ISSN 2723-6471

Impact of Sample Size on the Robustness of Machine Learning Algorithms for Detecting Loan Defaults Using Imbalanced Data

Boitumelo Tryphina Kobone[1], Tlhalitshi Volition Montshiwa[2,*]

[1,2]*Business Statistics and Operations Research, North West University, South Africa*

(Received: December 20, 2024; Revised: February 1, 2025; Accepted: April 5, 2025; Available online: July 10, 2025)

Abstract

This study aimed to assess the impact of sample size on the robustness of five machine learning classifiers: Support Vector Machine (SVM), Random Forest (RF), Naïve Bayes (NB), Decision Trees (DT), and K-Nearest Neighbour (K-NN). Although there are data-balancing techniques that aid in addressing data imbalance, they have some limitations which are discussed in this paper. The current study continues the trend in the application of these five ML classifiers for credit default detection, but it makes a contribution by examining whether sample size increment can better their performance when they are trained using a different imbalanced loan default dataset which has not been the focus of previous studies, although most ML algorithms are known to perform well when trained with large datasets. The study used a secondary loan default imbalanced dataset from Kaggle.com, where 85% of participants made loan payments and 15% defaulted. Stratified random sampling was used to select different sample sizes starting with 2% of the total observations, followed by 5%, then 10% up to 90% of the dataset, with the dependent variable being the stratum. The study found no consistent change in the classification metrics with the change in sample size, but RF and DT achieved 100% performance regardless of sample size and are therefore recommended as the most robust to data imbalance in loan default detection. The average classification metrics for NB and K-NN ranged from 72% to 92%, and SVM produced the lowest averages which were between 69% and 75%. NB, K-NN and SVM yielded poor sensitivity rates of 0% to 53%, indicating poor loan payments prediction but they had sensitivity scores in range of 84% to 86%, indicating good loan default classification. Future studies should consider other sampling methods, deep and hybrid learning methods with comparison to RF and DT.

Keywords: Machine Learning Classifiers, Imbalanced Data, Sample Size, Loan Default

1. Introduction

Data classification is a unique data mining technique whose objective is to determine the target class to which a specific object belong, and the results of a classification algorithm are generally related to data characteristics [1], such as the lack of density or information in the training data, the overlap between the classes, small disjuncts (disjuncts that classify few training samples), and noisy data which depend on the class imbalance [2]. In classification studies, the more powerful Machine Learning (ML) algorithms should be able to learn complex nonlinear relationships between input and output features and ML algorithms are robust to noise, show high variance which means that predictions vary based on the specific data used to train them [3].

One significant challenge in credit risk modelling is dealing with class imbalance, where the number of default instances is significantly smaller than the non-default instances [4], which is a prevalent occurrence in loan default datasets as less individuals default on their loan payments in comparison to those who made payments. In a binary-class problem, the minority class is also realized as the positive class whereas the majority class is the negative class [5]. [6] further stated that given the impracticality of manually processing massive volumes of data, ML is the most practical way to create classifiers for predicting loan default risks. Classifiers trained with an imbalance dataset tend to predict the majority class (frequently occurring) more accurately than the minority class [1].

According to [1], applying suitable sampling techniques such as oversampling, under sampling and Synthetic Minority Oversampling Technique (SMOTE) for reducing class imbalance issues can enhance the classifier's performance.

***Corresponding author: Tlhalitshi Volition Montshiwa (volition.montshiwa@nwu.ac.za)**
DOI: https://doi.org/10.47738/jads.v6i3.713
This is an open access article under the CC-BY license (https://creativecommons.org/licenses/by/4.0/).
© Authors retain all copyrights

However, these data balancing methods have limitations. SMOTE may bring noise and other problems [7] while [8] found that under sampling and Random Under-Sampling (RUS) worked well when the dataset was large, but there was a chance of losing valuable information while oversampling and random over-sampling can increase the possibility of overfitting the model and more complex computational computing when the size of data is large enough. A possibility of losing some instances from the dataset that can affect the accuracy of the model have also been noted [8].

These limitations of data balancing methods are the mains problem identified in this study which seeks to explore whether sample size increment can improve the ML algorithms' robustness to data imbalances for future studies to use sample size increment as one of the methods of dealing with data imbalances when using ML classifiers. In this study we used SMOTE-Tomek data balancing technique for comparison purposes to determine whether the impact of sample size variations when the data are balanced will be different from when the data are imbalanced, and whether sample size increment alone can improve the accuracy of the competing classifiers. Although the focus of the current study is theoretical since the study on the ML algorithms and how their robustness to class imbalances is impacted by the change in the sample size, the study also seeks to make a practical contribution to credit default classification. This study extends the application of ML in predicting loan default by comparing SVM, RF, NB, DT, and K-NN while focusing on identifying the sample size at which these classifiers become more robust to class imbalances, and to determine whether the sample size variations impact this robustness to class imbalance as well as identify the ultimate ML classifier for loan default prediction from the five.

