UNIVERSITY of RWANDA

AFRICAN CENTER OF EXCELLENCE IN DATA SCIENCE

ACE-DS

Enhancement of Credit Score Prediction for imbalanced Datasets Using Data Mining Approaches

(Case study: Bank of Kigali)

Masters Research Thesis

By: BETT KIPKIRUI ERICK

Registration Number: 219014268

A dissertation submitted in partial fullment of the requirements for the degree of MASTER OF SCIENCE IN DATA SCIENCE (OPTION: DATA MINING).

In the college of Business and Economics

Supervisor: **Dr. Marcel Ndengo**

September, 2020.

DECLARATION

Declaration by the candidate

I declare that this dissertation contains my own work except where specifically acknowledged, and it has been passed through ant-plagiarism system and found to be complaint and this is the approved version of the Thesis:

Bett Kipkirui Erick

Reg No 219014268

Signature:

Date: __9/16/2020_________________

Supervisor

Dr. Marcel Ndengo

Date: _____9/16/2020______________

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude to my supervisor Dr. Marcel Ndengo for his guidance, positive criticism and a lot of patience during his supervision of this thesis. I am greatly indebted to Prof. Cheruiyot, Dr. Geoffrey Yegon, Mrs Chebet Noela and Mr. C.Mutai for their valuable comments on this thesis.

My sincere gratitude also goes to my dear parents Mr. Solomon Kibet Chebole and Mrs Rachel Cherotich Chebole, for their love, support and prayers which made everything worthwhile.

I must express lots of appreciation to the entire staff of Africa Centre of Excellence including Dr Charles, Dr Kabano and other staffs at large for their support. Finally, I wish to thank my friends and classmates Francis Yego among others for their constant assistance during write up of this thesis.

ABSTRACT

Credit score prediction is the most effective way of analyzing whether a potential client is eligible for loan or not especially in financial institutions where class imbalance problems are prevalent. However, limited number of credit score prediction models in banking institutions take into consideration imbalance data and again, the best resampling technique to be applied with imbalanced data is still a challenge.

Therefore, in an attempt to address these problems, this research presents an empirical comparison of various combinations of data imbalance resampling techniques and machine learning algorithms used to address this challenge of imbalance data. This study utilized credit score secondary data from bank of Kigali with 58096 customer transaction with 31 variables. The time scope of data was limited to 2018-2019. Modelling the data and handling class imbalance was done using python jupyter notebook libraries.

The credit score prediction from each combination were evaluated with F_Beta, F1_score, precision, recall score to avoid biasness towards majority class hence taking into consideration effectiveness in each technique and model used which have not been considered in similar studies.

An experimental result was done in this research using resampling techniques and machine learning algorithms to enhance credit score prediction for bank of Kigali clients. The findings suggest that combining oversampling technique and random forest algorithm yield the best prediction of 96.42% and F_Beta of 97.56% among the rest hence the most effective way of evaluating the eligibility of customers in the banking institution.

KEY WORDS

Machine learning, credit Score, imbalance data, algorithm, defaulter, non-defaulter

LIST OF SYSMBOLS AND ACRONYM

SMOTE-Synthetic Minority Oversampling Technique

KNN- K Kearest Neighbour

XGB-Extreme gradient boosting

ECL- Expected Credit Losses

SVM-Support Vector Machine

Rwf-Rwandan Franc

BNR- National Bank of Rwanda

ID-Identification

NC-Negative Correlation

PR-Probability

ß -Beta

θ -Theta

IDE- Integrated Development Environment

NB-Naïve Bayes

EDA- Exploratory Data Analysis

RF-Random Forest

ROC-Receiver Operating Characteristics Curve

TPR- True Positive Rate

FPR- False Positive Rate

TABLE OF CONTENTS

LIST OF TABLES

Table N^{o} **Title** **Page N^{o}**

LIST OF FIGURES

Figure N^o **Title** **Page N^o**

CHAPTER ONE: INTRODUCTION

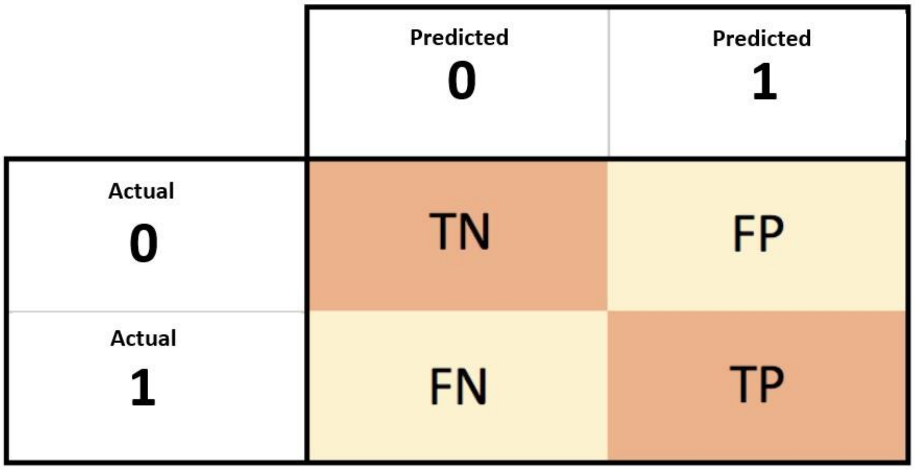
1.0 INTRODUCTION

Imbalance data issues occur as a result of skewness of data. This challenge is common in data mining thereby creating a negative impact on classification processes in machine learning [1]. It occurs when classes have different ratios of specimens in which a large number of specimens belong to one class and the other class has fewer specimens that is usually an essential class, but unfortunately misclassified by many classifiers. So far, different approaches such as oversampling, under-sampling, SMOTE among others have been implemented to mitigate this problem [2].

This study focused on machine learning techniques to improve credit score prediction. Credit score is essential in banking institutions since it determines if the customer is eligible to borrow loan based on the outlined rules and regulations of the banking institution. In most cases, customers are credited based on their history and can be classified as ’good’ or ’bad’ customer. Due to imbalanced data, most institutions end up denying loan to clients with ‘good’ history instead of allowing them simply because of misclassification [3].

In another scenario, the classifier might allow loan to customer with bad history simply because of misclassification caused by imbalanced data which leads to false positive in the minority class.

Confusion matrix table:



	Predicted 0	Predicted 1
Actual 0	TN	FP
Actual 1	FN	TP

Fig. 1.0 Confusion Matrix

In the above fig.1, **0** represents eligible/non-defaulters while **1** represents. True positives (TP) where the

True positives (TP): These are the cases where clients are predicted as being ineligible when they actually do not qualify to be given loan.

True negatives (TN): These are the cases where clients are predicted as being eligible when they actually qualify to be given loan.

False positives (FP): These are the cases where clients are predicted as ineligible when they actually qualify to be given loan

False negatives (FN): These are the cases where clients are predicted as eligible when they actually do not qualify to be given loan

A number of reasons have been highlighted which commonly bring about poor performance on existing classification algorithms on skewed data [17]. These include:

1. Assumption that all classes have same distribution of data.

2. Accuracy is the main goal i.e. target to minimize the overall error caused by minority class which in most cases are very little.

3. The assumption that the error coming from both majority class and minority class have equal cost. Therefore, with imbalance data sets, data mining algorithms produce degraded models which fail to take into consideration minority class since most of them assume balanced dataset.

The goal of this research was to address these challenges in the financial sections using data mining and machine learning techniques. Some of the techniques applied to handle imbalance data include the SMOTE (Synthetic minority over Sampling Technique) [4]. This technique creates synthetic samples. It applies neighbor's algorithm to generate new and synthetic data that is used in training the model. Oversampling technique is another important technique explained by [5] which involved simply adding more samples of minority class in order to merge or balance with majority class. The majority class samples are generated by applying one of the following methods: Repetition, SMOTE or Bootstrapping. This method is applicable especially when there is insufficient dataset. Under-sampling technique minimizes the size of majority class thereby balancing the data [6]. It is used especially when there is sufficient data. This is done by maintaining rare class samples (minority class samples) and selecting a random sample of size from majority class therefore a balanced new dataset was achieved by modelling further. NearMiss is a type of under-sampling technique which aims to balance class distribution in the data set by randomly eliminating majority class samples. This technique therefore create space between two classes by completely removing instances of majority class when they are different and close to one another [7].

Cost-Sensitive Learning technique is another method of data mining that considers misclassification cost (and probably other forms of cost) [8]. The aim of this type of learning is to reduce the overall cost. The main difference between cost-insensitive and cost-sensitive learning techniques is that cost-sensitive learning handles various misclassifications differently. Cost insensitive learning does not consider misclassification cost. Its objective is to achieve a high accuracy classification of examples into a set of well-known classes. Class imbalance occurs in several real-world applications where there is high skewness class distributions of data. To solve this problem, cost-sensitive learning is the main approach.

Feature engineering is another approach in handling imbalance data [9]. This is a technique where the features are manually or automatically filtered/selected thus contributing to predicted variable (y predictor). Having unnecessary features in a dataset which affects prediction negatively lowers your prediction accuracy hence needs to be wiped out from training data [10].

Ensemble learning technique combines machine learning algorithms to enhance prediction on credit score. These includes support vector machine, k nearest neighbor [K-NN], logistic regression, random forest, naïve bayes, and gradient boosting, XGBoosting and decision tree. Based on the nature of the data, these machine learning techniques enable boosting of credit score prediction [11].

1.1 PROBLEM STATEMENT

According to [12], real world data sets encounter three main challenges of data imbalance. The first one is machine learning problem. Machine learning algorithms are developed to reduce classification errors. But due to the significant high number of instances in the majority class, the algorithms are biased towards classifying new instances to the majority class. In a loan portfolio for instance where default rate of 5%, the algorithm average has the tendency to classify new loan applications to non-default class since it would be accurate 95% of the time.

Secondly, usual practices are mostly confirmed by experts in credit risk as oppose to empirical studies [3]. This is indeed not optimal, by the fact that the population might be very different from the other bank's population. This implies that different loan portfolio works independently with one another.

In credit control, imbalanced credit dataset is a database where class of defaulters is highly under-represented as oppose to class of non-defaulters [10]. That is, majority class (defaulters) is highly under-represented compared to minority class (non-defaulters). Imbalance data if not well handled can cause serious problems in credit prediction. Furthermore, credit scoring evaluates credit risks occurrences, improves cash flow in the institution, reduces possible risks and assists in making managerial decisions.

