
-0.017994878       Experience in Business
-0.009464623       Marital Status
 0.000000000       Gender
 0.049840324       Credit History
 0.078679242       Number of Children
 0.340513038       ROI PA
 0.703285499       Amount Applied
 1.000000000       Amount Approved
```

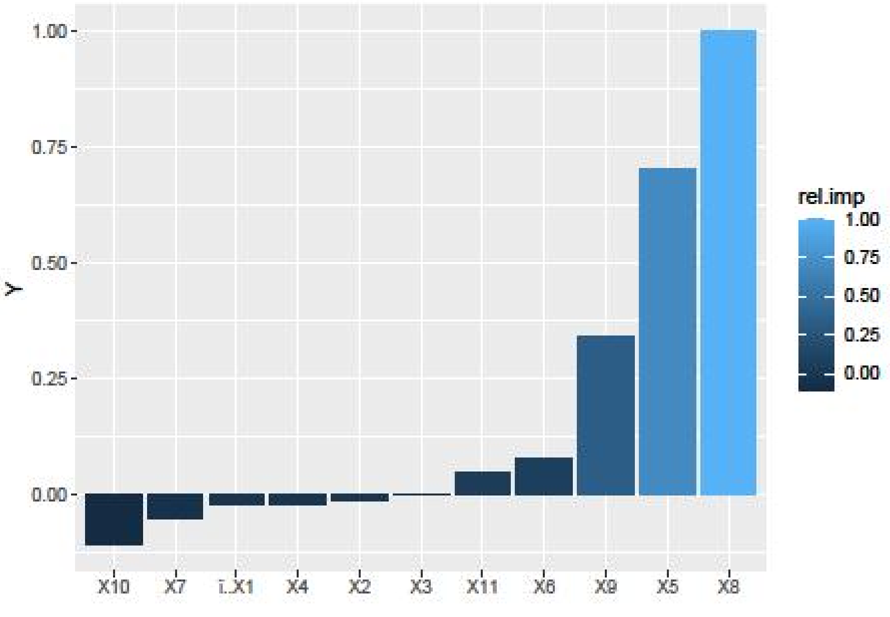


Figure 4.3: Relative importance of the explanatory variables.

Although Garson algorithms do not provide a fixed threshold for variable selection in artificial neural network models. The threshold for variable incursion depends on the user's discretion. A recommended top-down approach is used in this research, where we start with the most important variables and gradually include the subsequent ones until a satisfactory level of model performance is reached.

Variables with relative importance greater than or equal to 0.05 are considered to have the highest relative importance to the dependent variable (creditworthiness of the loan applicant). The variables are: Amount applied, Number of Children, Number of dependency, Amount Approved, ROI PA and Capital Assess.

4.5.3 Results

Table 4.8: Artificial Neural Network Results Summary

Number of Nodes	Error	Steps	Area Under Curve (AUC)
1	51.31	6086	0.9062
2	43.72	5825	0.7856
3	41.63	56074	0.8583
4	28.09	51527	0.8240
5	38.22	66481	0.8785
6	32.02	86727	0.7598

The above table shows the results of an artificial neural network with different numbers of hidden nodes. The artificial neural network with one hidden node was selected as the best, with the highest AUC (discriminative ability) of 0.9062. The confusion matrix and performance metrics scores are shown in Table 4.9 and 4.10, respectively.

Table 4.9: Artificial Neural Network Confusion Matrix:

	Bad	Good
Bad	25	6
Good	13	90

Table 4.10: Artificial Neural Networks Performance Metrics Scores:

AUC	0.9062
Precision	0.8065
Accuracy	0.8582
Recall	0.6579
F1 Score	0.7247

AUC (Area Under the Curve): A value of 0.9062 for AUC indicates that the Artificial neural network model has a high level of discrimination, with a high probability of ranking a randomly chosen positive instance (belonging to one of the output classes) higher than a randomly chosen negative instance.

Precision: The Artificial neural network model achieved a high precision of 0.8065, suggesting that it accurately detected around 80.65% of positive occurrences out of all positive instances classified as positive.

Accuracy: an accuracy of 0.8582 means that the artificial neural network model classified approx imately 85.82% of all credit applicants correctly.
A recall of 0.6579 shows that the artificial neural network model correctly identified approximat ely 65.79% of creditworthy applicants.
An F1 score of 0.7247 shows that the model achieves a very good balance between precision and recall.

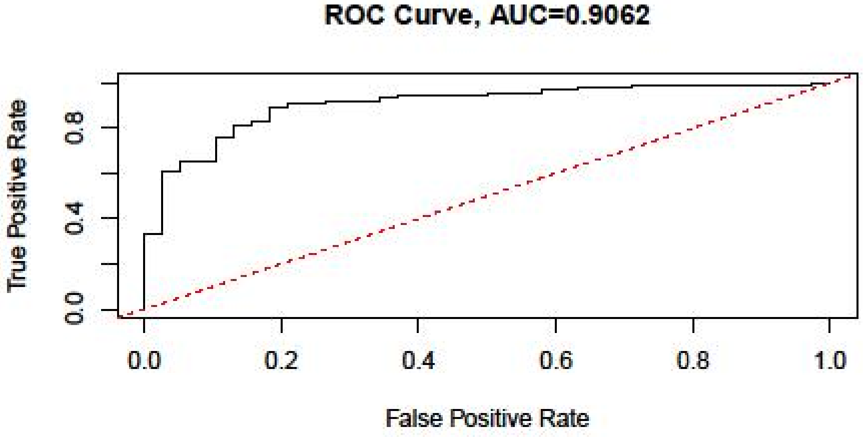


Figure 4.4: Artificial Neural Network ROC

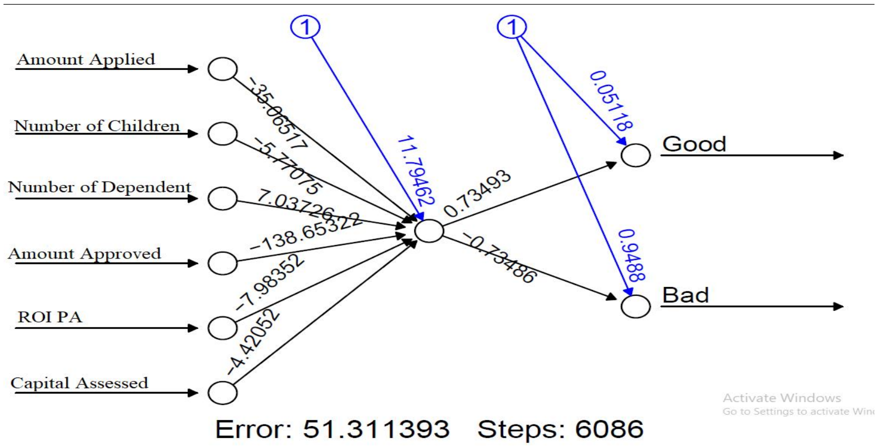
Overall, the artificial neural network model with 6 input nodes, 2 output nodes, 1 hidden layer, and 1 hidden node trained using the Rprop algorithm demonstrates high precision (0.8065), and good discriminative ability (AUC = 0.9062). These results indicate that the artificial neural network model is effective in classifying instances and has the potential to make accurate predictions on unseen data. Hence, the model has the ability to generalize its results. The architecture of the final selected model is summarized in the Table 4.11 below.

Table 4.11: The Architecture of the Final Selected Artificial Neural Networks Model

Number of hidden layers	1
Number of hidden nodes	1
Number of input variable	6 (Amount Appled, Number of Children, Number of Dependent, Amount Approved, ROI PA and Capital Assessed)

Number of output	2 (Y= Good/Bad)
Activation function	Logistic
Algorithm	Rprop
Number of repetition	20
Threshold	0.01

Figure 4.5: Final selected Artificial Neural Network model Structure:



4.6 Support Vector Machine

4.6.1 Model Building

The support vector machine was trained using R statistical software to classify instances into two classes: "Creditworthy" and "non-creditworthy" using “e1071” R packages on the same dataset, using the same partitioning proportion that was used in linear discriminant analysis, Logistic Regression and Artificial neural network. The R package *pROC* and *caret* were used to determine the performance metrics scores and plot the ROC curve. (**Appendix B9** display the R Code for Support Vector Machine).

4.6.2 Results

Table 4.12: Support Vector Machine Confusion Matrix:

	Bad	Good
Bad	87	19
Good	04	31

Table 4.13: Support Vector Machine Performance Metrics Scores:

AUC	0.8532
Precision	0.8208
Accuracy	0.8369
Recall	0.9560
F1 Score	0.8833

From Table 4.13 above;

The AUC (Area Under the ROC curve) value of 0.8532 means the support vector machine has a strong discriminative power.

Precision: The support vector machine achieved a high precision of 0.8208, suggesting that it accurately detected around 82.08% of creditworthy clients out of all creditworthy clients.

Accuracy: an accuracy of 0.8369 means that the support vector machine model classified approxi mately 83.69% of all credit applicants correctly.

A recall of 0.9560 shows that the artificial neural network model correctly identified approximat ely 95.60% of creditworthy applicants.

An F1 score of 0.8833 indicates that the support vector machine model achieves a very good bala nce between precision and recall.

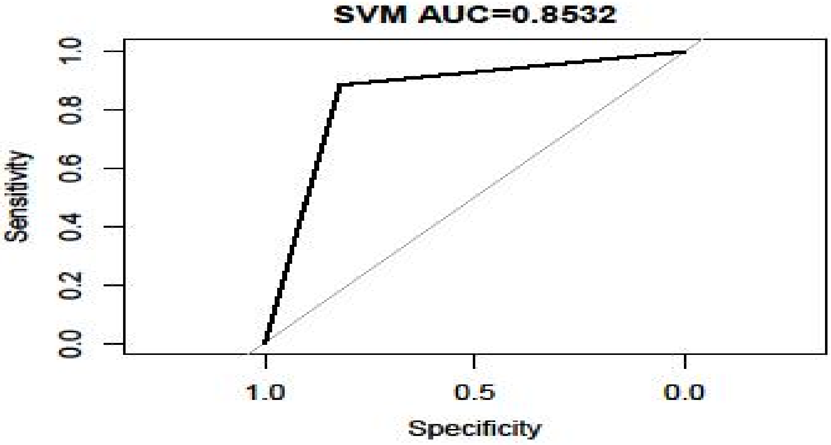


Figure 4.6: Support Vector Machine ROC

4.7 Decision Tree

4.7.1 Model Building

A decision tree was employed on the same dataset that was used in in training and testing phases of linear discriminant analysis, logistic regression, artificial neural network, and support vector machine on the same partition size. *rpart, rpart.plot, caret* and *pROC* packages of R statistical software were used. **Appendix B9** displays the R Code for the Decision Tree.

4.7.2 Results

Table 4.14: Decision Tree Confusion Matrix:

	Bad	Good
Bad	94	21
Good	10	22