2. Literature Review

According to [6], loan default happens when someone does not make their contractually required payments on time. One of the major problem banking sectors faces in this ever-changing economy is the increasing rate of loan defaults, and the banking authorities are finding it more difficult to correctly assess loan requests and tackle the risks of people defaulting on loans [9], which may result in the financial institutions experiencing significant financial losses when a defaulter is mistakenly classified as a non-defaulter during the default prediction process [6]. If an applicant defaults on the loan, the bank must act, which consumes time, energy, and money [6].

The commonly used classification algorithms from previous studies on loan default dataset or credit default prediction were RF in studies conducted by [6], [10], [11], [12] and [13], Logistic Regression (LR) in studies conducted by [6], [12], [13] and [14], DT in studies conducted by [6], [10], [11], [12] and [13], other classifiers employed were K-NN, NB, SVM, Neural Networks (NN) and XGBoost. The current study continues the trend in the application of RF, DT, SVM, NB and K-NN for credit default detection but it makes a significant contribution by examining whether sample size increment can better their performance when they are trained using a different imbalanced loan default dataset which has not been the focus of previous studies.

Previous studies such as the ones conducted by [8] and [15] showed that RF consistently achieve the highest accuracy across all contexts (different datasets), and that RF is unbiased. RF and DT classifiers are known for their ability to handle imbalanced datasets effectively by capturing complex decision boundaries, exhibiting robustness to class imbalance and handling both minority and majority classes well [4]. [4] also stated that DT algorithm is a popular method for loan default prediction due to its simplicity, interpretability and its ability to handle large datasets with high dimensionality. NB is a fast and space efficiency classifier which only requires a small amount of training data to estimate the parameters required for classification as [16] explains. [17] stated that K-NN is easy to implement and understand but has a major drawback of becoming significantly slow as the size of the data in use grows or increases. In addition, SVM had been found to have accuracy advantages at larger datasets as explained by [18] and [19] and has outperformed other classifiers in terms of accuracy on some studies [1].

Literature showed that most of the studies conducted focused only on either the effect of the sample size as in the studies conducted by [15], [18], [19], [20] and [21] or on the effect of imbalanced data as in the studies conducted by [1], [22] and [23], while other studies mainly sought to determine the performance of ML algorithms when predicting loan defaults with examples being [6], [10], [11], [12] and [13] but not all of these issues in one study. This is therefore a gap in literature around ML classifiers. As such, the interest of the current study is on both sample size variations, imbalanced dataset as well as the prediction on loan defaults.

 ISSN 2723-6471

Literature shows that the use of SMOTE-Tomek to balance the data has some advantages over standalone under sampling and oversampling methods hence it was preferred to be used for comparison purposes in the current study. [24] explain that under sampling assists in reducing the majority group by randomly eliminating cases or observations from the majority class but although it is advantageous since it can improve run time and address storage problems, under sampling may eliminate important data potentially making the remaining data biased and to be unable to provide class distribution accuracy. [24] compared the performance of the standalone SMOTE and Tomek links to SMOTE-Tomek when used for balancing the data before implementing NB, K-NN and SVM and found that SMOTE- Tomek yielded improved performance across all classifiers. [25] also found that SMOTE-Tomek improved the results of RF compared to when the data is not balanced and when using oversampling through the standalone SMOTE. This literature supports our choice to use SMOTE-Tomek and not standalone under sampling and oversampling techniques in the current study which also implements the RF, NB, K-NN and SVM among the other classifiers that are evaluated in the current study.

3. Methodology

3.1. Data description

The study used the Anonymized Loan default dataset which were sourced from Kaggle.com and can be accessed through the following link http://www.kaggle.com/datasets/joebeachcapital/loan-default. The data comprises 38477 observations, and include variables such as the loan amount, term, interest rate, instalment, employment length, home ownership, annual income, loan status and purpose as repay_fail (which quantifies loan default). In this study, categorical independent variables will be converted to dummy variables in the data analysis phase since the ML classifiers used in the study require continuous or dummy features. In this study, the variable repay_fail is used as a target variable with repay_fail = 0 denoting loan payments and repay_fail = 1 representing loan defaults. The independent variables are Loan amount (loan_amnt), Loan Term (Term) indicating whether the term is 36 or 60 months, Loan Interest rate (int_rate), Loan Instalments (installment), Employment length (emp_length) starting from less than 1 year to 10 years or more with the not applicable option for partcipants who are not in employment, Annual income (annual_inc), Home Ownership (home_ownership) indicating whether the applicant is renting or having a mortgrage, and purpose of loan with options including car, debt consolidation and house improvement, to mention a few. The categorical variables were converted to dummy variables as a data preparation step for the ML classifiers.