This problem of data imbalance in credit card fraud is a world application challenge. According to [8], blue chip companies in 2017 lost to approximate $2 billion, JP Morgan chase $23.5 billion and Berkshire Hathaway $24 billion. Experts on the same note predicted 32 billion lost on online credit card fraud.

In banking institution in Rwanda, loan defaulting is still high simply because the measures taken to mitigate and issue the loan is still a challenge to loan lenders in financial institutions. According to [13] an audit done on 1st January 2018 by ECL (expected credit losses ECL) group, a total of FRw 32.463 billion was lost on credit loan. This sum of money is huge and is attributed to two main factors: Economic scenarios whereby there is limited significant judgement to use while giving out loans and model estimations which use migration matrix techniques such as cohort method and duration method on a 12-month basis. The probability of default for off-balance sheet items has been assumed to be equal to the probability of default of the business segment they relate to.

These two main challenges are far and wide caused by skewed nature of data and traditional prediction techniques while lending out loans. Misclassification due to skewed data affects credit score negatively. These challenges are based on poor insight of the client/customer history which ends up misclassifying the client due to skewed nature of the data.

This research therefore focused on solving data misclassification using data mining techniques and machine learning algorithms which include SMOTE, oversampling, under-sampling, NearMiss, cost sensitive learning and ensemble learning techniques to enhance credit score prediction processes.

1.1.0 RESEARCH OBJECTIVES

The main objective of this work is to enhance credit score prediction through utilization of imbalance data mining approach. Specific objective of this study includes the following:

1. To assess the performance of imbalance data mining techniques on credit score prediction
2. To determine the performance of each resampling technique on credit score prediction.
3. To evaluate effectiveness of each machine learning algorithm using performance metrics on credit score prediction

1.1.1 RESEARCH QUESTIONS:

1. How is the performance of imbalance data mining techniques on credit score prediction?
2. What is the level of performance of each resampling technique on credit score prediction?
3. What is the effectiveness of performance metrics on each machine learning algorithms on credit score prediction?

1.2 SIGNIFICANCE OF THE STUDY

This study brings significant impact in both the academia world and financial institutions. On the academia side, this research sets a platform for other researchers to learn on the same field and do further research in order to get more insight which will contribute to enhancement of classification issues especially on credit scoring. While on financial institutions, it will assist to better fix the data imbalance before solving any classification problem therefore improving prediction which will eradicate misclassification issues. With improved prediction on the credit score, loan losses would reduce drastically. To managers of bank of Kigali, more insight will be obtained hence enable loan portfolio to be efficiently managed due to actual and accurate report from prediction of credit scores since the right people with good history will be eligible for loan while the rest are denied due to questionable history.

1.3 JUSTIFICATION

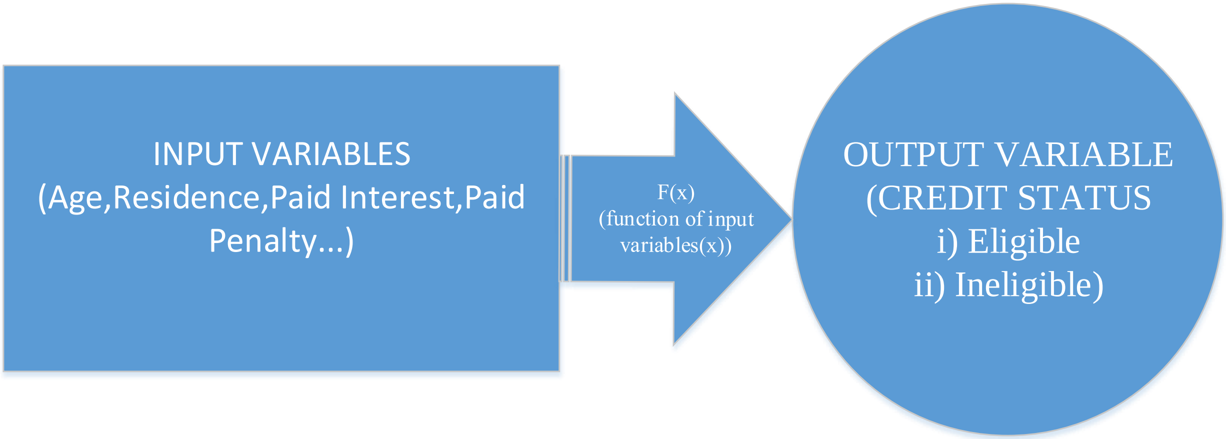
This research assisted in balancing imbalance dataset thus helping in predicting credit of each client/customer. Data imbalance handling techniques such as SMOTE, Oversampling, under-sampling, NearMiss, cost sensitive learning, feature engineering, and machine learning models like random forest, Naïve Bayes, decision tree, gradient boosting, XGBoosting, SVM and KNN were applied [6].

1.4 SCOPE

This study was applied in a financial institution (Bank of Kigali). The main focus was to fix misclassification problem applied to loan applications by solving two main issues: First, balancing the dataset using data imbalance handling techniques (SMOTE, oversampling, under-sampling, NearMiss, Cost sensitive and feature engineering) and secondly, choosing the best machine learning algorithm within a pre-defined set of algorithms to improve credit score prediction hence solving misclassification problem in the financial sector. A data set from Bank of Kigali was used to implement this research. Methodology applied quantitative paradigm where data imbalance handling techniques and machine learning algorithms were used for modelling.

1.5 THEORETICAL FRAMEWORK

This study is rooted on prospect of banking industry theory. It describes in detail crucial information that need to be observed and possible risks incurred when poor measures are put in place to check insights on customers details while making decisions on credit [14]. Therefore, this credit risk information is no exception in any financial institution. Thus, it is paramount to have a basis for decision on variables to focus on financial institution setting. These will be predictor features with much weight that this study seeks to utilize in order to achieve its objectives. Below is illustration of application theory of the current study.


INPUT VARIABLES
(Age,Residence,Paid Interest,Paid Penalty...)
F(x)
(function of input variables(x))
OUTPUT VARIABLE
(CREDIT STATUS
i) Eligible
ii) Ineligible)

1.6 LIMITATIONS OF THE STUDY

As far as the study entails credit scoring in the banking institutions, there emerge some limitation during the research. This includes the following:

1.6.1 Limited access to data

The bank did not provide complete open access to all the information available about clients. This led to limited findings in the study.

1.6.2 Time Constrains

The time required to complete the study was somehow short given a number of previous studies both empirical and theoretical had to be undertaken before embarking on the analysis and drawing conclusion on the same field scope. Again, more time was required on analysis in order to obtain detailed findings.

CHEPTER TWO: LITERATURE REVIEW

2.1 INTRODUCTION TO THE REVIEW

This section highlights previous studies and achievement in solving the problem of data imbalance in the financial sector. It goes ahead to check various concepts under the study and draw limitations from the past studies. Theoretical framework is also provided herein

2.2 EMPIRICAL REVIEW ON DATA IMBALANCE APPROACH

Loan defaulting is the single biggest threat to banks profitability and efficiency. It remains a big problem since the process of sieving out defaulters is very stringent. Getting credit worthiness relies on the customer's history of credibility [11]. In the past years, paper collections and other excellent surveys that capture advances in imbalanced learning field have been published [15],[10],[16]and [17].

A research done by [1] discussed in depth data preprocessing techniques such as data cleaning, transformation, reduction, and time allocated to prepare, sample and even clean imbalanced datasets. In addition, two resampling techniques (oversampling and under-sampling) were explained. From the results obtained, prediction with imbalance data was significantly worse compared with balanced data. Moreover, the results showed that SMOTE and random under-sampling techniques were not significantly different.

The impact of oversampling technique on imbalance data was discussed by [18]. From the experiment carried out, oversampling technique increases computation cost of learning algorithm thus convenient when applied on data set with minority samples. Similar conclusion in imbalance data set after using the same technique (oversampling) was obtained by [19].

Another study by [20] investigated various imbalanced credit score issues with different imbalance levels for the performance and worthiness of one-class classifiers. The results suggested that one-class classifiers perform better easily when the minority class comprises less than or 2% of the data, while on the contrary, the two-class classifiers are chosen when the minority class constitutes at least 15% of the data.

On the other hand, a new oversampling technique (oversampling) of handling imbalanced data was proposed. The technique worked well by creating artificial minority class sample to match with the majority class. As much as oversampling works well, other techniques like hybrid sampling, under-sampling, ensemble learning and even cost sensitive learning were not explored well on the same research [21] , [22]. Two years later, [10] came up with more sophisticated techniques of handling imbalance credit datasets. The techniques include SMOTE, Oversampling and under sampling techniques. On the same research, machine learning algorithms were not utilized well in order to map the best handling imbalance learning technique with suitable algorithms thus leaving the issue for further research to settle misclassification problem.

In Rwanda, loan defaulting is still a big problem in the financial sector, although significant efforts have been put to improve bad loans. As reported by the [23], a Rwandan newspaper, the non-performing loans in mining rose to Rwf2.9 billion in June 2019 from Rwf22 million in June 2018 due to the fact that the prices of international commodities were slowing down. In terms of trade, the non-performing loans can be tracked down to four big loan facilities accounting for Rwf15.3 billion in only two banks.

Another report given by the National Bank of Rwanda(BNR), [24] reported that, the overall ratio of bad loans in microfinance to aggregate loans was 12.3% as of June 2017, almost double the 7.5% recorded for the same period 2016. The central bank attributed this to haphazard procedures in credit approval processes, which led to a drastic loss of Rwf3.6 billion ($4.3 million) made by microfinance institutions as at June 2017. Most of these loan defaulting issues are attributed to imbalanced data problems.

2.3 THEORETICAL REVIEW OF THE STUDY.

Lending in financial institution plays a key role in business operation. This facilitates economic efficiencies by financing small businesses and meeting emerging needs of clients which assist in tackling their problems [10]. In addition, it boosts the financial institution due to interest imposed on their clients whenever they take loan.