Table 4.15: Decision Tree Performance Metrics Scores:

AUC	0.7524
Precision	0.8174
Accuracy	0.7891
Recall	0.9038
F1 Score	0.8584

The AUC (Area Under the ROC curve) value of 0.7524 represents the discriminative power of the Decision tree technique. A higher AUC means better discrimination between the positive and negative outcomes.

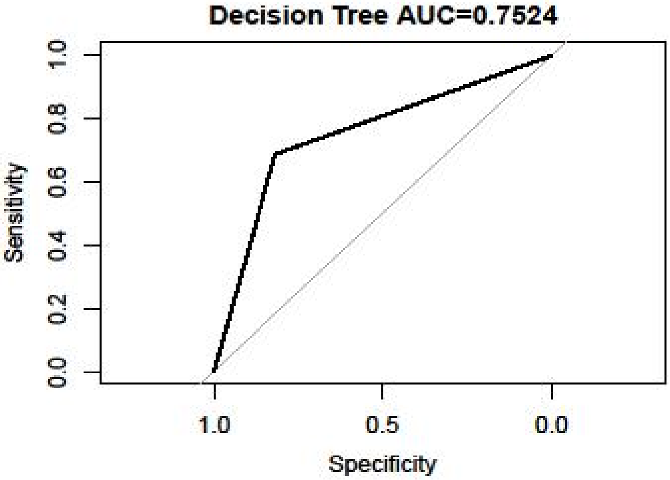
The precision of 0.8174 means the decision tree technique correctly identified approximately 81.74% of the applicants that are actually creditworthy as creditworthy.

Accuracy: An accuracy of 0.7891 means that the model classified approximately 78.91% of all credit applicants correctly.

A recall of 0.9038 points out that the model correctly identified approximately 90.38% of creditworthy applicants.

An F1 score of 0.8584 shows that the model achieves a good balance between precision and recall.

Figure 4.7: Decision Tree ROC



4.8 *k*-Nearest Neighbors

4.8.1 Model Building

A *k*-nearest neighbour was employed on the same dataset that was used in linear discriminant analysis, logistic regression, artificial neural network, support vector machine and decision tree on the same partition size. The *e1071, caTools, Class, tidyverse, caret* and *pROC* packages of R statistical software were used. The size of the *k*-hyperparameter is determined by evaluating the model on different values of *k* until the model with the value of *k* that gives the highest accuracy is obtained. **Appendix B9** displays the R Code for the *k* -nearest neighbours.

4.8.2 Results

Table 4.16: Determining *k*-Hyperparameter

k-Hyperparameter	Accuracy
k = 1	0.9492
k = 2	0.9322
k = 6	0.9492
k = 7	0.9576
k = 10	0.9746
k = 11	0.9661
k = 12	0.9661
k = 13	0.9576
k = 14	0.9492

The above table 4.16 shows the different accuracy results of the *k*-nearest neighbour under the different *k*-hyperparameters. The *k*-nearest neighbour with *k*=10 shows a slightly high accuracy score of 0.9746 compared to other *k* values. Thus, *k*=10 is used.

Table 4.17: *k*-Nearest Neighbors Confusion Matrix:

	Bad	Good
Bad	87	0
Good	03	28

Table 4.18: *k*-Nearest Neighbors Performance Metrics Score:

AUC	0.9833
Precision	1.00
Accuracy	0.9746
Recall	0.9667
F1 Score	0.9831

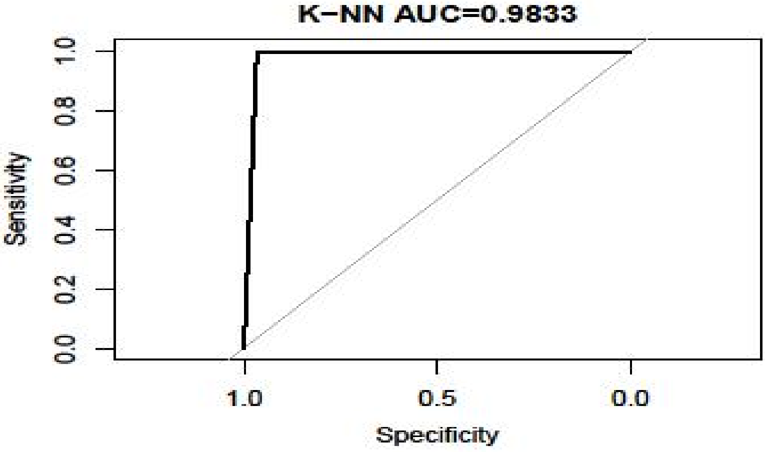
AUC (Area Under the Curve): A value of 0.9833 for AUC indicates that the *k*-Nearest Neighbors technique has very excellent discriminative power, with a high probability of ranking a randomly chosen positive instance higher than a randomly chosen negative instance.

Precision: The *k*-Nearest Neighbors Performance model achieved a perfect precision of 1.00, suggesting that the technique accurately detected 100% of the creditworthy credit applicants.
Accuracy: an accuracy of 0.9746 means that the *k*-Nearest Neighbors Performance classified app roximately 97.46% of all credit applicants correctly.

A recall of 0.9667 shows that the *k*-Nearest Neighbors Performance correctly identified approximately 96.67% of creditworthy applicants.

An F1 score of 0.9831 shows that the model achieves an excellent balance between precision and recall.

Figure 4.8: *k*-Nearest Neighbor ROC



4.9 Model Performance Assessment

Based on the results obtained and summarised in Table 4.19 below, we assess the performance of linear discriminant analysis, logistic regression, artificial neural network, support vector machine, decision tree and *k*-nearest neighbours in credit classification.

Table 4.19: Models Performance Assessment Summary

Model	AUC	Precision	Accuracy	Recall	F1
Linear Discriminant Analysis	0.8494	0.1389	0.8582	0.6579	0.7247
Logistic Regression	0.8537	0.1495	0.1600	0.5333	0.2335
Artificial Neural Network	0.9062	0.8065	0.8582	0.6579	0.7247
Support Vector Machine	0.8532	0.8208	0.8369	0.9560	0.8833
Decision Tree	0.7524	0.8174	0.7891	0.9038	0.8584
k-Nearest Neighbors	0.9833	1	0.9746	0.9667	0.9831

Overall, the performance of the models can be ranked as follows:

1. K-nearest neighbour: The *k-nearest* Neighbor technique outperformed all the techniques, demonstrating outstanding performance on all the performance metrics. The perfect precision indicates its ability to perfectly identify creditworthy applicants among the applicants predicted as creditworthy. Additionally, the *k-nearest* Neighbor achieved the highest AUC value, indicating superior discriminative performance.

2. Artificial neural network: The artificial neural network model also shows outstanding performance second to *k*-nearest neighbours, demonstrating high precision in comparison with other techniques (SVM, CART, LR and LDA), indicating a strong ability to correctly identify positive instances. Additionally, it achieved the highest AUC value, indicating superior discriminative performance.

3. Logistic regression and Support vector machine: The logistic regression and support vector machine show a good AUC value suggesting a good level of discrimination in both techniques. However, on the other metrics (Precision, Accuracy, Recall and F1 score) Support vector machine records good performance while Logistic regression records low scores on all the metrics, suggesting its inability to detect a potential creditworthy applicant.

4. Linear Discriminant Analysis: The Linear discriminant analysis model has good precision, indicating a strong ability to correctly identify creditworthy applicants. The AUC score showed a good level of discrimination.

5. Decision tree: The decision tree technique has good discriminative ability, precision and accuracy. However, it record the lowest AUC score compared to the other techniques.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATION

5.1Summary

This study assesses the efficacy of six machine-learning techniques (linear discriminant analysis, logistic regression, artificial neural networks, support vector machines, decision tree and *k*-nearest neighbour) in predicting the creditworthiness of credit clients and identifies the variable contributing to default in microcredit. The model was fitted. Based on the outcomes of the performance metrics, the k-nearest neighbour emerged the best, followed by an artificial neural network. The most contributing variables to default in microcredit loans were found to be the Amount Approved, Amount Applied, and Return over investment per annum (ROI PA), Garson's measure of the relative importance of the independent variables.