Using stratified random sampling, various samples were selected from the 38477 observations starting with 2% to 90% of the data. These samples were drawn for experimental purposes to mimic a situation where there are different sample sizes of the same variables to enable the researchers to study how the ML classifiers perform when the sample size increases while the dependent variable is imbalanced. Each of the randomly selected samples from were then balanced using SMOTE-Tomek to achieve a ratio of loan defaults - to - loan payments of 50: 50, across all sample sizes. This data balancing is used for comparison purposes to determine whether the impact of sample size variations when the data are balanced will be different from when the data are imbalanced, and whether sample size increment alone can improve the accuracy of the competing classifiers.

The frequencies of the variable repay_fail at different sample sizes before and after implementing SMOTE-Tomek are shown in table 1. The table shows the frequency of the variable repay_fail at different sample sizes when the imbalance is kept constant across all the sample sizes and the degree of imbalance was kept constant for it not to bias the results. That is, the objective was to study how varying sample size impacts the performance of the ML classifiers and not to determine how the extent of imbalance affects the performance of the ML classifiers per say. Therefore, if the imbalance varied with each sample size, the impact of the extent of imbalance on ML classifiers would have brough another aspect that needed to be empirically tested which is beyond the scope of this study which was to determine the impact of sample size on the performance of ML classifiers when the data is imbalanced. It also shows that for all the samples, the majority class is the loan default class (repay_fail = 1).

Each of the randomly selected samples from the original data were balanced using SMOTE-Tomek to achieve a ratio of loan defaults - to - loan payments of 50: 50, across all sample sizes. This data balancing is used for comparison purposes to determine whether the impact of sample size variations when the data are balanced will be different from

 ISSN 2723-6471

when the data are imbalanced, and whether sample size increment alone can improve the accuracy of the competing classifiers. The frequencies of the variable repay_fail at different sample sizes after implementing SMOTE-Tomek are also shown in table 1.

Table 1. Frequency table for variable repay_fail by sample size

Original dataset (Before Balancing)				SMOTE-Tomek balanced dataset		
Total Sample size		**Sample size per each group of Repay_Fail**		**Total Sample size**	**Sample size per each group of Repay_Fail**	
		0	**1**		**0**	**1**
2%	770	653	117	**806**	403	403
5%	1924	1633	291	**1974**	987	987
10%	3848	3265	583	**3984**	1992	1992
20%	7695	6530	1165	**8108**	4054	4054
30%	11543	9795	1748	**12262**	6131	6131
40%	15391	13060	2331	**16310**	8155	8155
50%	19239	16325	2914	**20284**	10142	10142
60%	23086	19590	3496	**24740**	12370	12370
70%	26934	22855	4079	**28516**	14258	14258
80%	30782	26120	4662	**32724**	16362	16362
90%	34629	29385	5244	**36854**	18427	18427
100%	38477	32650	5827	**41242**	20621	20621

The data in this study was split into 70% training data and 30% validation data which is a commonly used training- to-validation data splitting ratio. The Statistical Package for Social Scientists (SPSS) version 26 was used for stratified random sampling and for running descriptive statistics for the data, specifically the frequency tables to show the distribution of the dependent variable across the different sample sizes. The main analysis, which is model training and evaluation, were done using the following packages in Python 3.12.2: matplotlib, imblearn and sklearn.