A super technique approach of using lasso logistic regression algorithm to curb risks caused by large unbalanced data sets due to misclassification was proposed by [25]. Since there are two levels on class imbalance problem that is data and algorithm levels. Considering other algorithms like decision tree, KNN, lasso logistic regression outshines the other methods on crediting score prediction from the findings [25]. On the other hand, [17] did a survey of some of the data imbalance handling techniques. Some of the techniques include random oversampling, random under sampling and feature engineering. Random oversampling yields better results on classification performance than the other techniques.

In a study of class imbalance problem, [26] explained the performance of AdaBoost.NC. This algorithm is an ensemble learning technique which combines the capabilities of boosting method and negative correlation learning. This is a hybrid algorithm majorly applied in multiclass imbalanced data problem. The research results showed that combination of AdaBoost.NC and random oversampling improves the accuracy of prediction while maintaining the overall performance.

Furthermore, [27] suggested that both ensemble learning and sampling techniques are effective in classification accuracy improvement of data streams that are skewed. Moreover, it was found out that SMOTEBoost which is a combination of SMOTE algorithm and AdaBoost is a hybrid technique which boosts performance of the components.

Modelling risks is also a key element to be factored in. A research done by [10], noticed dangers associated with modelling if not keenly taken into consideration. For instance, his findings showed that random forest had the best accuracy of 0.8176 but its specificity was too low (0.015) and at the same time this algorithm returned the highest false positive of 0.98. Therefore, accuracy should only be considered as an evaluation metric but rather other evaluations metrics such sensitivity, F1 score, F_Beta score, precision and specificity should be taken into consideration.

2.5 GAP IN THE PAST STUDIES

This research builds on the previous studies where they only side on resampling techniques. For instance, [10] tried to elaborate a number of handling techniques but some of them were not applied and mapped with the best algorithm to which they can give best prediction. On the other hand, [25] concentrated on lasso logistic regression technique thus limiting other algorithmic techniques which can be explored to yield better performance based on the nature of the dataset.

This research therefore tried to find out the best data imbalance handling techniques which include SMOTE, oversampling, under-sampling, cost sensitive learning, NearMiss and which machine learning algorithms are suitable to handle financial imbalance credit dataset using Bank of Kigali dataset to enhance credit score prediction.

2.6 CONCLUSION

Most studies majorly concentrate on imbalanced data without mapping the best machine learning algorithm that can solve misclassification problem. This research therefore dwelled on an array of imbalanced data handling techniques which includes SMOTE (synthetic minority oversampling technique), oversampling techniques, under sampling techniques, feature selection, NearMiss, cost sensitive learning, ensemble learning and map with the best machine learning algorithms such as decision tree, XGBoost, naïve bayes, logistic regression, k nearest neighbour, random forest and gradient boosting that yield better results thus classifying well this misclassification problem [7]. Furthermore, other evaluation metrics such precision score, sensitivity, F1 score, F Beta and specificity were taken into consideration in this research to check effectiveness of the model performance hence enhancing credit score prediction.

CHEPTER THREE: RESEARCH METHODOLOGY

3.1 INTRODUCTION

This section describes the fundamental techniques and methods that were utilized to meet the research objectives. These include software, resampling techniques and machine learning algorithms.

3.2 DATA COLLECTION

This study utilized secondary data from banking institution in Rwanda (Bank of Kigali). The data set contains a number of input variables and some of the includes: Customer identification, date of birth, age, place of residence, paid tax details, payment status, and duration of payment. To analyze the data from the banking institution, collected data history was limited from 2018 to 2019.

3.3 DATA CLEANING

Some data cleaning techniques were essential before modeling applied. The issues that required cleaning included; irrelevant attributes, missing attributes and redundant attributes [28] for instance variable ‘unnamed 0’ was dropped since it was duplication of variable ‘customer ID’ hence has no impact in the data set. Irrelevant attributes were identified and removed. Removing them aided in getting better results and made the data learning task less computationally expensive [9] this reduces the number of errors due reduction of number of features, that is curse of dimensionality.

3.4 DATA UNDERSTANDING AND RANDOMIZATION

After preparing data, it was explored in order to comprehend insights about it and figure out the analytical techniques to be employed. Python software was utilized to visualize the data using scatterplots to display relationship between numerical variables, seaborn for data visualization and matplotlib to enhance interactive visualization. The data was also randomized so that the training does not base on the sequence of the data.

3.5 MODELLING

This is an essential process in implementing credit score prediction. At this stage, prediction and feature engineering is spelt out as explained in [9] by selecting the features in the data set which have more weight in the output variable . The history of data set from Bank of Kigali was trained to predict the label variable. This process of modelling was basically training the machine algorithms to predict credit from features, tuning and validating the data set. Variables that were used include; age, gender, place of residence, principal amount, paid interest, paid penalty, amount due, due principal, due interest, overdue days, effective date, duration. Python programming language was applied to implement machine learning algorithms in this research. Libraries involved includes Scikit-Learn library which is a free software for machine learning. For the data to be ready for implementation, encoding of categorical variables that is converting variables to numeric which enables computer to understand. Moreover, imputation of missing values was performed using fillna function (mean) to check the variance.

3.6 MACHINE LEARNING ALGORITHMS AND IMBALANCE DATA HANDLING TECHNIQUES

The problem under study classification problem. It entailed predicting the class label for data points in the dataset. In this research on the banking industry, the aim was to predict whether a client belongs to the defaulting group or non-defaulting group. One of the methods used to solve classification problems is logistic regression. It is a suitable method for conducting analysis when the output variable is clear and dichotomous/ binary in nature such a way that it had two categories that's defaulters and non-defaulters. For example, (defaulter, non-defaulter).

3.6.1 Logistic Regression

In logistic regression, the objective is to estimate the $pr(Y = 1|X = x)$ and select the class k with the largest probability [25]**.**

Where: pr is probability, Y is output and X is stochastic variable, x is a possible value

Given that there are only two classes, that is defaulters and non-defaulters, we let the

$pr(x) = pr(Y = 1|X = x)$……………………….. Equation (1)

We suppose that the log of odds:

$\text{Log}\ (^{p}/_{1\text{-}p}) = ß_0 + ß_{1x1} + \dots\ ß_{nxn}$………………. equation (2) is a linear function of the predictor variables x. Therefore, estimating the output variable p(y) becomes much easier in the presence of parameter estimates ß$_0$, ß$_1$,… ß$_n$ .To find the probability that a customer is a defaulter or non-defaulter, we then apply maximization of the likelihood function using the Newton Raphson method [29] where the likelihood function L(β) is a function from R$^{(p+1)}$ to R .

3.6.2 Decision Trees

Decision trees is another common method used for classification problems. A decision tree is a tree structure that is flowchart-like, whereby every node represents a test on an attribute value, every branch denotes a test outcome, and the leaves of the tree denote class distributions or the classes [30].

The mathematical formulation of classification criteria using decision trees can be expressed as follows as explained by [31] :

Given training set vectors $xi \in R^n, i = 1, \dots, l$ and a label set vector $y \in R^l$ recursive partitioning of space is performed by decision tree grouping together samples having equal labels. Let Q represent the data point in a node. For every candidate split $\theta = (j, t_m)$ comprising of a feature j and threshold t_m, divide the data into two parts as:

Q_{left} (θ) and $Q_{right}(\theta)$ subsets :

$Q_{left}(\theta) = (x, y)|x_j <= t_m$ ……………………………………………………… equation (3)

$Q_{right}(\theta) = Q \setminus Q_{left}(\theta)$ ……………………………………………………..... equation (4)

The value of impurity at m is calculated with the aid of an impurity function $H()$, the choice depending on classification problem or regression:

$G(Q, \theta) = \frac{n_{left}}{N_m} H(Q_{left}(\theta)) + \frac{n_{right}}{N_m} H(Q_{right}(\theta))$ …………………………...... equation (5)

Parameters minimizing impurity are selected as:

$\theta^* = \text{argmin}_\theta\, G(Q, \theta)$ ……………………………………………………… equation (6)

Recurse, the subsets $Q_{left}(\theta^*)$ and $Q_{right}(\theta^*)$ until we reach a maximum depth that is allowable,

$N_m < \underset{samples}{min}$ or $N_m = 1$ …………………………………………… equation (7)

If a target is as a result of classification outcome having values 0, 1,…,K-1, for node m, representing a region R_m with N_m observations, let:

$p_{mk} = 1/N_m \sum_{x_i \in R_m} I\ (y_i = k)$ ……………………………….…. equation (8)

be the proportion of class k in node m. Some of the measures of impurity that are commonly used are Entropy given by:

$H(X_m) = \sum_k P_{mk} \log(p_{mk})$ ………………………………………. equation (9)

Gini $H(X_m) = \sum_k P_{mk}\ (1 - p_{mk})$ ……………………………………………….. equation (10)

and misclassification:

$H(X_m) = 1 - max(p_{mk})$ ………………………………………………………. equation (11)

whereby X_m represents the training set found in node m

3.6.3 Random Forest implementation in Scikit-learn

Scikit-learn calculates importance of nodes using Gini importance assuming two child nodes only for every decision tree [32] as:

$ni_j = w_j\, C_j - w_{left(j)} C_{left(j)} - w_{right(j)} C_{right(j)}$ ……………………………. equation (12)

Where:

- ni_j= the importance of node j
- w_j = weighted number of samples reaching node j
- C_j= the impurity value of node j
- $left_j$= child node from left split on node j
- $right_j$ = child node from right split on node j

we then calculate the importance of each feature *i* that is fi_i on decision tree:

$$fi_i = \frac{\sum_{j:node\ j\ splits\ on\ feature\ i} ni_j}{\sum_{k\ \in all\ nodes} ni_k} \text{ ………………………………. equation (13)}$$

The importance of each feature can then be subjected to normalization to a value in the range (0,1) by dividing by the summation of all the values of the feature importance as follows:

$$norm\ fi_i = \frac{fi_i}{\sum_{j\ \in all\ features} fi_j} \text{ …………………………………… equation (14)}$$

At the level of the random forest, the feature importance is eventually calculated by calculating the average over all trees. This is found by summing up the value of the feature importance on every tree then dividing it by the total number of trees as calculated below:

$$RFfi_i = \frac{\sum_{j\ \in all\ trees} norm\ fi_{ij}}{T} \text{ …………………………………….. equation (15)}$$

Where:

- $RFfi_i$= the feature importance of i computed from all trees in the Random Forest model
- $normfi_{ij}$= the normalized importance for feature i in tree j
- T = number of total trees

3.6.4 SMOTE

SMOTE (Synthetic minority over Sampling Technique) is one of the techniques applied to carry out imbalance data. According to [4], this technique creates synthetic samples. It applies neighbor's algorithm to generate new and synthetic data that is used in training the model.