5.2 Conclusion

This research assesses the performance of six credit scoring methodologies (linear discriminant analysis, logistic regression, artificial neural networks, support vector machines, decision tree and *k*-nearest neighbour) in classifying credit applicants into their correct classes based on the 660 credit client data.

Based on the results obtained, the *k*-nearest neighbour and artificial neural network techniques emerged as the best model with outstanding discriminative performance (outstanding high accuracy, precision and excellent discriminative ability), followed by logistic regression, support vector machine, linear discriminant analysis and decision tree. Although, all techniques achieve significantly good accuracy except logistic regression.

The advanced techniques show more robustness to the non-normality and imbalanced nature of the credit data over the traditional methods. However, the complex structures of modern techniques make it challenging to understand individual variable contributions to model prediction.

The top three most contributing variables to default and delinquency according to Garson's measure of the relative importance of the independent variables are Amount Approved, Amount Applied, and Return over investment per annum (ROI PA).

Overall, *k*-nearest neighbour and artificial neural networks are robust and effective in classifying instances and have the potential to make accurate predictions on unseen data. Hence, the model has the ability to generalise its results.

5.3 Recommendation

5.3.1 Banks and Microcredit Institutions

- Institutions should work to ensure improved data collection and administration procedures are positioned. This provides a solid foundation for improved model performance and risk management.
- Integrate statistical modelling credit scoring prediction and classification and other methods.

5.3.2 Area for Further Research

- Further studies should use k-nearest neighbour or artificial neural network as their credit evaluation model, which is most suitable to the non-linear nature of the credit data. For generalisation and accuracy of the credit scoring model's result, it's recommended to use larger data and more variables.
- The evolution of a borrower's credit quality (how can a client gradually change from a good credit quality to a poor credit quality), the evolution mechanism, and the performance status of a borrower's credit quality at various stages are good research questions recommended for future studies.

REFERENCES

Abdelmoula, K. A., (2015). "Bank credit risk analysis with K-nearest neighbour classifier: cases of Tunisian Banks". *Journal of Accounting and Management Information Systems, Vol. 14, No. 1, PP, 79-104.*

Abdou, Hussein. A. H., and Pointon, J. (2011). "Credit scoring, statistical techniques and evaluation criteria: A review of the literature". *Intelligent Systems in Accounting, Finance & Management*, *18*(2–3), 59–88.

Abdou, H., Delamaire, L. and, Pointon, J. (2009). "Credit card fraud and detection techniques: a review". *Banks and Bank Systems, 4 (2). pp. 57-68.* http://eprints.hud.ac.uk/id/eprint/19069/.

Abellan, J., and Castellano, J. G. (2017). "A comparative study on base classifiers in ensemble methods for credit scoring". *Expert Systems with Applications, Vol. 19, 1-10.*

Aizerman, Mark A.; Braverman, Emmanuel M. & Rozonoer, Lev I. (1964). "Theoretical foundations of the potential function method in pattern recognition learning". Automation and Remote Control. 25: 821–837.

Ala'raj M., and Abbod M. F. (2015). "Classifiers consensus system approach for credit scoring" *Knowledge-Based System Journal. Volume 104, Pages 89-105 doi: https://doi.org/10.1016/j.knosys.2016.04.013*

Angelini, E., di Tollo, G., and Roli, A. (2008). "A neural network approach for credit risk evaluation". *The Quarterly Review of Economics and Finance, Elsevier, 48*(4), 733–755.

Antonio B., Rafael P., Juan L., and Salvador R. (2013) "Credit Scoring Models for MFI using Neural Networks: Evidence from Peru", *Expert Systems with Applications, Vol.40, pp 356-364.*

Baesens, B., Gestel, T. V., Viaene, S., Stepanova, M., Suykens, J., and Vanthienen, J. (2003). "Benchmarking state-of-the-art classification algorithms for credit scoring". *Journal of the Operational Research Society, Vol. 54*, 627–635.

Baesens, B., Setiono, R., Mues, C., and Vanthienen, J. (2003). "Using Neural Network Rule Extraction and Decision Tables for Credit-Risk Evaluation". *Journal of Operational research and management science*, *Vol. 56, 312-329.*

Bank for International Settlement (2010). "Basel Committee on Banking Supervision Publication (BCBS Publication)" https://www.bis.org/bcbs/publications.htm

Barfield J., Poulsen, J. and French, A. (2004)," Discriminant Function Analysis (DA)", Retrieved August 18, 2014, from: http://userwww.sfsu.edu/efc/classes/biol710/discrim/discriminant.htm

Bellotti, T., and Crook, J., (2009). "Support Vector Machine for Credit Scoring and Discovery of Significant Features". *Journal of Expert System with Application, March 1, 2009. Vol. 36, Issue 2, Part 2, pp 3302-3308. doi: https://doi.org/10.1016/j.eswa.2008.01.005*

Bensic, M., Sarlija, N., and Zekic-Susac, M. (2005). "Modelling Small - Business Credit Scoring by Using Logistic Regression, Neural Networks and Decision Trees". *Intelligent Systems in Accounting, Finance and Management, 133–150.*

Boser, B.E., Guyon, I.M. and Vapnik, V.N. (1992) "A Training Algorithm for Optimal Margin Classifiers". *Proceedings of the 5th Annual Workshop on Computational Learning Theory (COLT'92), Pittsburgh, 27-29 July 1992, 144-152.*