For instance if (xnew) is generated synthetic for xi, SMOTE basically selects an element xi in Kxi and xi in χmin. Therefore xnew feature vector is the sum of feature vectors and value.

This can be obtained by multiplying vector difference of xi and xi with the random value δ from 0 to 1 (δϵ[0,1]) thus equation

$$Xnew=xi+(xi-xi)*\delta$$

Where xi is an element in kxi

3.6.5 NearMiss

NearMiss is a type of under-sampling technique that aims to create a balance in distribution of the classes in the data set by randomly eliminating majority class samples. This technique therefore create space between two classes by completely removing instances of majority class when they are different and close to one another [7]. This is achieve by finding the distance between all instances of minority and majority classes where majority class is under sampled. Secondly the n instances with the smallest distances to the instances in minority class are chosen from the majority classes. Finally, the nearest method will produce k*n instance of the majority class if there are k instances in the minority class.

3.6.6 Oversampling

Oversampling technique is another important technique explained by [5] which involved simply adding more samples of minority class in order to merge or balance with majority class. The majority class samples are generated by applying one of the following methods: Repetition, SMOTE or Bootstrapping. This technique is applicable especially when there is insufficient dataset and if not handled well can lead to overfitting of minority class hence leading to increased generalization error

3.6.7 Under-sampling

Under-sampling technique creates a balance in the data by minimizing the majority class size [6]. It is mostly used when there is abundant data. This is accomplished by maintaining the rare class samples and randomly selecting same sample size from majority class therefore a balanced new dataset was achieved by modelling further.

3.7 VALIDATION AND PREDICTION

Validation of data is paramount. [33] Demonstrate the need to apply cross validation when selecting an optimal model. This is obtained by splitting the data set into two: Training set and Validation set. Usually the splitting ratio is 80:20 respectively [33]. In this research, validation set was 20% which acted as control experiment of the whole data set. A number of algorithms were applied to check the one which would give the best accuracy. These include; decision tree, Guasian Naïve Bayes, KNN, XGBoost, Gradient Boosting, SVM and random forest.

3.8 SOFTWARE TOOLS

A number of softwares were involved in order to carry out this research. For instance, Python, Jupyter notebook and pandas were used to explore the data [34]. Pycharm which is IDE (Integrated Development Environment) assisted in editing the code while programming. Python-Django was also used to provide user interface platform to predict and visualize the data.

CHEPTER FOUR: RESULTS AND DISCUSSIONS

4.0 INTRODUCTION

This chapter displays the results and discussions for the research. In this study of on enhancement of credit score prediction using data mining techniques, classification models are well appreciated in order to classify the clients which are eligible for loan and those that are not eligible. In the data set, Credit score group is classified as the output variable/ target variable with binary value of 0 and 1. The value 0 classifies the client who is eligible for loan while 1 classifies the non-eligible clients. The data set is imbalanced as only 27 % (15960) transactions are defaulters (ineligible) whereas the 73% (58096) are non-defaulters as shown in figure 4.0 below:

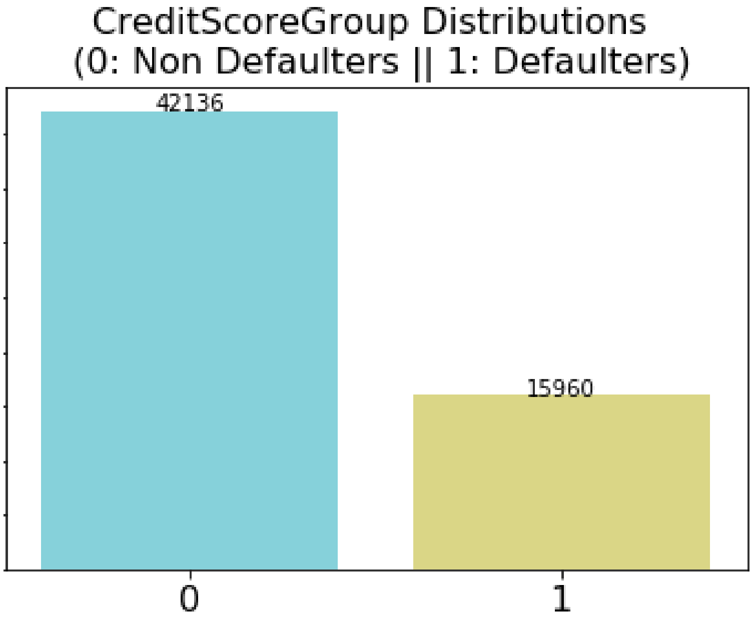


Fig 4.0: Bar graph showing the distribution of the credit score group.

Due to this disparity, misclassification problem is likely to occur during credit score prediction while classifying the customers. Machine learning algorithms are designed to perform with maximum accuracy and minimum error when the samples of every class are nearly equal. Therefore, to avoid this misclassification problem, data imbalance handling techniques should be applied in the dataset.

4.1 Data collection

The research explored and utilized secondary data from Bank of Kigali. The history of the clients of Bank of Kigali is well captured and private information like names are not used for security reasons. The data used contains 58096 transactions and 31 variables. Variables used to determine the eligibility of the customer in Bank of Kigali includes Age of the customer, Province, Customer branch, principal amount, paid interest, amount due, due fee, due tax, payment status, duration, returning customer, class and overdue days.

4.2 Data cleaning

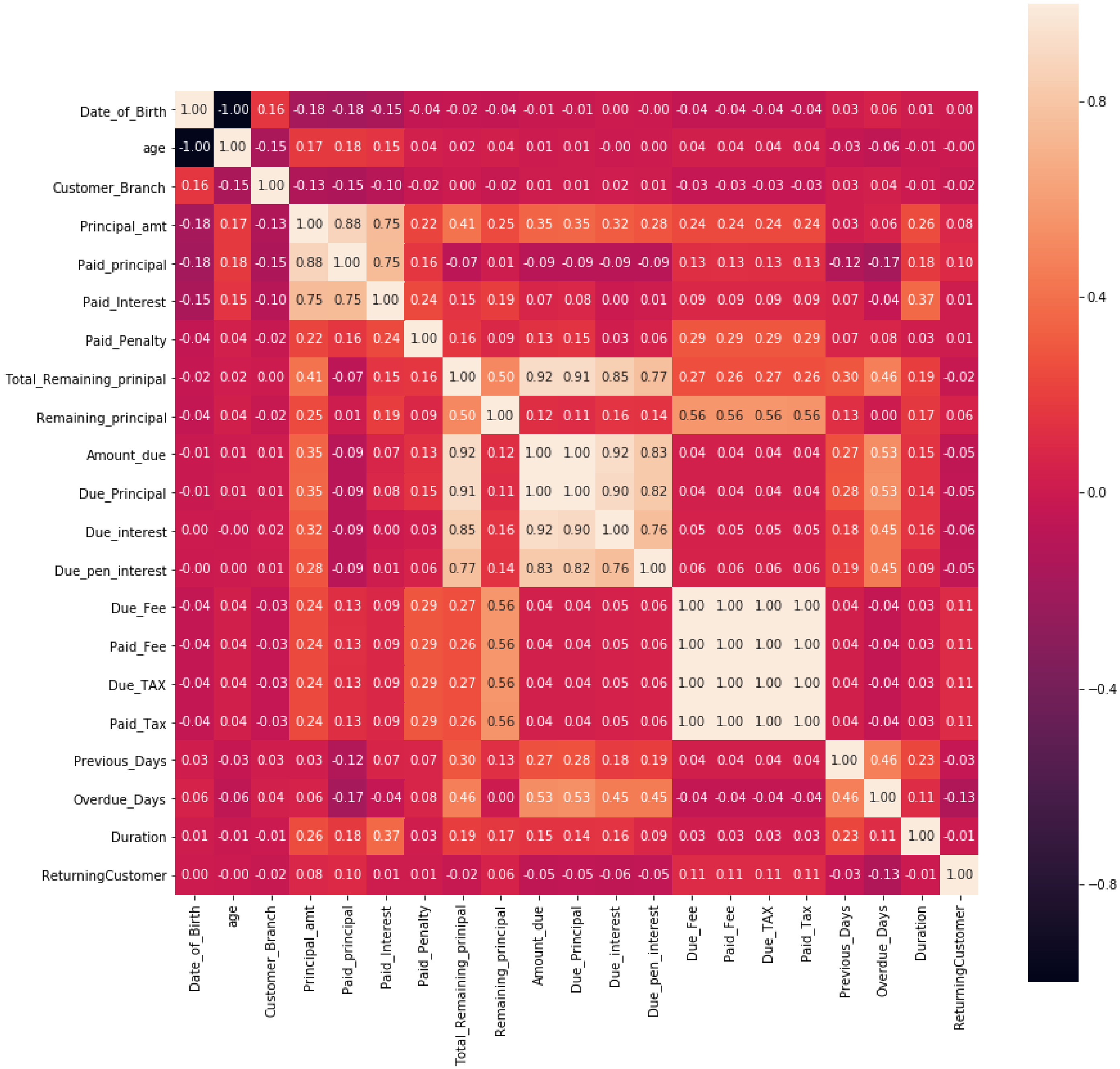
Data cleaning played a crucial role during research analysis. Before cleaning the data, loading the data set was initiated. Three variables had NAN (missing) values that is province, district and class. The two categorical variables province and district missing values were dropped whereas the missing values were imputed using KNN imputation by finding the k nearest neighbors of the row with missing values based on the non-missing values.

Apart from missing values in the dataset, unnecessary input variables were also dropped. For instance, unnamed variable in the data set was dropped since it does not add any impact to the dataset.

4.3 Data understanding and randomization

Data profiling is paramount at this stage where data set is reviewed, the structure is understood and interrelationships between the variables is established.

Exploratory data analysis (EDA) was carried out to comprehend the characteristics of data before proceeding with analysis using visuals. Data profiling was performed to establish the correlation between the variables which guide in masters the features which contribute most to the output variable.



	Date_of_Birth	age	Customer_Branch	Principal_amt	Paid_principal	Paid_Interest	Paid_Penalty	Total_Remaining_prinipal	Remaining_principal	Amount_due	Due_Principal	Due_interest	Due_pen_interest	Due_Fee	Paid_Fee	Due_TAX	Paid_Tax	Previous_Days	Overdue_Days	Duration	ReturningCustomer
Date_of_Birth	1.00	-1.00	0.16	-0.18	-0.18	-0.15	-0.04	-0.02	-0.04	-0.01	-0.01	0.00	-0.00	-0.04	-0.04	-0.04	-0.04	0.03	0.06	0.01	0.00
age	-1.00	1.00	-0.15	0.17	0.18	0.15	0.04	0.02	0.04	0.01	0.01	-0.00	0.00	0.04	0.04	0.04	0.04	-0.03	-0.06	-0.01	-0.00
Customer_Branch	0.16	-0.15	1.00	-0.13	-0.15	-0.10	-0.02	0.00	-0.02	0.01	0.01	0.02	0.01	-0.03	-0.03	-0.03	-0.03	0.03	0.04	-0.01	-0.02
Principal_amt	-0.18	0.17	-0.13	1.00	0.88	0.75	0.22	0.41	0.25	0.35	0.35	0.32	0.28	0.24	0.24	0.24	0.24	0.03	0.06	0.26	0.08
Paid_principal	-0.18	0.18	-0.15	0.88	1.00	0.75	0.16	-0.07	0.01	-0.09	-0.09	-0.09	-0.09	0.13	0.13	0.13	0.13	-0.12	-0.17	0.18	0.10
Paid_Interest	-0.15	0.15	-0.10	0.75	0.75	1.00	0.24	0.15	0.19	0.07	0.08	0.00	0.01	0.09	0.09	0.09	0.09	0.07	-0.04	0.37	0.01
Paid_Penalty	-0.04	0.04	-0.02	0.22	0.16	0.24	1.00	0.16	0.09	0.13	0.15	0.03	0.06	0.29	0.29	0.29	0.29	0.07	0.08	0.03	0.01
Total_Remaining_prinipal	-0.02	0.02	0.00	0.41	-0.07	0.15	0.16	1.00	0.50	0.92	0.91	0.85	0.77	0.27	0.26	0.27	0.26	0.30	0.46	0.19	-0.02
Remaining_principal	-0.04	0.04	-0.02	0.25	0.01	0.19	0.09	0.50	1.00	0.12	0.11	0.16	0.14	0.56	0.56	0.56	0.56	0.13	0.00	0.17	0.06
Amount_due	-0.01	0.01	0.01	0.35	-0.09	0.07	0.13	0.92	0.12	1.00	1.00	0.92	0.83	0.04	0.04	0.04	0.04	0.27	0.53	0.15	-0.05
Due_Principal	-0.01	0.01	0.01	0.35	-0.09	0.08	0.15	0.91	0.11	1.00	1.00	0.90	0.82	0.04	0.04	0.04	0.04	0.28	0.53	0.14	-0.05
Due_interest	0.00	-0.00	0.02	0.32	-0.09	0.00	0.03	0.85	0.16	0.92	0.90	1.00	0.76	0.05	0.05	0.05	0.05	0.18	0.45	0.16	-0.06
Due_pen_interest	-0.00	0.00	0.01	0.28	-0.09	0.01	0.06	0.77	0.14	0.83	0.82	0.76	1.00	0.06	0.06	0.06	0.06	0.19	0.45	0.09	-0.05
Due_Fee	-0.04	0.04	-0.03	0.24	0.13	0.09	0.29	0.27	0.56	0.04	0.04	0.05	0.06	1.00	1.00	1.00	1.00	0.04	-0.04	0.03	0.11
Paid_Fee	-0.04	0.04	-0.03	0.24	0.13	0.09	0.29	0.26	0.56	0.04	0.04	0.05	0.06	1.00	1.00	1.00	1.00	0.04	-0.04	0.03	0.11
Due_TAX	-0.04	0.04	-0.03	0.24	0.13	0.09	0.29	0.27	0.56	0.04	0.04	0.05	0.06	1.00	1.00	1.00	1.00	0.04	-0.04	0.03	0.11
Paid_Tax	-0.04	0.04	-0.03	0.24	0.13	0.09	0.29	0.26	0.56	0.04	0.04	0.05	0.06	1.00	1.00	1.00	1.00	0.04	-0.04	0.03	0.11
Previous_Days	0.03	-0.03	0.03	0.03	-0.12	0.07	0.07	0.30	0.13	0.27	0.28	0.18	0.19	0.04	0.04	0.04	0.04	1.00	0.46	0.23	-0.03
Overdue_Days	0.06	-0.06	0.04	0.06	-0.17	-0.04	0.08	0.46	0.00	0.53	0.53	0.45	0.45	-0.04	-0.04	-0.04	-0.04	0.46	1.00	0.11	-0.13
Duration	0.01	-0.01	-0.01	0.26	0.18	0.37	0.03	0.19	0.17	0.15	0.14	0.16	0.09	0.03	0.03	0.03	0.03	0.23	0.11	1.00	-0.01
ReturningCustomer	0.00	-0.00	-0.02	0.08	0.10	0.01	0.01	-0.02	0.06	-0.05	-0.05	-0.06	-0.05	0.11	0.11	0.11	0.11	-0.03	-0.13	-0.01	1.00

Fig 4.3: Data profiling table.

Fig 4.3 shows data profiling table as per correlation between the variables on every axis. The range of the correlation is from -1 and +1. The values which are around zero like returning customer and date of birth has no linear trend between them hence has no correlation therefore contribute less to output variable. On the other hand, when the correlation is closer to 1, their correlation is strong hence contribute strongly to the output variable. For instance, in the variables such as amount due and due principal, the value is actually 1 hence definitely they are strongly correlated. Other variables in table varies between the range of -1 and 1.

4.4 Modelling, Machine learning algorithms and Evaluation metrics

Modelling of data imbalance was done in two levels. One is in data level whereby data was reconstructed using resampling techniques. Oversampling, under sampling, SMOTE (synthetic oversampling technique), and NearMiss sampling are some of techniques applied on data-level reconstruction. All the above methods use decision tree, random forest, naïve Bayes, support vector machine, K Nearest Neighbors, and XGBoost classifier to check their performance on credit score prediction.

The second level is the algorithm level especially in cost sensitive learning algorithm technique where the algorithms such as a decision tree, random forest, support vector machine and logistic regression were employed to classify the data without considering their state of nature.

Data validation was obtained by dividing the data set into the validation set and training set. The splitting ratio used is default ratio which is 80:20, that's training set taking 80% and validation set (test set) taking portion of 20%. Performance of the predictions of the algorithms is measured on the basis of precision score, recall score, F1_Score, F-Beta and accuracy as stated above. Credit score group is the target variable in the dataset.

The results of the study were geared to answer each of the three specific objectives of the research as follows:

Objective i: To assess the performance of imbalance data mining techniques on credit score prediction

The first objective of this research is to assess the performance of imbalance data mining techniques on credit score prediction. These techniques include oversampling, under sampling, SMOTE, NearMiss and cost sensitive learning algorithms. The results of the balanced data were tabulated and to check their performance, credit score prediction is applied whereby machine learning models are used to assess the extent of their performance. To answer this objective, imbalance data was subjected to each technique above and machine learning algorithms were used to assess the extent of their performance on credit score prediction in terms of efficiency using accuracy performance metric. The techniques applied are discussed below:

i) Under sampling Technique

This technique creates a balance in the data by minimizing the majority class size (non-defaulters) to level of minority class. This was done by maintaining the rare class samples (minority class) and selecting equal sample size randomly from majority class therefore a balanced new dataset is achieved. Scikit learn package was used to execute random under sampling technique as shown in fig 4.6

```
Random under-sampling:
1    15960
0    15960
Name: Credit score group
```

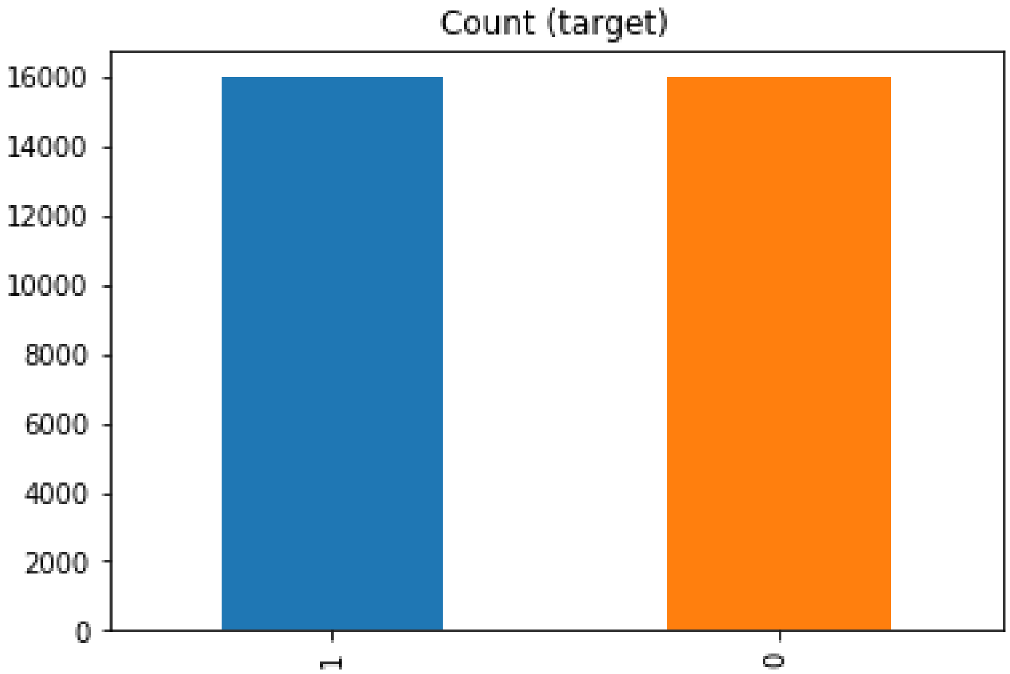


Fig 4.6 Random under sampling technique.