Coffman, J. (1986). The proper role of tree analysis in forecasting the risk behaviour of borrowers. *Journal of Management Decision Systems, Atlanta, MDS Reports, 3* (4), 7.

Christopher E. and, Abba S. G. (2020): "Classification Performance for Credit Scoring using Neural Network", *an International Journal of Emerging Trends in Engineering Research. Volume 8. No. 5, May 2020*. DOI: https://doi.org/10.30534/ijeter/2020/19852020

Davis, R., Edelman, D., and Gammerman, A. (1992). "Machine learning algorithms for credit card applications ". *IMA Journal of Mathematics Applied in Business and Industry, 43-51.*

Desai, V. S., Crook, J. N., and Overstreet Jr, G. A. (1996). "A comparison of the neural networks and linear scoring models in the credit union environment". *European Journal of Operational Research, 95* (1), 24–37.

Earky. M. D. (1977). "Warning of Blank Failure: A Logit Regression Approach". *Journal of Banking and Finance*, 249-276.

Eisenbeis, R.A. (1978). "Problems in applying discriminant analysis in credit scoring models". *Journal* of banking and finance Volume 2, Issue 3, Pages 205-219. https://doi.org/10.1016/0378-4266(78)90012-2.

Fabrigar, L.R., Wegener, D.T., MacCallum, R.C. and Strahan, E.J. (1999). "Evaluating the use of exploratory factor analysis in psychological research". *Psychological Methods, Vol. 4, No. 3, pp. 272-299.*

Fensterstock, F. (2005). "Credit Scoring and the Next Step". *Journal of Business Credit*, 107(3): 46-49. New York: National Association of Credit Management.

Finlay, S. (2012). “Instance sampling in credit scoring: An empirical study of sample size and balancing”. *International Journal of Forecasting*, 224-238.

Fisher, R.A. (1936). “The Use of Multiple Measurements in Taxonomic Problems”. *Annals of Eugenics, 7,179-188*. http://dx.doi.org/10.1111/j.1469-1809.1936.tb02137.x

Fuente Fernández, S. (2011). “Análisis de Correspondencias Simples y Múltiples”. *Económicas y Empres., pp. 58–59.*

Galindo, J., and Tamayo, P. (2000). “Credit risk assessment using statistical and machine learning: basic methodology and risk modelling applications”. *Computational Economics, 107–143.*

Garson, G.D. (1991). “Interpreting neural network connection weights”. *Artificial Intelligence Expert*, Vol.6, No.4, pp.46-51.

Germanno T., Joel J., Rodrigues C., Ricardo A. L., Rabelo S. A., and Kozlov A., (2020) “Artificial neural network and Bayesian network models for credit risk prediction”. *Journal of Artificial Intelligence and Systems, Vol 2, 118–132. https://doi.org/10.33969/AIS.2020.21008 https://iecscience.org/journals/AIS*

Ghodselahi A. (2011). “A Hybrid Support Vector Machine Ensemble Model for Credit Scoring”. *International Journal of Computer Applications. 17, 5, 1-5. DOI: https://doi.org/10.5120/2220-2829*

Gonçalves. E. B. and, Gouvêa. M. A(2021). “Credit Risk Analysis Applying Logistic Regression, Neural Networks and Genetic Algorithms Models”. *An International Journal of Advanced Engineering Research and Science (IJAERS) Peer-Reviewed Journal Vol-8, Issue-9; Sep 2021. DOI: https://dx.doi.org/10.22161/ijaers.89.20*

Guo, Y., (2020). “Credit Risk Assessment of P2P Lending Platform towards Big Data based on BP Neural Network”. J. Vis. Commun. Image. Represent. 71, 102730.

Grablowsky, J. B. (1975). A Behavioral Risk in Consumer Credit. *The Journal of Finance, 30(3), 915-916. http://dx.doi.org/10.2307/2326880*

Greene, W.H. (1993): *Econometric Analysis*. Englewood Cli_s, NJ: Prentice Hall, 801 pages.

Han, J. and Kamber, M. (2006). *Data Mining: Concepts and Techniques* (2nd ed.). San Franscisco: Elsevier, 770 pages.

Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome (2001). "The Elements of Statistical Learning". *Springer. pp. 269–272. ISBN 0-387-95284-5.*

Henley, W. E. (1995). *Statistical aspects of credit scoring* (PhD thesis). UK: Open University.

Henley, W., and Hand, D. (1996). "A k-nearest neighbour classifier for assessing consumer credit risk". *The Statistician, 45* (1), 77–95.

Henley, W. E., and Hand, D. J. (1997). "Construction of a k-nearest neighbour credit-scoring system". *IMA Journal of Mathematics Applied in Business and Industry, 8*, 305–321.

Hosmer, D., and Lemeshow, S. (1989). *Applied Logistic Regression.* New York: John Wiley & Sons, Inc.

Hosmer Jr, D.W., Lemeshow, S. and Sturdivant, R. X. (2000). *Model-building strategies and methods for logistic regression.* Applied Logistic Regression, Third Edition, pp. 89-151.

Huang, C. L., Chen, M. C., and Wang, C. J. (2007). "Credit scoring with a data mining approach based on support vector machines". *Expert System with Applications, 847-856*

Jagric, T. and Jagric, V. (2011). "A Comparison of Growing Cell Structures Neural Networks and Linear Scoring Models in the Retail Credit Environment", *Eastern European Economics, vol. 49, no. 6, pg. 74-96*

Jaimes, F., Farbiarz, J., Alvarez. and Martinez, D. (2005). "Comparison between logistic regression and neural networks to predict death in patients with suspected sepsis in the emergency room". *Critical care, Vol. 9, No. 2, pp. 150-156.*

Jha, G.K. (2007). *Artificial neural networks and its application*. IARI, NewDehli, girish_iasri@rediffmail.com, pp. 1-10.

Khashman, A. (2010). "Neural networks for credit risk evaluation: Investigation of different neural models and learning schemes". *Expert System with Application*, 6233-6239.

Khemakhem, S. and Boujelbène Y.(2015): "Credit risk prediction: A comparative study between discriminant analysis and the neural network approach". *Journal of Accounting and Management Information Systems Vol. 14, No. 1, pp. 60-78, 2015*

Koh, H.C. & Tan, W.C. and Goh, C.P. (2006): "A Two-step Method to construct Credit Scoring Models with Data Mining Techniques". *International Journal of Business and Information.*

Lahsasna, A., Ainon, R.N., and Wah, Y. T.,(2010): "Credit Scoring Models Using Soft Computing Methods: A Survey". *International Arab Journal of Information Technology 7(2):115-123*

Lee, T., Chui, C. Lu, and I. Chen. (2002). "Credit Scoring Using the Hybrid Neural Discriminant Technique". *Expert Systems with Applications, 23*, 245-254.