From fig 4.6, majority class (non-defaulters) have been randomly reduced to 15960 samples which is the same as minority class sample hence producing a balanced dataset. Therefore, applying any machine learning algorithm to perform credit score prediction there will be no biasness since there exist equality on both sides of the binary class. The below table clearly shows the performance of every model by evaluation metrics to check effectiveness and efficiency after balancing original data using random under sampling technique.

Table. 4.4 Model prediction on Under-Sampling Technique.

	Model	Precision score	Recall score	F1 score	F1 Beta	Accuracy
0	Logistic regression	0.792605	0.766707	0.762393	0.863724	0.767962
1	Gaussian NB	0.741446	0.619776	0.566919	0.854816	0.622807
2	Random forest	0.901287	0.901281	0.901211	0.896338	0.901211
3	Decision tree classifier	0.857028	0.857050	0.857033	0.856551	0.857038
4	SVM	0.742241	0.616916	0.561973	0.855053	0.619987
5	KNeighbors classifier	0.797229	0.796606	0.796629	0.811239	0.796784
6	Gradient boosting	0.901469	0.901479	0.901420	0.897345	0.901420
7	XGB classifier	0.902600	0.902620	0.902569	0.899086	0.902569

From the table 4.4, machine learning models including logistic regression, Guassian Naïve Bayes, random forest, decision tree, support vector machine, k nearest Neighbors, gradient boosting and XGBoost were utilized to assess the performance of credit score prediction on under sampling technique. To achieve the first objective, efficiency of each model above was checked using accuracy and from the results the best model is XGBoost Classifier with an accuracy of 0.902569. The other models Logistic regression, gaussianNB, SVM, K nearest Neighbors, decision tree, Gradient Boosting and random forest had an accuracy varying between 0.62 and 0.901420.

ii) Oversampling Technique

This technique tries to balance disparity in data set by adding more samples of the minority class in order to merge with majority class. In the data set, majority class sample is 42136 therefore instances are picked randomly from the minority class with replacement till the desired balance of data samples (42136) is achieved on the minority side. The balance data is thereafter used to tune the credit score group class distribution. This technique if not handled well can lead to overfitting of minority class hence leading to increased generalization error.

Random over-sampling:

```
1    42136
0    42136
Name: credit score group
```

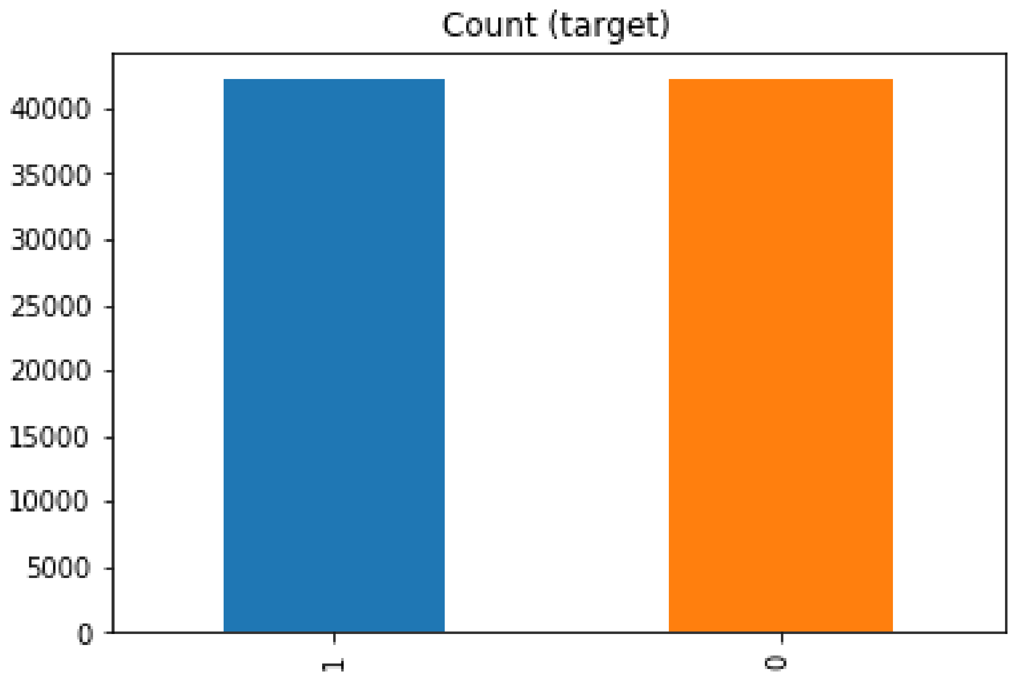


Fig 4.8: Over-sampling technique

From fig 4.8, instances have been picked randomly from minority class with replacement till desired balance data samples of 42136 is achieved on the minority side (defaulters). Due to equality achieved on the dataset, applying any machine learning algorithm to perform prediction will not be biased since there exist equality on both sides of the binary class.

The performance of the oversampling technique is measured based on F_Beta, precision score, recall score, F1_Score and accuracy so as to check the effectiveness and efficiency of the model as shown in table 4.5

Table 4.5 Model prediction on Oversampling Technique

	Model	Precision score	Recall score	F1 score	F1 Beta	Accuracy
0	Logistic regression	0.797207	0.773728	0.770396	0.865072	0.775319
1	Gaussian NB	0.744801	0.625464	0.576091	0.856812	0.629487
2	Random forest	0.964972	0.963947	0.964133	0.975661	0.964165
3	Decision tree classifier	0.949002	0.947047	0.947293	0.964796	0.947375
4	SVM	0.755836	0.668006	0.639161	0.861248	0.671373
5	KNeighbors classifier	0.841081	0.836193	0.836132	0.873659	0.836844
6	Gradient boosting	0.910055	0.910019	0.909878	0.902552	0.909878
7	XGB classifier	0.927421	0.927464	0.927434	0.926461	0.927440

From the table 4.5, machine learning models such as logistic regression, guassian naïve bayes, random forest, decision tree, support vector machine, K Nearest Neighbors, gradient boosting and XGBoost were used to assess the performance of credit score prediction on oversampling technique. From the results, the accuracy obtained by Random Forest algorithm is 0.964165 which is the best as compared to the other algorithms; logistic regression, gaussian naïve bayes, support vector machine, decision tree classifier, k nearest Neighbors, gradient Boosting and XGBoost classifier which have an accuracy that varies between 0.61 and 0.95

iii) SMOTE (Synthetic Minority Oversampling Technique)

This technique increases the number of minority class by generating the synthetic training samples using linear interpolation for the minority class. SMOTE applies neighbor's algorithm to generate new and synthetic data that is used in the training model. Minority class is actually over-sampled by systematically taking minority samples and introducing synthetic samples along the line segments minority nearest neighbors. The K nearest neighbors are chosen randomly. On SMOTE technique, standard scaler is used. Standard scaler assists by scaling each feature in the train data set to a given range and at the same time translates each feature individually in every given range for instance in credit score group variable, the range is between zero and one. SMOTE technique uses scikit learn library package to synthesize the datasets and Standard scaler tuner package from the same scikit learn library.

Table. 4.6 Model prediction on SMOTE on Technique

	Model	Precision score	Recall score	F1 score	F1 Beta	Accuracy
0	Logistic regression	0.849648	0.849265	0.849042	0.862192	0.849066
1	Gaussian NB	0.740157	0.626477	0.574145	0.851712	0.623139
2	Random forest	0.941143	0.941135	0.941139	0.940271	0.941145
3	Decision tree classifier	0.906720	0.906748	0.906729	0.907350	0.906734
4	SVM	0.875780	0.875401	0.875469	0.866561	0.875527
5	KNeighbors classifier	0.906998	0.907029	0.906971	0.910481	0.906971
6	Gradient boosting	0.925007	0.924992	0.924999	0.923746	0.925007
7	XGB classifier	0.936721	0.936664	0.936686	0.934389	0.936695

From the table 4.6, machine learning models such as logistic regression, Guassian Naïve Bayes, random forest, decision tree, support vector machine, K Nearest Neighbors, gradient boosting and XGBoost were used to assess the performance of credit score prediction on SMOTE technique. From the results, the accuracy obtained by Random Forest algorithm (0.941145) is the best as compared to the other algorithms; Logistic regression, gaussianNB, decision Tree classifier, K nearest Neighbors, Support vector machine, gradient Boosting and XGBoost Classifier which have an accuracy that varies between 0.62 and 0.94.

iv) NearMiss Technique

NearMiss is a type of under-sampling technique that aims to create a class distribution balance in the data set by randomly eliminating majority class samples. This technique therefore create space between two classes by completely removing instances of majority class when they are different and close to one another.

Table. 4.7 Model prediction on NearMiss Technique

	Model	Precision score	Recall score	F1 score	F1 Beta	Accuracy
0	Logistic regression	0.700584	0.700585	0.700584	0.703073	0.700606
1	Gaussian NB	0.651106	0.566492	0.494064	0.225623	0.563283
2	Random forest	0.879751	0.879648	0.879677	0.883933	0.879699
3	Decision tree classifier	0.826235	0.826259	0.826230	0.824655	0.826232
4	SVM	0.247859	0.500000	0.331425	0.000000	0.495718
5	KNeighbors classifier	0.716810	0.716771	0.716684	0.710421	0.716688
6	Gradient boosting	0.874976	0.874568	0.874630	0.883916	0.874687
7	XGB classifier	0.888057	0.888031	0.888042	0.890071	0.888053

From the table 4.7, machine learning models such as logistic regression, Guassian Naïve Bayes, random forest, decision tree, support vector machine, K Nearest Neighbors, gradient boosting and XGBoost models were used to assess the performance of credit score prediction on NearMiss technique. The results shows that XGBoost model has an accuracy (0.888053) which is the best as compared to other models. The rest such as Logistic regression, gaussianNB, decision Tree, K nearest Neighbors, Support vector machine, Gradient Boosting and Random Forest obtained an accuracy that varies between 0.495718 and 0.879699.

v) Cost sensitive learning

Cost sensitive learning algorithm is another technique applied in this research. Naïve Bayes, Random forest, decision tree classifier, logistic regression and Ridge classifier are some of the algorithms that penalizes mistakes in the sample data sets. This technique applies class weight approach to penalize mistakes in both classes[i] (majority and minority) with class weight[i] instead of 1. Therefore the higher class-weight means more emphasis on that class. Hence, if class 0 is more than class 1, then class weight of class 1 should be increased to match up with class 0 and if class weight doesn't sum to 1, then it will change regularization parameter.

Table 4.8 Model prediction on Cost sensitive learning Technique

	Model	Precision score	Recall score	F1 score	F1 Beta	Accuracy
0	Logistic regression	0.787704	0.731963	0.751160	0.568112	0.819324
1	Gaussian NB	0.653653	0.628883	0.471496	0.710032	0.473579
2	Random forest	0.904186	0.892532	0.898097	0.839364	0.919846
3	Decision tree classifier	0.851315	0.852762	0.852033	0.787359	0.881691
4	SVM	0.362385	0.500000	0.420212	0.000000	0.724769

From the table 4.8, machine learning models including logistic regression, Guassian Naïve Bayes, random forest, decision tree, support vector machine models were used to assess the performance of credit score prediction in Cost sensitive learning technique. From the results, the accuracy obtained Random Forest (0.919846) is the best as compared to the other algorithms; Logistic regression, gaussianNB, support vector machine, decision Tree classifier which have an accuracy that varies between 0.473579 and 0.89

Objective ii: To determine the performance of each resampling technique on credit score prediction.

According to objective two of this research; to determine the performance of each resampling technique on credit score prediction, we determine the level of performance of each technique in terms of efficiency using accuracy on credit score prediction.

From the results, the accuracy obtained by oversampling technique as shown in table 4.5 emerged to be the best with an accuracy of 0.964165 using the Random Forest model. It was followed closely by SMOTE technique with an accuracy of 0.941145 using Random Forest model as observed form table 4.6. The third ranked technique was the cost sensitive learning technique which obtained an accuracy of 0.919846 based also on Random Forest model as shown in table 4.8. This was followed closely by under sampling technique with an accuracy of 0.902569 based on the XGBoost model as seen on table 4.4. Finally, the worst technique is NearMiss Technique with an accuracy of 0.888053 as observed from results in table 4.7. This was based on the XGBoost model.

Objective iii: To evaluate effectiveness of each machine learning algorithm using performance metrics on credit score prediction

To answer the third objective of this research; to evaluate effectiveness of each machine learning algorithm using performance metrics on credit score prediction, it was fair enough to check effectiveness of each model applied. To check effectiveness of these models, each algorithm was evaluated based on F1_Beta, F1_score, precision score, and recall score across all the techniques and each shows different results.

On under sampling technique, the results on table 4.4 shows that XGBoost Classifier was the most effective model by obtaining the highest F1_Beta of 0.899086. Moreover, on precision score, recall score and F1_Score, XGBoost Classifier had the highest value of 0.902600, 0.902620 and 0.902569 respectively. Therefore, we can conclude that XGBoost Classifier performs well also in terms of effectiveness on under-sampling technique as compared to other models.

On the basis of oversampling technique, the results on table 4.5 shows that Random Forest classifier was the most effective model with highest F1_Beta of 0.975661. Moreover, on precision score, recall score and F1_Score, Random Forest had the highest values of 0.964972, 0.963947 and 0.964133 respectively. Therefore, we can conclude that Random Forest performed better as compared to other models on oversampling technique in terms of model effectiveness.

SMOTE technique was also intervened. The results from table 4.6 shows that Random Forest obtained the highest F1_Beta of 0.940271. Moreover, on precision score, recall score and F1_Score, Random Forest had the highest value of 0.941143, 0.941135 and 0.941139 respectively as compared to other models. Therefore, we can conclude that Random Forest was the most effective model on SMOTE technique as compared to the rest.

According to NearMiss Technique, the results obtained from table 4.7 shows that, XGBoost Classifier obtained the highest F1_Beta of 0.890071. Moreover, on recall score, precision score and F1_Score, XGBoost Classifier had the highest value of 0.888031, 0.888057 and 0.888042 respectively. Therefore, we can conclude that XGBoost Classifier is the most effective algorithm on NearMiss resampling technique as compare to other models.

From the results obtained on Table 4.8 Model prediction on cost sensitive learning technique, Random Forest obtained the highest F1_Beta of 0.839364. Moreover, on precision score, recall score and F1_Score, Random Forest had the highest value of 0.904186, 0.892532 and 0.839364 respectively. Therefore, we can conclude that Random Forest is the most effective model in cost sensitive learning technique as compared to rest.

ROC curve

Receiver operating characteristics curve is one of evaluation metrics used to access the performance of classifiers in this research. ROC curve tells the capability of the model to distinguish between the classes where in. It is usually plotted with true positive rate (TPR) against false positive rate (FPR). An excellent model has ROC near to 1 therefore means it has a good measure of separability while at the same time poor model has ROC curve near to 0 which means it has the worst measure of separability. At the same time when ROC measurement is 0.5, it implies that the model has no capability of class separation. It is just similar as flipping a coin.

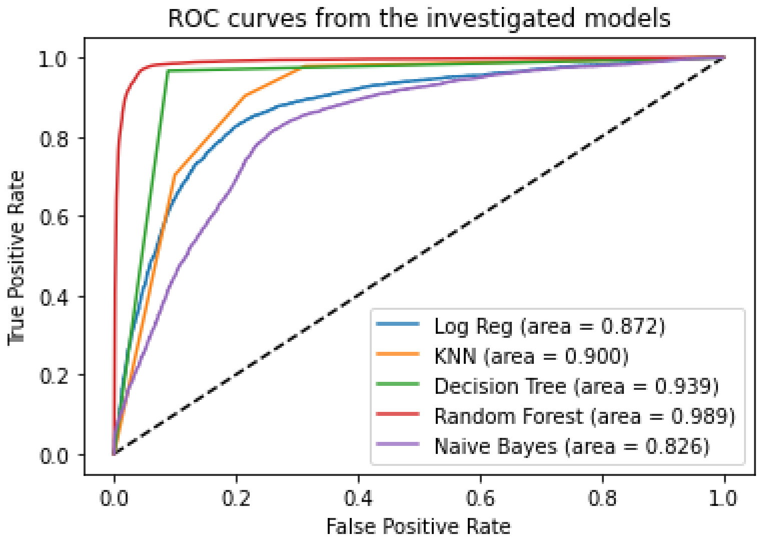


Fig. 4.9 ROC curve on the algorithm.

From the fig 4.9, the ROC curve obtained by Random Forest (0.989) is the excellent as compared to other algorithms; Logistic regression, KNN, decision Tree classifier, Naïve Bayes which have an ROC curve that varies between 0.82 and 0.94. This means random forest has a good measure of separability between the classes hence the best model as compared to the rest.

vi) Interpretation of Findings

Interpretation of these findings come in two categories, efficiency and effectiveness of the data imbalance techniques and the algorithms used. We evaluate the efficiency of all the classifiers on the basis of its accuracy while concurrently checking how effective the model is by comparing performance metrics based on F_Beta, F1_Score, recall score and precision score. In both cases (efficiency and effectiveness) shows how perfect the model is if its performance is high (closer to 1)

Therefore, on the results obtained above, every data imbalance handling technique performs differently due to the nature of data set. From model prediction on Under-sampling Technique in table 4.4, the accuracy obtained by XGBoost Classifier algorithm was (0.902569) which was the best as compared to other models; Logistic regression, gaussianNB, decision Tree classifier, K nearest Neighbors, Support vector machine, Gradient Boosting and Random Forest which had an accuracy that varies between 0.62 and 0.901420.

Using the F1_Beta to check model effectiveness on credit score prediction, XGBoost Classifier obtained the highest F1_Beta of 0.899086. Moreover, on precision score, recall score and F1_Score, XGBoost Classifier had the highest value of 0.902600, 0.902620 and 0.902569 respectively. . Therefore, we can conclude that XGBoost Classifier performed well on under-sampling technique as compared to other models.

Oversampling technique has the highest performance compared to the rest of data imbalance handling technique. From table 4.5, applying different algorithms on the oversampling technique, the accuracy obtained by Random Forest algorithm (0.964165) is the best as compared to the other algorithms; Logistic regression, gaussianNB, decision Tree classifier, K nearest Neighbors, Support vector machine, gradient Boosting and XGBoost Classifier which have an accuracy that varies between 0.61 and 0.95.

Using the F1_Beta, Random Forest obtained the highest precision score of 0.975661. Moreover, precision score, on recall score and F1_Score, Random Forest had the highest value of 0.964972, 0.963947 and 0.964133 respectively. Therefore, we can conclude that Random Forest is the best algorithm on oversampling technique as compared to the rest.

SMOTE technique was the second best technique. From the results from table 4.6 , accuracy obtained by Random Forest algorithm was (0.941145) which was the best as compared to the other algorithms; Logistic regression, gaussianNB, decision Tree classifier, K nearest Neighbors, Support vector machine, gradient Boosting and XGBoost Classifier which have an accuracy that varies between 0.62 and 0.94.

Using the F1_Beta, Random Forest obtained the highest precision score of 0.939943. Moreover, on precision score, recall score and F1_Score, Random Forest had the highest value of 0.941143, 0.941135 and 0.941139 respectively. Therefore, we can conclude that Random Forest perform well on SMOTE technique as compared to other models.

NearMiss technique had the least performance compared to the rest of techniques. From table 4.7, the highest accuracy obtained by XGBoost Classifier algorithm was (0.888053) which was the best as compared to the other algorithms; Logistic regression, gaussianNB, decision Tree classifier, K nearest Neighbors, Support vector machine, Gradient Boosting and Random Forest which have an accuracy that varies between 0.495718 and 0.879699.

Using the F1_Beta, XGBoost Classifier obtained the highest F1_Beta of 0.890071. Moreover, on recall score, precision score and F1_Score, XGBoost Classifier had the highest value of 0.888031, 0.888057 and 0.888042 respectively. Therefore, we can conclude that XGBoost Classifier is the best algorithm on NearMiss resampling technique.

Last but not least was cost sensitive learning technique. From fig 4.8 , the accuracy obtained by Random Forest without resampling the data was (0.919846) which was the best as compared to the other algorithms; Logistic regression, gaussianNB, decision Tree classifier which have an accuracy that varies between 0.473579 and 0.89.

Using the F1_Beta, Random Forest obtained the highest F1_Beta of 0.839364. Moreover, on precision score, recall score and F1_Score, Random Forest had the highest value of 0.904186, 0.892532 and 0.839364 respectively. Therefore, we can conclude that Random Forest is the best classifier in cost sensitive learning technique as compared to other models.