Lessmann, S., Baesens, B., Seow, H.-V. , and Thomas, L. C. (2015). "Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research". *European Journal of Operational Research, 247*, 124–136.

Li, J., Liu, J., Xu, W., and Shi, Y.,(2004). "Support Vector Machines Approach to Credit Assessment" *computational Science-ICCS 2004, 4th International Conference, Krakow, Polan, Proceedings, Part IV.* http://dx.doi.org/10.1007/978-3-540-25944-2_115

Liang, D., Tsai, C.F., and Wu, H.T., (2015). "The effect of feature selection on financial distress prediction". *Knowledge Based System. 73, 289–297.*

Liang, Qi (2005). *Research on Credit Risk Measurement of Commercial Bank.* Beijing: China Financial Publishing House.

Lubis H., Sirait P., and Halim A., (2021). "Knn Method on Credit Risk Classification With Binary Particle Swarm Optimization Based Feature Selection" *JURNAL INFOKUM, Volume 9, No. 2.* http://infor.seaninstitute.org/index.php/infokum/index

Makowski, P. (1985). Credit scoring branches out. *Credit World, 75* (1), 30–37.

Malhotra, R. And Malhotra D.K., (2003). "Evaluating Consumer Loans Using Neural Networks", *The international journal of management science. OMEGA 31 (2003) 83-96.*

Martin, D., (1977). "Early Warning of Bank Failure: A Logit Regression Approach". *Journal of Banking and Finance 249-276.*

Mallouh, A. A., Qawaqneh, Z., and Barkana, D.B., (2018). "New transformed features generated by deep bottleneck extractor and a GMM-UBM classifier for speaker age and gender classification". *Neural Computation Application 30, 2581–2593.*

May, R., Dandy, G. and Maier, H. (2011). "Review of Input Variable Selection Methods for Artificial Neural Networks". *Artificial Neural Networks- Methodological Advances and Biomedical Applications, Prof. Kenji Suzuki (Ed.), InTech,* DOI: 10.5772/16004. [online] Available from: http://www.intechopen.com/books/artificial-neural-networks-methodological-advances-and-biomedical/-applications-review-of-input-variable-selection-methods/-for-artificial-neural-networks, [4 February 2017].

McCulloch, Warren S.; Pitts, Walter (1943-12-01). "A logical calculus of the ideas immanent in nervous activity". *The Bulletin of Mathematical Biophysics. 5 (4): 115–133.* *https://doi.org/10.1007/BF02478259 ISSN 1522-9602.*

Mendoza, M., Rafael V., Dorantes C., Joel, E,. Monroy, C,. José, and Jasso Arriaga, Xóchitl. (2017). “The Statistical method of discriminant analysis as an interpretation tool of the mobile addiction study”, *Ibero-American Journal for the Educational Research and Development 7(14), 222-247. https://doi.org/10.23913/ride.v7i14.282*

Mester, L. (1997). *What's the Point of Credit Scoring?* Federal Reserve Bank of Philadelphia Business Review, 3-16.

O'Brien, R. (2007). “A caution regarding the rules of thumb for variance inflation factors”. *Quality and Quantity, Vol. 41, No 5, pp. 673-690.*

Oreski, S., Oreski, D. and, Oreski, G. (2012). “Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment”. *Expert Systems with Applications* 39(16):12605–12617. DOI:10.1016/j.eswa.2012.05.023.

Pagès, J. (2004). “Analyse Factorielle De Données Mixtes: Principe Et Exemple D’application”. *Laboratoire de mathématiques appliquées.*

Park, H.A. (2013) An Introduction to Logistic Regression: From Basic Concepts to Interpretation with Particular Attention to Nursing Domain. *Journal of Korean Academy of Nursing. 43(2):154-164 DOI:. https://doi.org/10.4040/jkan.2013.43.2.154.*

Pławiak, P., Abdar, M., Acharya, U.R., (2019). “Application of new deep genetic cascade ensemble of SVM classifiers to predict the Australian credit scoring”. *Application Software Computation. 84, 105740.*

Reichert, A. K., Cho, C. C. and, Wagner, M. G. (1983). “An Examination of the Conceptual Issues Involved in Developing Credit-Scoring Models”. *Journal of Business & Economic Statistics Pages 101-114 Volume 1, 1983 - Issue 2.*

Rojas, R. (1996). *Neural Networks: A Systematic Introduction*. Springer Verlag Berlin, 509 pages.

Schebesch, K. B., and Stecking, R. (2005). “Support vector machines for classifying and describing credit applicants: detecting typical and critical regions” *Journal of the Operational Research Society June 2005, 56(9):1082-1088. DOI: https://doi.org/10.1057/palgrave.jors.2602023*

Song, J.H., Venkatesh, S.S., Conant, E.A., Arger, P.H. and Sehgal, C.M. (2005). "Comparative analysis of logistic regression and artificial neural network for computer-aided diagnosis of breast masses". *Academic radiology, Vol. 12, No. 4, pp. 487-495.*

Stoltzfus, J.C. (2011). "Logistic regression: a brief primer". *Academic Emergency Medicine, Vol. 18, No. 10, pp. 1099-1104*.

Suleiman, S. Issa, Suleiman, U. Usman *(2014). "*Predicting an Applicant Status Using Principal Component, Discriminant and Logistic Regression Analysis". 54 *International Journal of Mathematics and Statistics Invention (IJMSI), 2(10), pp.05-15.*

Srinivasan, V., & Kim, Y. H. (1987). "Credit granting: A comparative analysis of classification procedures". *The Journal of Finance, 42* (3), 665–681.

Tabachnick, B.G., and Fidell, L. S. (1996), "*Using multivariate statistics*", (3rd ed.), New York: Harper Collins publishers.

Tap, B. W., Ong. And, Husain. S. (2011). "Using data mining to improve assessment of credit worthiness via credit scoring models". *Expert Systems with Applications*, 13274-13283.

Tam, K. Y., and Kiang, M. Y. (1992). "Managerial applications of neural networks: The case of bank failure predictions". *Journal of Management Science, 38* (7), 926–947.

Tekić D, Mutavdžić B , Milić D, Novković N , Zekić V , and Novaković T(2021). "Credit risk assessment of agricultural enterprises in the republic of Serbia: logistic regression vs discriminant analysis". *Journal of Economics of Agriculture, Year 68, No. 4, 2021, (pp. 881-894),* Belgrade. doi: https://doi.org/10.5937/ekoPolj2104881T

Tucker, J. (1996). Neural networks versus logistic regression in financial modelling:

A methodological comparison in Proceedings of the 1996 World First Online Workshop on Soft Computing (WSC1).

United Nations Sustainable Development goals, goal 10. https://sdgs.un.org/goals/goal10

Vapnik, V. and Cortes, C. (1995) "Support-Vector Networks". *Machine Learning, 20, 273-297.*

http://dx.doi.org/10.1007/BF00994018

Veall, M. and Zimmermann, K.(1996). "Evaluating Pseudo-R2's for Binary Probit Models". *Quality & Quantity: International Journal of Methodology, Springer, vol. 28(2), pages 151-164, May. DOI: https://doi.org/10.1007/BF01102759*

West, D. (2000). "Neural network credit scoring models". *Computers & Operations Research, 27* (11–12), 1131–1152.

Wu, J., Tennyson, R.D. and, Hsia, T. (2010). "A study of student satisfaction in a blended e-learning system environment". *Journal of Computers & Education 55(1):155-164 DOI:10.1016/j.compedu.2009.12.012*.

WU Shinong. (2003). *Research on the Share Market Risk in China Share Market.* Beijing: China Renmin University Press.

Xia, Y., Liu, C., Li, Y., and Liu, N., (2017). "A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring". *Expert System Application. 78, 225–241.*

Xu, X., Saric , Z., & Kouhpanejade, A. (2014). "Freeway Incident Frequency Analysis Based on CART Method". *Promet - Traffic & Transportation, 191 - 199.*

Yap, B., Ong, S., and Husain, N. (2011). "Using data mining to improve assessment of credit worthiness via credit scoring models". *Expert Systems with Application*

Yobas, M. B., Crook, J. N., and Ross, P. (2000). "Credit scoring using neural and evolutionary techniques". *IMA Journal of Mathematics Applied in Business and Industry, 11*, 111–125.

Zhang, S., Zhenyun D., Ming Z., and, Xuelian D (2018). "A novel kNN algorithm with data-driven k parameter computation" *Pattern Recognition Letters 109 (2018) 44–54.* https://doi.org/10.1016/j.patrec.2017.09.036 0167-8655/

Zhao, Z., Xu, S., Kang, B. H., Kabir, M. M., and Liu, Y. (2015). "Investigation and improvement of multi-layer perceptron neural". *Expert Systems with Applications, 42, 3508– 3516.*

Zhao, Z., Xu, S., Kang, B. H., Kabir, J., Liu, Y., and Wasinger, R. (2015). "Investigation and improvement of multi-layer perceptron neural networks for credit scoring". *Expert Systems with Applications, 3508-3516.*

Zhirov, V. K, Staroverova, N.A, Shustrova M.L and, Tomilova M. N.(2021). "Neural network as a tool to solve the problem of credit scoring". *Journal of Physics: Conference Series(International Conference on IT in Business and Industry) 2032 (2021) 012120 IOP Publishing doi:* https://doi.org/10.1088/1742-6596/2032/1/012120

Zhou H., Wang J., Wu J., Zhang L., Lei P., and Chen X., (2013). "Application of the Hybrid SVM-KNN Model for Credit Scoring". *2013 Ninth International Conference on Computational Intelligence and Security* doi: https://doi.org/10.1109/CIS.2013.43

Zhu Y, Xie C, Sun B, Wang G and Yan X(2016). “Predicting China’s SME Credit Risk in Supply Chain Financing by Logistic Regression, Artificial Neural Network and Hybrid Models”. *Journal of Sustainability 2016, 8, 433; doi: https://doi.org/10.3390/su8050433*

APPENDIX A

Graph and Figures
EXPLORATORY DATA ANALYSIS PLOTS

A1: Univariate Exploratory Data Analysis plots

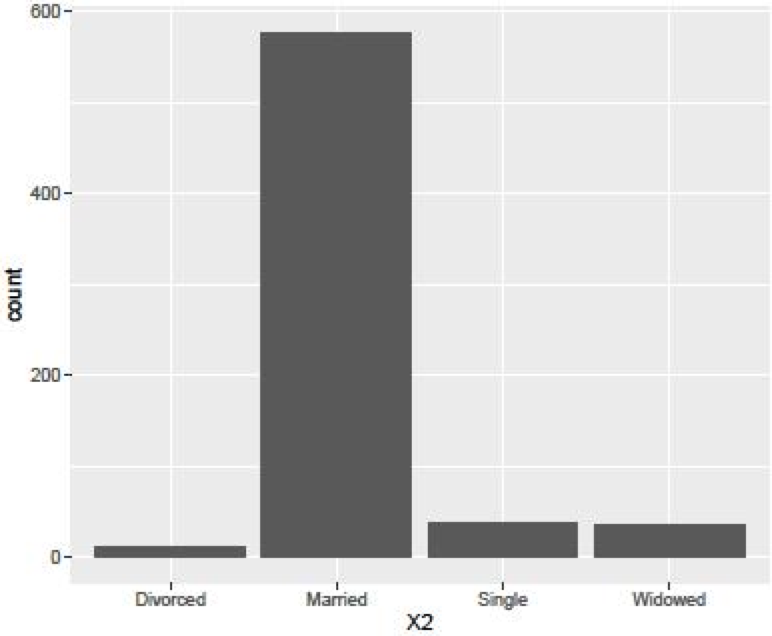


Figure 6.1: Categorical data distribution by Marital Status

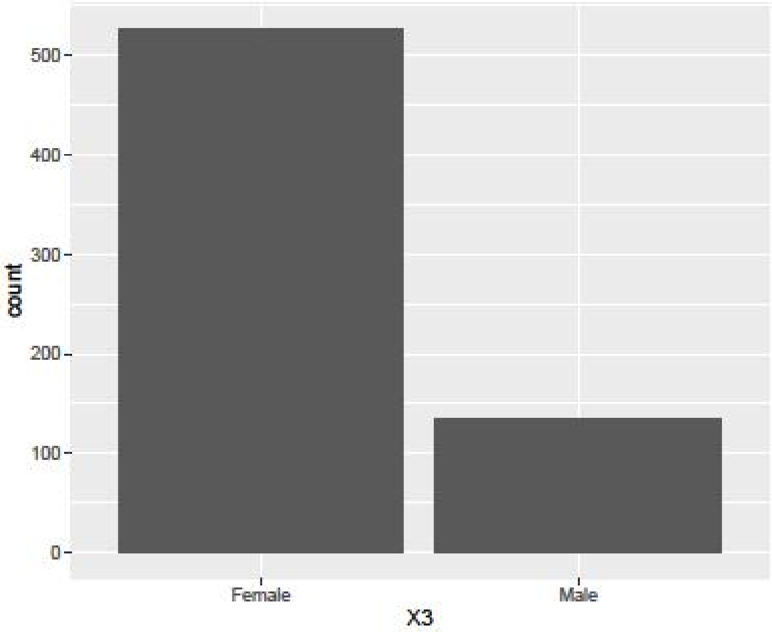


Figure 6.2: Categorical data distribution by sex (Gender)

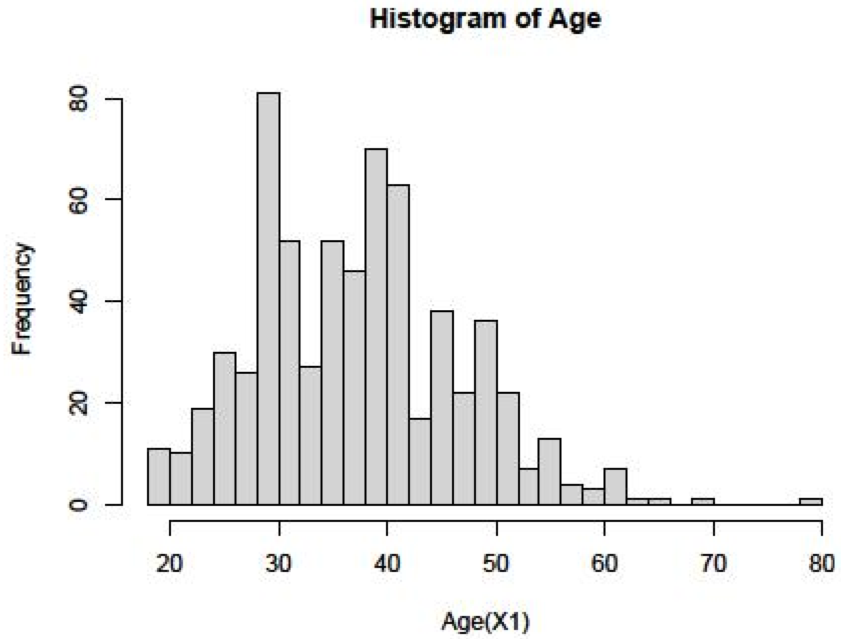


Figure 6.3: Data Distribution by Age

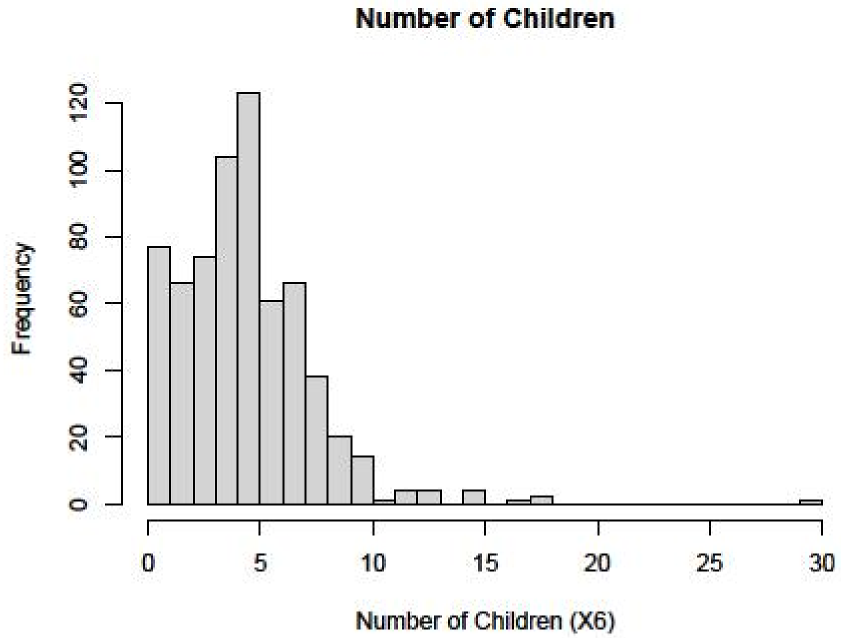


Figure 6.4: Data distribution by number of children

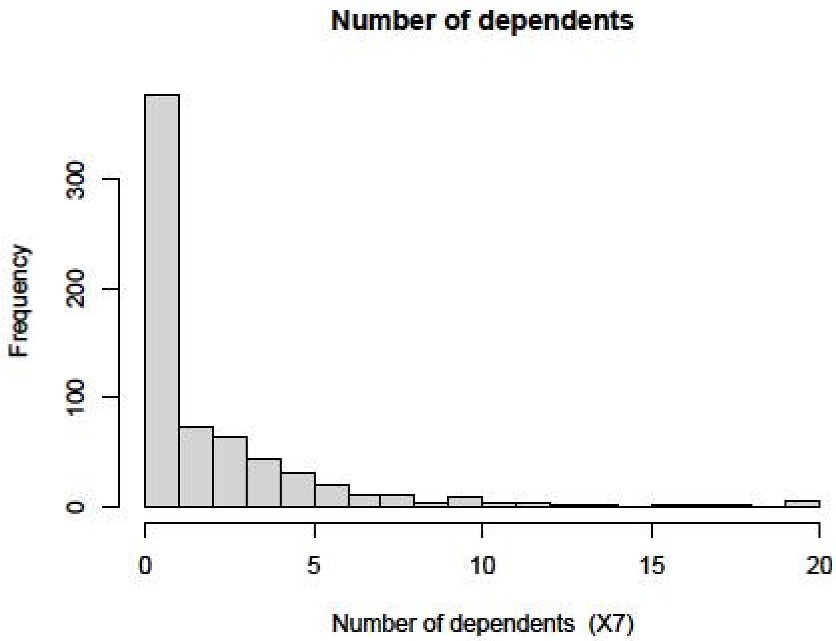


Figure 6.5: Data distribution by number of dependent

A2: Bivariate Exploratory Data Analysis Plots

Table 6.0: Correlation Matrix of the Dataset

	Age	Exp.in Buss.	Amnt. Applied	No. of Child.	No. of Dependt	Amnt. Approved	ROI PA	Cap. Assessed
Age		0.39	-0.06	0.46	0.14	-0.03	-0.08	
Exp.in Buss.	1.00	1.00	-0.01	0.29	0.17	-0.01	-0.01	-0.04
Amnt. Applied	0.39	-0.01	1.00	-0.08	0.10	0.41	-0.03	0.05
No. of Child.	-0.06	0.29	-0.08	1.00	0.42	-0.02	0.01	0.49
No. of Dependt	0.46	0.17	0.10	0.42	1.00	0.14	-0.04	-0.02
Amnt. Approved	0.14	-0.01	0.41	-0.02\	0.14	1.00	-0.05	0.24
ROI PA	-0.03	-0.01	-0.03	0.01	-0.04	-0.05	1.00	0.50
Cap. Assessed	-0.08	0.05	0.49	-0.02	0.24	0.50	-0.08	-0.08
	-0.04							1.00

A3: Multivariate Exploratory Data Analysis Plots

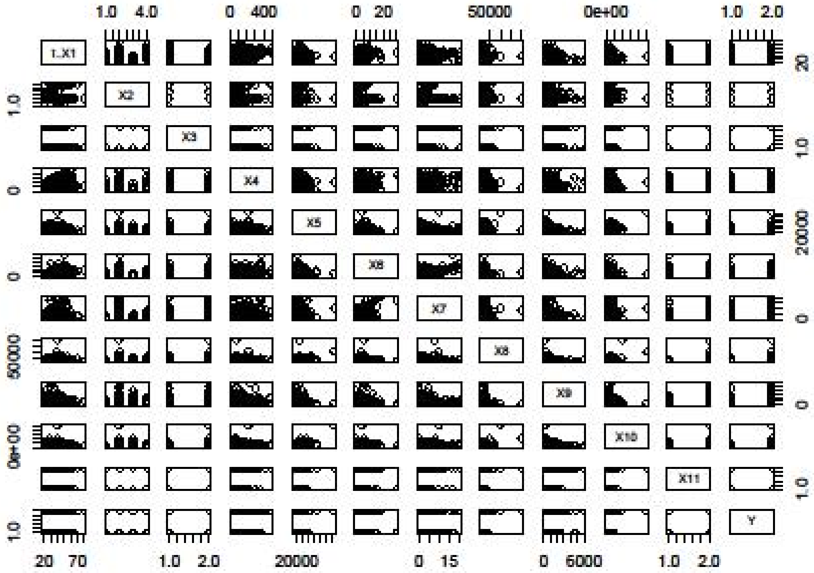


Figure 6.6: Multivariate plot of the dataset

A4: Artificial Neural Network Structure and ROC plots

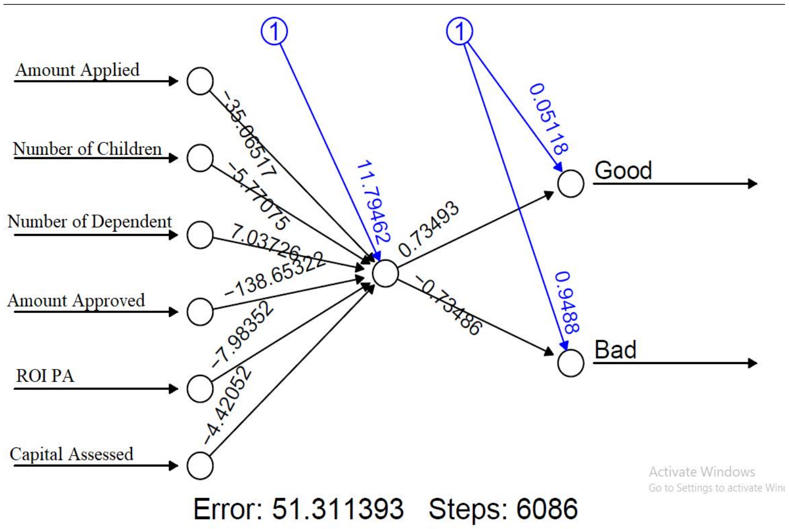


Figure 6.8: Artificial Neural Network model Structure with one hidden node

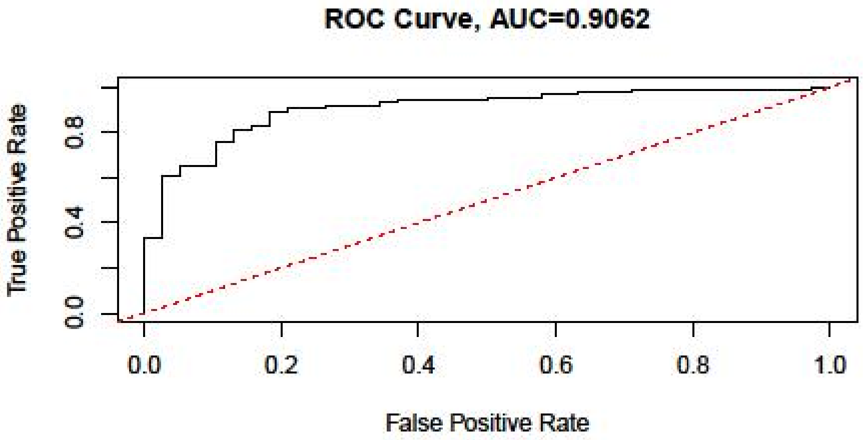


Figure 6.9: Artificial Neural Network model ROC curve with one hidden node and an AUC=0.9062

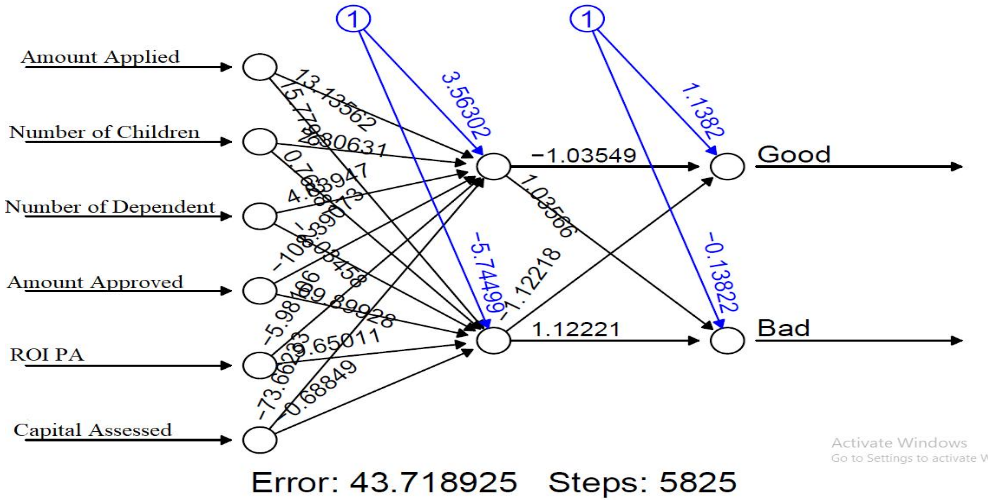


Figure 6.10: Artificial Neural Network model Structure with two hidden nodes

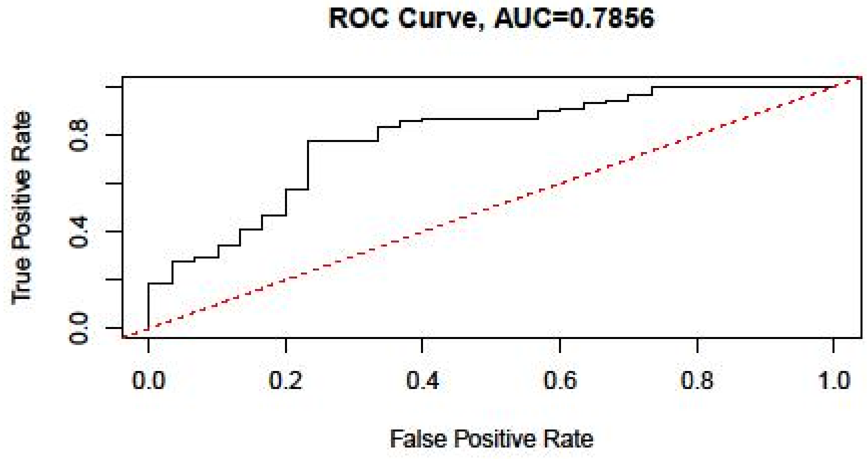


Figure 6.11: Artificial Neural Network model ROC curve with one hidden node and an AUC=0.7856

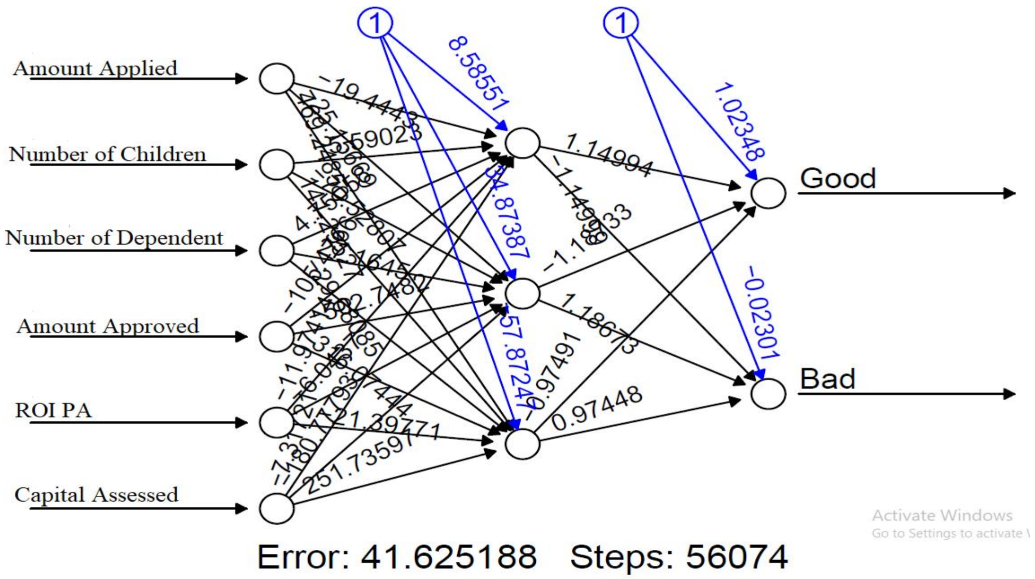


Figure 6.12: Artificial Neural Network model Structure with Three hidden nodes

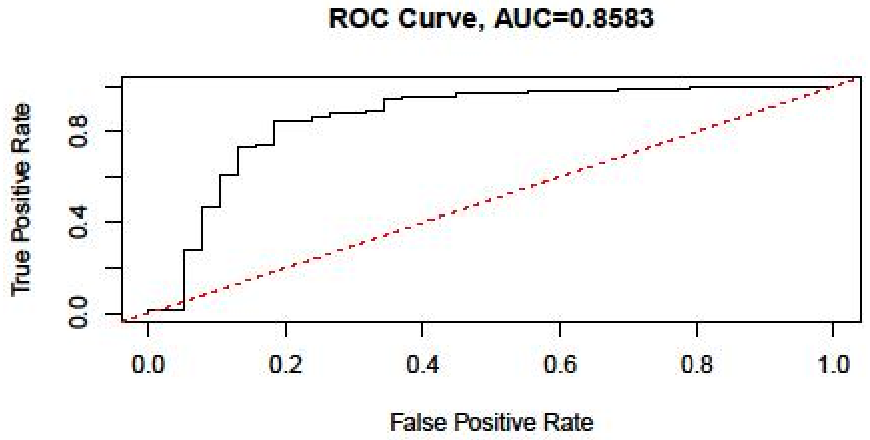


Figure 6.13: Artificial Neural Network model ROC curve with one hidden node and an AUC=0.8583

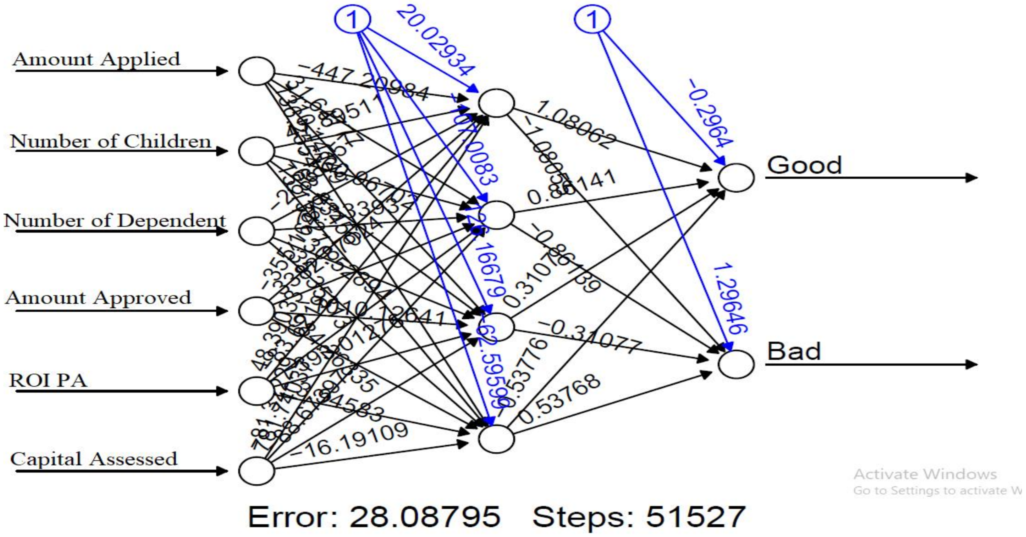
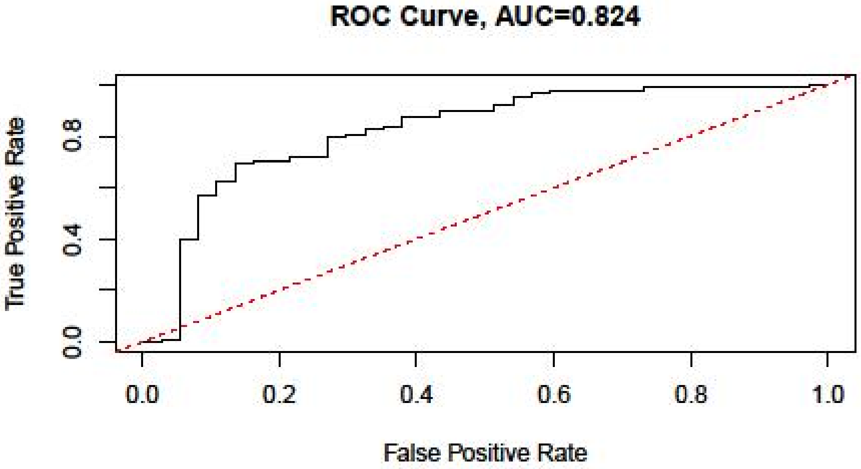


Figure 6.14: Artificial Neural Network model Structure with Four hidden



nodes

Figure 6.15: Artificial Neural Network model ROC curve with one hidden node and an AUC=0.824

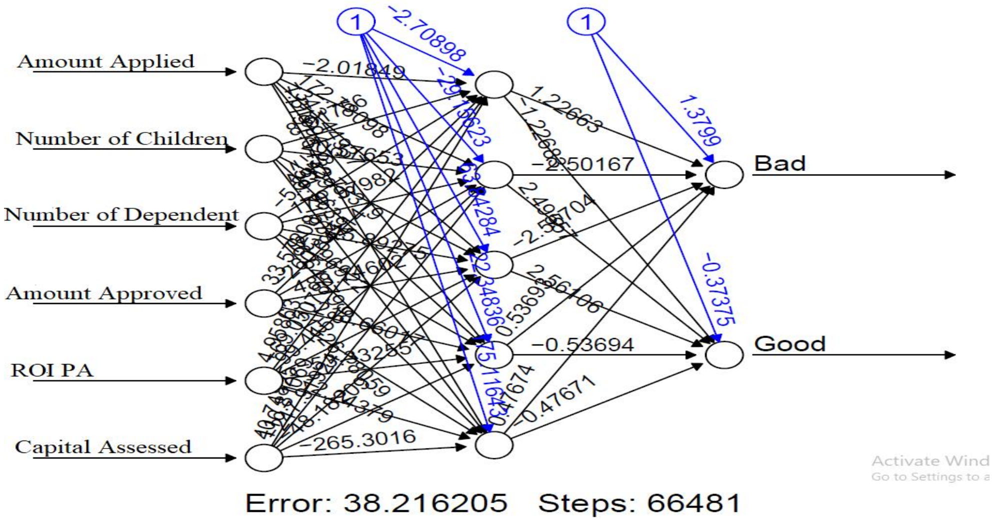


Figure 6.16: Artificial Neural Network model Structure with Five hidden nodes

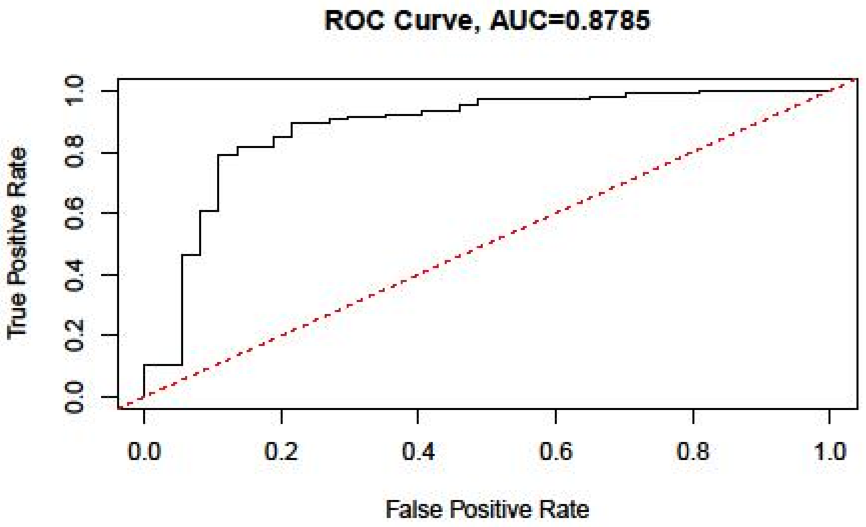


Figure 6.17: Artificial Neural Network model ROC curve with one hidden node and an AUC=0.8785

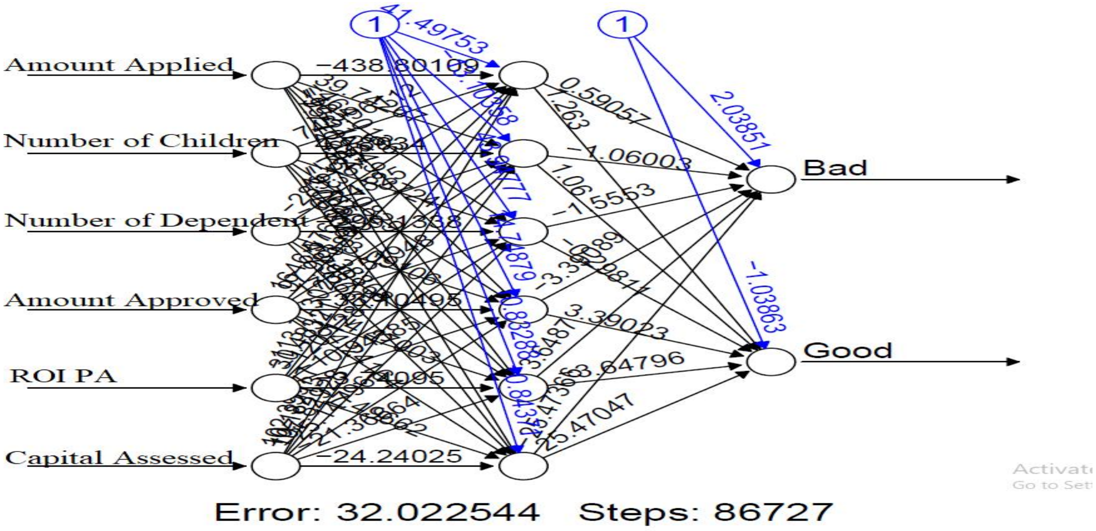


Figure 6.18: Artificial Neural Network model Structure with Six hidden nodes

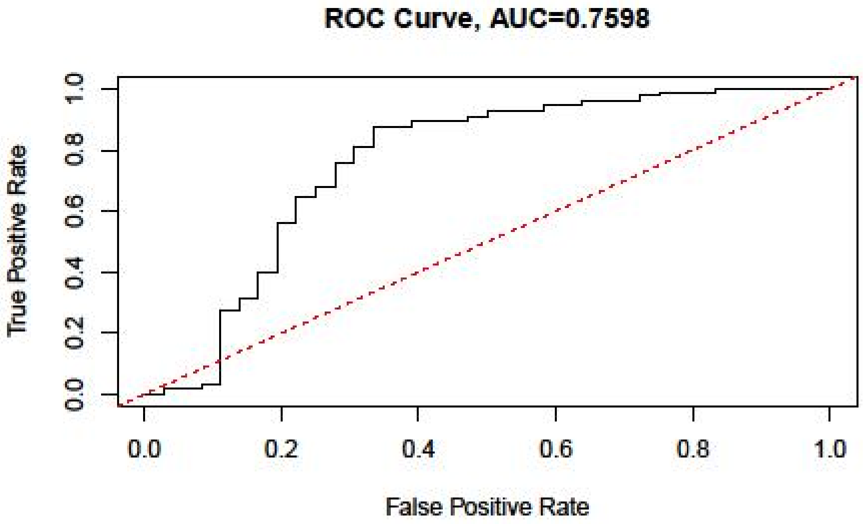


Figure 6.19: Artificial Neural Network model ROC curve with one hidden node and an AUC=0.7598

APPENDIX B

R Codes Used in Data Analysis

B1: Required Packages