Therefore from the results obtained from the five techniques, oversampling technique outperformed other techniques in balancing the data of bank of Kigali. In addition, random forest algorithm as outlined on equation (15)

$$RFfi_i = \frac{\sum_{j \in all\, trees} norm\ fi_{ij}}{T} \text{ ………………………………….. equation (15)}$$

outperformed other algorithms with accuracy of 96.4165 % and F1_Beta of 97%. This is clear indication that random forest classifier works best with oversampling technique in predicting credit score of the customers.

CHEPTER FIVE: SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.0 SUMMARY

From the findings, oversampling technique is placed at the top as a well suited technique for handling imbalanced data set of bank of Kigali. This is evident with its efficiency and effectiveness using accuracy, F1_Beta, precision score, recall score and F1_score. Applying this technique helps to correct misclassification of customers in the bank of Kigali hence reducing unnecessary lost witness due to leakage of information caused by traditionally technique in curbing misclassification and insufficient metrics to check and predict the eligibility of the customers.

5.1 CONCLUSION

To analyze financial institution data (Bank of Kigali), various data mining approaches and machine learning methods are available. To build accurate and computer efficient classifiers for financial institution applications is challenging in data mining and machine learning areas. In this study, we employed five types of data imbalance handling techniques which includes, under sampling, oversampling, SMOTE, NearMiss and a number of machine learning algorithms which includes decision tree, logistic regression, random forest, support vector machine (SVM), GuassianNB, KNN classifier, Gradient Boosting, XGBoosting classifier. Effectiveness and efficiency of the applied algorithms was checked using evaluation metrics such as precision score, recall score, F1 score, F1_Beta, ROC curve and Accuracy.

Of the five imbalance handling technique, Oversampling outperform other imbalance technique by producing the best algorithm with 96.42% accuracy. Therefore in conclusion, Oversampling technique work best in reconstructing imbalance dataset of bank of Kigali. From the evaluation metrics in this research, it is evident that random forest algorithm outshines other classifiers with its efficiency and effectiveness of 96.42% accuracy, F1_Beta of 97.5661 and ROC of 98.9%.

5.2 RECOMMENDATIONS

From the findings in this research, it is evident that data needs to be balanced to avoid misclassification. In the case of bank of Kigali, the best data imbalance handling technique we do recommend is oversampling technique. This has been proven by several data mining and machine learning algorithms applied in this research which rightfully place oversampling as the suitable technique for automating credit operations in bank of Kigali. Furthermore, random forest algorithm came as the best classifier for predicting credit score with its efficiency measure accuracy of 96.42% and its efficiency performance of F1_Beta of 97% with ROC curve of 98.9%.

REFERENCES

[1] V. García, A. I. Marqués, and J. S. Sánchez, “Improving risk predictions by preprocessing imbalanced credit data,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7664 LNCS, no. PART 2, pp. 68–75, 2012, doi: 10.1007/978-3-642-34481-7_9.

[2] I. Ben-gal, “Outlier detection Irad Ben-Gal Department of Industrial Engineering,” *Res. gate*, no. November, pp. 0–11, 2014, doi: 10.1007/0-387-25465-X.

[3] S. F. Crone and S. Finlay, “Instance sampling in credit scoring: An empirical study of sample size and balancing,” *Int. J. Forecast.*, vol. 28, no. 1, pp. 224–238, 2012, doi: 10.1016/j.ijforecast.2011.07.006.

[4] N. V Chawla, K. W. Bowyer, and L. O. Hall, “SMOTE : Synthetic Minority Over-sampling Technique,” vol. 16, pp. 321–357, 2002.

[5] T. Boyle, “Dealing with Imbalanced Data,” *towards data science*, Feb. 04, 2019. https://towardsdatascience.com/methods-for-dealing-with-imbalanced-data-5b761be45a18 (accessed Jun. 10, 2020).

[6] V. Ganganwar, “An overview of classification algorithms for imbalanced datasets,” *Int. J. Emerg. Technol. Adv. Eng.*, vol. 2, no. 4, pp. 42–47, 2012, [Online]. Available: http://www.ijetae.com/files/Volume2Issue4/IJETAE_0412_07.pdf.

[7] et al Tyagikartik, “ML | Handling Imbalanced Data with SMOTE and Near Miss Algorithm in Python - GeeksforGeeks.” https://www.geeksforgeeks.org/ml-handling-imbalanced-data-with-smote-and-near-miss-algorithm-in-python/ (accessed Jun. 11, 2020).

[8] R. Pierre, “Detecting Financial Fraud Using Machine Learning: Winning the War Against Imbalanced Data,” Jun. 2018. .

[9] J. Novaković, P. Strbac, and D. Bulatović, “Toward optimal feature selection using ranking methods and classification algorithms,” *Yugosl. J. Oper. Res.*, vol. 21, no. 1, pp. 119–135, 2011, doi: 10.2298/YJOR1101119N.

[10] A. Namvar, M. Siami, F. Rabhi, and M. Naderpour, “Credit risk prediction in an

imbalanced social lending environment," *Int. J. Comput. Intell. Syst.*, vol. 11, no. 1, pp. 925–935, 2018, doi: 10.2991/ijcis.11.1.70.

[11] M. Zięba and J. Świątek, "Ensemble classifier for solving credit scoring problems," *IFIP Adv. Inf. Commun. Technol.*, vol. 372 AICT, no. i, pp. 59–66, 2012, doi: 10.1007/978-3-642-28255-3_7.

[12] H. Minh, "How to Handle Imbalanced Data in Classification Problems," Oct. 2018. .

[13] EY, "IFRS 9 Expected Credit Loss - Making sense of the transition impact," p. 18, 2018, [Online]. Available: https://www.ey.com/Publication/vwLUAssets/ey-ifrs-9-expected-credit-loss/$File/ey-ifrs-9-expected-credit-loss.pdf.

[14] H. J. Johnson, "Prospect theory in the commercial banking industry," *J. Financ. Strateg. Decis.*, vol. 7, no. 1, pp. 73–89, 1994, doi: http://dx.doi.org/10.1108/eb028702.

[15] A. I. Marqués, V. García, and J. S. Sánchez, "On the suitability of resampling techniques for the class imbalance problem in credit scoring," *J. Oper. Res. Soc.*, vol. 64, no. 7, pp. 1060–1070, 2013, doi: 10.1057/jors.2012.120.

[16] S. Kalid, K. C. Khor, K. H, and C. Y. Ting, "Effective Classification for Unbalanced Bank Direct Marketing Data with Over-sampling," vol. i, no. August, pp. 12–15, 2014.

[17] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder, and N. Seliya, "A survey on addressing high-class imbalance in big data," *J. Big Data*, vol. 5, no. 1, 2018, doi: 10.1186/s40537-018-0151-6.

[18] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 20–29, 2004, doi: 10.1145/1007730.1007735.

[19] Y. Yan *et al.*, "Oversampling for Imbalanced Data via Optimal Transport," *Proc. AAAI Conf. Artif. Intell.*, vol. 33, pp. 5605–5612, 2019, doi: 10.1609/aaai.v33i01.33015605.

[20] K. Kennedy, B. Mac Namee, and S. J. Delany, "Learning without default: A study of one-class classification and the low-default portfolio problem," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6206 LNAI, pp. 174–187, 2010, doi: 10.1007/978-3-642-17080-5_20.

[21] A. W. C. Faria, C. L. de Castro, and A. de P. Braga, “A New Oversampling-Based Approach for Class Imbalance Problem,” no. i, pp. 1–6, 2016, doi: 10.21528/cbic2013-020.

[22] H. Duan, Y. Wei, P. Liu, and H. Yin, “A novel ensemble framework based on K-means and resampling for imbalanced data,” *Appl. Sci.*, vol. 10, no. 5, 2020, doi: 10.3390/app10051684.

[23] New Times Daily, “Non-performing loans drop to 5.6 per cent | The New Times | Rwanda,” Aug. 22, 2019.

[24] N. Bank, “ANNUAL FINANCIAL STABILITY REPORT,” no. June, 2019.

[25] H. Wang, Q. Xu, and L. Zhou, “Large unbalanced credit scoring using lasso-logistic regression ensemble,” *PLoS One*, vol. 10, no. 2, pp. 1–20, 2015, doi: 10.1371/journal.pone.0117844.

[26] X. Guo, Y. Yin, C. Dong, G. Yang, and G. Zhou, “On the class imbalance problem,” *Proc. - 4th Int. Conf. Nat. Comput. ICNC 2008*, vol. 4, no. October 2008, pp. 192–201, 2008, doi: 10.1109/ICNC.2008.871.

[27] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, “RUSBoost: A hybrid approach to alleviating class imbalance,” *IEEE Trans. Syst. Man, Cybern. Part ASystems Humans*, vol. 40, no. 1, pp. 185–197, 2010, doi: 10.1109/TSMCA.2009.2029559.

[28] W. S. B. Suad A, Alasadi, “Review of Data Preprocessing Techniques in Data Mining,” *J. Eng. Appl. Sci.*, vol. 12, no. 16, pp. 4102–4107, 2017.

[29] R. Han, “The Math Behind Machine Learning ,” *Towards Data Science*, Oct. 2018. .

[30] J. Han, M. Kamber, and J. Pei, “Data Mining. Concepts and Techniques, 3rd Edition (The Morgan Kaufmann Series in Data Management Systems),” 2011.

[31] “1.10. Decision Trees ,” *scikit-learn 0.23.1 documentation*. .

[32] 1.10. Decision Trees, “1.10. Decision Trees — scikit-learn 0.23.1 documentation.” Accessed: Jun. 10, 2020. [Online]. Available: https://scikit-learn.org/stable/modules/tree.html#mathematical-formulation.

[33] M. S. Santos, J. P. Soares, P. H. Abreu, H. Araujo, and J. Santos, “Cross-validation for

imbalanced datasets: Avoiding overoptimistic and overfitting approaches [Research Frontier],” *IEEE Comput. Intell. Mag.*, vol. 13, no. 4, pp. 59–76, 2018, doi: 10.1109/MCI.2018.2866730.

[34] W. McKinney, “pandas: a Foundational Python Library for Data Analysis and Statistics,” *Python High Perform. Sci. Comput.*, no. January 2011, pp. 1–9, 2011.