```
library(HDoutliers);library(FactoMineR);library(factoextra);library(car);libr
ary(mlogit);library(MASS);library(readr);library(caret);library(pROC);library
(tidyverse);library(leaps);library(vegan);library(betapart);library(pastecs);
library(psych);library(ggplot2);library(ROCR); library(tibble);library(neural
net);library(NeuralNetTools);library(devtools); source_gist("6206737")
```

B2: Importing and Previewing Data

```
> data<-read.csv("C:/Users/ABBABELLO/Documents/Abba/MainData.csv",header=TRUE)
>
> data<- as_tibble(data)
> head(data)
> tail(data)
>
> str(data)
>
```

B3: Data Cleaning and Manipulation

```
> #Validation
> summary(data)

> #Consistency
> sum(is.na(data))

> #compleatness
> sum(duplicated(data))

> #Uniformity
> colnames(data)

> #Outliers
> outliers<-HDoutliers(data)
> outliers
```

B4: Exploratory Data Analysis

B4.1: Univariate

```
#Histogram
> hist(data$X1,xlab = "Age(X1)", main = "Histogram of Age", breaks = sqrt(nro
w(data)))
> hist(data$X4,xlab = "Experience in Business  (X4)", main = "Histogram of Ex
perience in Business  ", breaks = sqrt(nrow(data)))
> hist(data$X5,xlab = "Amount Applied (X5)", main = "Amount Applied", breaks
= sqrt(nrow(data)))
> hist(data$X6,xlab = "Number of Children (X6)", main = "Number of Children",
 breaks = sqrt(nrow(data)))
> hist(data$X7,xlab = "Number of dependents  (X7)", main = "Number of depende
nts", breaks = sqrt(nrow(data)))
> hist(data$X8,xlab = "Amount Approved (X8)", main = "Amount Approved", break
s = sqrt(nrow(data)))
> hist(data$X9,xlab = "ROI PA (X9)", main = "ROI PA", breaks = sqrt(nrow(dat
a)))
```

```
> hist(data$X10,xlab = "Capital Assessed(X10)", main = "Capital Assessed", br
eaks = sqrt(nrow(data)))

#Bar Chart
> A<-ggplot(data, aes(x=X2)) +geom_bar()
> A
> B<-ggplot(data, aes(x=X3)) +geom_bar()
> B
> C<-ggplot(data, aes(x=X11)) +geom_bar()
> C
> D<-ggplot(data, aes(x=Y)) +geom_bar()
> D

> Descriptive_Stat<-round(stat.desc(data[,c("ï..X1","X4","X5","X6","X7","X8",
"X9","X10")],basic = TRUE,desc=TRUE,norm = TRUE,p=0.99),digits = 2)
> Descriptive_Stat
```

B4.2: BIVARIATE

```
#Covariance

> round(cov(data[,c("ï..X1","X4","X5","X6","X7","X8","X9","X10")], use="compl
ete.obs"),digits = 2)

#Correlation

> corr <- round(cor(data[,c("ï..X1","X4","X5","X6","X7","X8","X9","X10")], us
e='complete.obs'),2)

> corr
```

B4.3: Multivariate

```
> ggcorrplot(corr, lab=TRUE, title='Correlation Heatmap', colors=c('#022D36',
 'white', '#48AAAD'))
```

B5: Variable selection

```
>Model<-glm(factor(Y)~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10+X11,data=Data,family=bin
omial())
Var_Sel_model<-stepAIC(model,direction="both",trace=FALSE)
```

B6: Data Scaling and Decoding

```
> ParameterTrain<-Train%>%preProcess(method=c("center","scale"))
> NomTrain<-ParameterTrain%>%predict(Train)

> scaledata$Y<-factor(scaledata$Y,levels=c("Good","Bad"),labels=c(0,1))
> scaledata$X11<-factor(scaledata$X11,levels=c("Good","Bad"),labels=c(0,1))
> scaledata$X3<-factor(scaledata$X3,levels=c("Male","Female"),labels=c(0,1))
> scaledata$X2<-factor(scaledata$X2,levels=c("Single","Married","Divorced","W
idowed"),labels=c(0,1,2,3))
```

B7: Dimensionality Reduction using FAMD

```
DimReductFAMD<-FAMD(data,ncp=5,graph=FALSE)
```

B8: Linear Discriminant Analysis

```
> LDA<-lda(Y~coord.Dim.1+coord.Dim.2+coord.Dim.3+coord.Dim.4+coord.Dim.5,data
=Train)
> LDA
> #PREDICTION
> prediction<-predict(LDA,Test[,1:5],type="response")
> table<-table(Test$Y,prediction$class)
> confusionMatrix(table)
> ModelAccuracy<-mean(prediction$class==Test$Y)
> ModelAccuracy
> pred<-prediction(prediction$posterior[,2],Test$Y)
> performance<-performance(pred,"tpr","fpr")
> AUC<-performance(pred,"auc")
> AUC1<-as.numeric(AUC@y.values)
> AUC;AUC1
> plot(performance,colorize=TRUE,main="ROC CURVE --- LINEAR DISCRIMINANT ANAL
YSIS; AUC= 0.8494318")
> #VALIDATION
> train.control <- trainControl(method = "cv", number = 10)
> model <- train(Y~coord.Dim.1+coord.Dim.2+coord.Dim.3+coord.Dim.4+coord.Dim.
5, data =Validation, method = "lda",trControl = train.control)
> model
```

B9: Logistic Regression

```
> model<-glm(factor(Y)~coord.Dim.1+coord.Dim.2+coord.Dim.3+coord.Dim.4+coord.
Dim.5,data=Training,family=binomial())
> summary(model)
> #PREDICTION
> D<-Test[,1:5]
> prob_pred<-predict(model,D,type="response")
> pred<-ifelse(prob_pred>0.5,"Bad","Good")
> ModelChiSqr<-model$null.deviance-model$deviance
> ModelChiSqr
> ChiSqrDF<-model$df.null-model$df.residual
> ChiSqrDF
> ChiSqrProb<-1-pchisq(ModelChiSqr,ChiSqrDF)
> ChiSqrProb
> Hosmer_and_Lemeshow_R2<-ModelChiSqr/model$null.deviance
> Hosmer_and_Lemeshow_R2
> Cox_and_Snell_R2<-1 - exp ((model$deviance - model$null.deviance) /125)
> Cox_and_Snell_R2
> Nagelkerke_R2<-Cox_and_Snell_R2/(1-(exp(-(model$null.deviance/125))))
> Nagelkerke_R2
> confusionMatrix(table(pred,Test$Y))
> #ROC
> ROC<-roc(Test$Y,prob_pred)
> ROC
> plot(ROC,colorize=TRUE,main="LOGISTIC REGRESSION AUC=0.8537")
> #VALIDATION
> train.control <- trainControl(method = "cv", number = 10)
> Validmodel <- train(factor(Y) ~ coord.Dim.1 + coord.Dim.2 + coord.Dim.4, da
ta = Validation, method = "glm",family = binomial(),trControl = train.control)
```

B10: Normalizing data using min-max normalization

```
> process<-preProcess(dt,method = c("range"))
> data<-predict(process,dt)
```

B11: Splitting data into training, validation and test sets (60:20:20)

```
> S<-sample(seq(1,3),size=nrow(data),replace=TRUE,prob=c(0.6,0.2,0.2))
> Train<-data[S==1,]
> Test<-data[S==2,]
> Valid<-data[S==3,]
> detach(package:caret,unload = T)
```

B12: Variable selection using Garson’s method

```
> cols<-colorRampPalette(c('lightgreen','lightblue'))
> G<-gar.fun("Y",nn1,bar.plot=T,struct=NULL,x.lab=NULL,y.lab=NULL,wts.only=F)
```

B13: One hidden node

```
> nn1<-neuralnet(Y~X5+X6+X7+X8+X9+X10,data=Train,hidden = 1, threshold = 0.01,
stepmax = 1e+05, rep = 20, startweights = NULL,learningrate.limit = NULL,lear
ningrate.factor = list(minus = 0.5, plus = 1.2),learningrate=NULL, lifesign =
 "none",lifesign.step = 2000, algorithm = "rprop+",err.fct = "sse", act.fct =
 "logistic",linear.output = TRUE, exclude = NULL,constant.weights = NULL, lik
elihood = TRUE)
> plot(nn1, rep="best")
> T<-subset(Test,select = c(X5,X6,X7,X8,X9,X10))
> predicted_probs<-predict(nn1,newdata = T,type="prob")
> positive_probs<-predicted_probs[,1]
> negative_probs<-predicted_probs[,2]
> confusionMatrix(table(pred1[,1],Test$Y))
> roc_data<-pROC::roc(response=Test$Y,predictor=positive_probs)
> tpr<-roc_data$sensitivities
> fpr<-1-roc_data$specificities
> AUC<-roc(Test$Y,positive_probs)
> AUC
> plot(fpr,tpr,type="l",main="ROC Curve, AUC=0.9062",xlab="False Positive Rat
e",ylab="True Positive Rate")
> abline(0,1,lty=2,col="red")
> pred1<-ifelse(predicted_probs>0.5,"Bad","Good")
```

B14: Two hidden nodes

```
> nn2<-neuralnet(Y~X5+X6+X7+X8+X9+X10,data=Train,hidden = 2, threshold = 0.01,
stepmax = 1e+05, rep = 20, startweights = NULL,learningrate.limit = NULL,lear
ningrate.factor = list(minus = 0.5, plus = 1.2),learningrate=NULL, lifesign =
 "none",lifesign.step = 2000, algorithm = "rprop+",err.fct = "sse", act.fct =
 "logistic",linear.output = TRUE, exclude = NULL,constant.weights = NULL, lik
elihood = TRUE)
> plot(nn2, rep="best")
> T<-subset(Test,select = c(X5,X6,X7,X8,X9,X10))
> predicted_probs2<-predict(nn2,newdata = T,type="prob")
> pred2<-ifelse(predicted_probs2>0.5,"Bad","Good")
> confusionMatrix(table(pred2[,1],Test$Y))
> positive_probs2<-predicted_probs2[,1]
> negative_probs2<-predicted_probs2[,2]
> roc_data2<-pROC::roc(response=Test$Y,predictor=positive_probs2)
> tpr2<-roc_data2$sensitivities
> fpr2<-1-roc_data2$specificities
> AUC2<-roc(Test$Y,positive_probs2)
> AUC2
> plot(AUC2)
> plot(fpr2,tpr2,type="l",main="ROC Curve, AUC=0.7856",xlab="False Positive R
ate",ylab="True Positive Rate")
> abline(0,1,lty=2,col="red")
```

B15: Three hidden nodes

```
> nn3<-neuralnet(Y~X5+X6+X7+X8+X9+X10,data=Train,hidden = 3, threshold = 0.01,
stepmax = 1e+05, rep = 20, startweights = NULL,learningrate.limit = NULL,lear
ningrate.factor = list(minus = 0.5, plus = 1.2),learningrate=NULL, lifesign =
 "none",lifesign.step = 2000, algorithm = "rprop+",err.fct = "sse", act.fct =
 "logistic",linear.output = TRUE, exclude = NULL,constant.weights = NULL, lik
elihood = TRUE)
> plot(nn3, rep="best")
> predicted_probs3<-predict(nn3,newdata = T,type="prob")
> pred3<-ifelse(predicted_probs3>0.5,"Bad","Good")
> confusionMatrix(table(pred3[,1],Test$Y))
> positive_probs3<-predicted_probs3[,1]
> negative_probs3<-predicted_probs3[,2]
> roc_data3<-pROC::roc(response=Test$Y,predictor=positive_probs3)
> tpr3<-roc_data3$sensitivities
> fpr3<-1-roc_data3$specificities
> AUC3<-roc(Test$Y,positive_probs3)
> AUC3
> plot(fpr3,tpr3,type="l",main="ROC Curve, AUC=0.8583",xlab="False Positive R
ate",ylab="True Positive Rate")
> abline(0,1,lty=2,col="red")
```

B16: Four hidden nodes

```
> nn4<-neuralnet(Y~X5+X6+X7+X8+X9+X10,data=Train,hidden = 4, threshold =
0.01,stepmax = 1e+05, rep = 20, startweights = NULL,learningrate.limit =
NULL,learningrate.factor = list(minus = 0.5, plus = 1.2),learningrate=NULL,
lifesign = "none",lifesign.step = 2000, algorithm = "rprop+",err.fct = "sse",
act.fct = "logistic",linear.output = TRUE, exclude = NULL,constant.weights =
NULL, likelihood = TRUE)

> plot(nn4, rep="best")
> T<-subset(Test,select = c(X5,X6,X7,X8,X9,X10))
> predicted_probs4<-predict(nn4,newdata = T,type="prob")
> pred4<-ifelse(predicted_probs4>0.5,"Bad","Good")
> confusionMatrix(table(pred4[,1],Test$Y))
> positive_probs4<-predicted_probs4[,1]
> negative_probs4<-predicted_probs4[,2]
> roc_data4<-pROC::roc(response=Test$Y,predictor=positive_probs4)
> tpr4<-roc_data4$sensitivities
> fpr4<-1-roc_data4$specificities
> AUC4<-roc(Test$Y,positive_probs4)
> AUC4
> plot(fpr4,tpr4,type="l",main="ROC Curve, AUC=0.824",xlab="False Positive Ra
te",ylab="True Positive Rate")
> abline(0,1,lty=2,col="red")
```

B17: Five hidden nodes

```
> nn5<-neuralnet(Y~X5+X6+X7+X8+X9+X10,data=Train,hidden = 5, threshold = 0.01,
stepmax = 1e+05, rep = 20, startweights = NULL,learningrate.limit = NULL,lear
ningrate.factor = list(minus = 0.5, plus = 1.2),learningrate=NULL, lifesign =
 "none",lifesign.step = 2000, algorithm = "rprop+",err.fct = "sse", act.fct =
 "logistic",linear.output = TRUE, exclude = NULL,constant.weights = NULL, lik
elihood = TRUE)
> plot(nn5, rep="best")
> T<-subset(Test,select = c(X5,X6,X7,X8,X9,X10))
> predicted_probs5<-predict(nn5,newdata = T,type="prob")
> pred5<-ifelse(predicted_probs5>0.5,"Bad","Good")
> confusionMatrix(table(pred5[,1],Test$Y))
> positive_probs5<-predicted_probs5[,1]
> negative_probs5<-predicted_probs5[,2]
> roc_data5<-pROC::roc(response=Test$Y,predictor=positive_probs5)
> tpr5<-roc_data5$sensitivities
> fpr5<-1-roc_data5$specificities
> AUC5<-roc(Test$Y,positive_probs5)
> AUC5
> plot(fpr5,tpr5,type="l",main="ROC Curve, AUC=0.8785",xlab="False Positive R
ate",ylab="True Positive Rate")
> abline(0,1,lty=2,col="red")
```

B18: Six hidden nodes

```
> nn6<-neuralnet(Y~X5+X6+X7+X8+X9+X10,data=Train,hidden = 6, threshold = 0.01,
stepmax = 1e+05, rep = 20, startweights = NULL,learningrate.limit = NULL,lear
ningrate.factor = list(minus = 0.5, plus = 1.2),learningrate=NULL, lifesign =
 "none",lifesign.step = 2000, algorithm = "rprop+",err.fct = "sse", act.fct =
 "logistic",linear.output = TRUE, exclude = NULL,constant.weights = NULL, lik
elihood = TRUE)
Warning message:
Algorithm did not converge in 5 of 20 repetition(s) within the stepmax.
> plot(nn6, rep="best")
> plot(nn6, rep="best")
> T<-subset(Test,select = c(X5,X6,X7,X8,X9,X10))
> predicted_probs6<-predict(nn6,newdata = T,type="prob")
> pred6<-ifelse(predicted_probs6>0.5,"Bad","Good")
> confusionMatrix(table(pred6[,1],Test$Y))
> positive_probs6<-predicted_probs6[,1]
> negative_probs6<-predicted_probs6[,2]
> roc_data6<-pROC::roc(response=Test$Y,predictor=positive_probs6)
> tpr6<-roc_data6$sensitivities
> fpr6<-1-roc_data6$specificities
> AUC6<-roc(Test$Y,positive_probs6)
> roc_data6
> AUC6
> plot(fpr6,tpr6,type="l",main="ROC Curve, AUC=0.7598",xlab="False Positive R
ate",ylab="True Positive Rate")
> abline(0,1,lty=2,col="red")
```

B19: Support Vector Machine

```
> model=svm(Y~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10+X11,data=Train,type="C-classific
ation",kernel="linear")
> prediction<-predict(model,Test[,1:11],type="class")
> ConfusionMatrix<-confusionMatrix(table(prediction,Test$Y))
> ROC<-roc(prediction,as.numeric(Test$Y))
> plot(ROC, main="SVM AUC=0.8532")
```

B20: Decision Tree

```
>model<-rpart(Y~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10+X11,data=Train,method = "class
")
> rpart.plot(model,extra = 106)
> prediction<-predict(model,Test[,1:11],type="class")
> confusionMatrix(table(prediction,Test$Y))
> ROC<-roc(prediction,as.numeric(Test$Y))
> plot(ROC, main="Decision Tree AUC=0.7524")
```

B21: K-Nearest Neighbors at K=1

```
> model<-knn(train = Train, test = Test,cl=Train$Y,k=1)
> missclass<-mean(model!=Test$Y)
> print(paste("Accuracy=",1-missclass))
```

B22: K-Nearest Neighbors at K=2

```
> model2<-knn(train = Train, test = Test,cl=Train$Y,k=2)
> missclass2<-mean(model2!=Test$Y)
> print(paste("Accuracy=",1-missclass2))
```

B23: K-Nearest Neighbors at K=6

```
> model6<-knn(train = Train, test = Test,cl=Train$Y,k=6)
> missclass6<-mean(model6!=Test$Y)
> print(paste("Accuracy=",1-missclass6))
```

B24: K-Nearest Neighbors at K=7

```
> model7<-knn(train = Train, test = Test,cl=Train$Y,k=7)
> missclass7<-mean(model7!=Test$Y)
> print(paste("Accuracy=",1-missclass7))
```

B25: K-Nearest Neighbors at K=10

```
> model10<-knn(train = Train, test = Test,cl=Train$Y,k=10)
> missclass10<-mean(model10!=Test$Y)
> print(paste("Accuracy=",1-missclass10))
```

B26: K-Nearest Neighbors at K=11

```
> model11<-knn(train = Train, test = Test,cl=Train$Y,k=11)
> missclass11<-mean(model11!=Test$Y)
> print(paste("Accuracy=",1-missclass11))
```

B27: K-Nearest Neighbors at K=12

```
> model12<-knn(train = Train, test = Test,cl=Train$Y,k=12)
> missclass12<-mean(model12!=Test$Y)
> print(paste("Accuracy=",1-missclass12))
```

B28: K-Nearest Neighbors at K=13

```
> model13<-knn(train = Train, test = Test,cl=Train$Y,k=13)
> missclass13<-mean(model13!=Test$Y)
> print(paste("Accuracy=",1-missclass13))
```

B29: K-Nearest Neighbors at K=14

```
> model14<-knn(train = Train, test = Test,cl=Train$Y,k=14)
> missclass14<-mean(model14!=Test$Y)
> print(paste("Accuracy=",1-missclass14))
```