3.2. Methodology of ML Classifiers Used in the Current Study

3.2.1. SVM

SVM is a generalized linear supervised classifier that can perform binary classification on data, and its decision boundary is the maximum margin hyperplane that solves the learning sample [26]. The method can be employed for high dimensional data and generally leads to accurate classification when coping with small sample size in comparison to the other ML methods [27]. SVM draws margins between classes such that the distance between the margin and the classes is maximum hence minimizing the classification error [17]. The performance of SVM largely depends on the suitable selection of a kernel function that generates the dot products in the higher-dimensional feature space [28]. The SVM classifier is built from a training set of N samples which are described by:

$$(X_1,Y_1)\ldots(X_i,Y_i),\ldots(X_N,Y_N), \quad (1)$$

For n-dimensional space, input data belongs to class 1 or class 2 and the associated labels be -1 for class 1 and +1 for class 2 such that $y_i \epsilon \{-1,1\}$ [29]. If the input data can be separated linearly, the separation hyper plane can be shown by Equation 2. This equation finds a maximum margin to separate the positive class from negative class, explain [30], as cited by [29].

$$f(x)=w^T x+ b, \quad (2)$$

w is n-dimensional weight vector, b is scalar multiplier or bias value. The decision function is shown in Equation 3.

$$f(x)=sgn(w^T x+ b) \quad (3)$$

 ISSN 2723-6471

If two classes can be separated linearly, the hyper plane that satisfies maximum margin between two classes is found by solving the following [29]:

$$\text{Minimise} \quad \frac{1}{2}\|w\|^2+C\sum_{i=1}^{n}\varepsilon_i \tag{4}$$

with $\frac{1}{2}\|w\|^2$ being the margin between two lines $w^Tx+ b=1$ and $w^Tx+ b=-1$ and Equation 4 is maximised subject to:

$$y_i(w_ix_i+b)\geq 0. \tag{5}$$

When the parameters of SVM are well tuned, classification performance is increased [29]. SVM training is performed by solving the optimisation problem in Equation 6:

$$L(\alpha)=\sum_{i=1}^{k}\alpha_i-\frac{1}{2}\sum_{i,j=0}^{k}\alpha_i\,\alpha_j y_i y_j k(x_i x_j) \tag{6}$$

Subject to: $\sum_{i=1}^{k} y_i\alpha_i=0$, $\alpha_i\geq 0$ for i=1…. k, $k(x_ix_j)$ is kernel function, α_i are Lagrange multipliers.

When the data cannot be separated linearly, kernel function mapping changes according to Equation 7.

$$k(x_ix_j)=k(x_ix_j)+\frac{1}{C}\delta_{ij} \tag{7}$$

A restriction-delimiting parameter C (a parameter that controls the amount of penalty during the SVM optimisation) is used to control penalization when training instances are classified incorrectly [31]. For a high value of C, the SVM tends to generate a smaller margin at the risk of overfitting [31]. A small value of C results in more erroneous classifications at the expense of training precision [32]. The ability to apply new kernels rather than linear boundaries also increases the flexibility of SVMs for the decision boundaries, leading to a greater classification performance [28]. In a study conducted by [33], their results demonstrated that the gamma γ , C, class and weight values (w) were key hyperparameters that could be used to train the most optimal SVM model using the RBF kernel for imbalanced data.

Both hyperparameters work as inverse regularization terms. A large C will place emphasis on lowering the number of support vectors since each one of them contributes to the $\sum_{i=1}^{n}\varepsilon_i$ cost in the optimization. A lower C will allow more support vectors, resulting in larger margins [34]. The γ parameter controls how fast the "influence" of a point decreases with distance. The kernel value for two points will decrease as γ increases. As γ increases, the decision surfaces become more" curvy" and fit closely to the training data. A smaller γ will generate decision surfaces that are flatter, and thus a simpler model [34]. SVM with an RBF kernel is usually one of the best classification algorithms for most datasets, but it is important to tune the two hyperparameters C and γ to the data itself [34]. SVM has been found to have advantages of accuracy at larger datasets like in the studies by [35] and [36] so the authors of this research wanted to determine whether this will still be true when the dataset is imbalanced. However, SVM is known to be sensitive to imbalanced dataset and decision boundary have bias towards minority class [37]. Therefore, it was worth including in this study to enable the researchers to determine whether this bias may decrease as the sample size increases or not.

3.2.2. DT Classifier

DT classifier is a tree-based technique in which any path beginning from the root is described by a data separating sequence until a Boolean outcome at the leaf node is achieved as explained by [38], [39] and [40]. The tree consists of three types of nodes, a root node, child node (decision node) and leaf node (terminal node) [41]. Each node in a decision tree represents a feature in an instance to be classified, and each branch represents a value that the node can assume [42]. The methodology for deriving a DT classifier discussed in this subsection was sourced from [43] unless otherwise specified. A score measure is defined to evaluate each variable and select the best one at each split using Equation 8:

$$\text{score}(S.T)=I(S)-\sum_{i=1}^{p}\frac{N_i}{N}I(S_i), \tag{8}$$

 ISSN 2723-6471

T is the candidate node that splits the input sample of S with size N into p subsets of N_i $(i = 1, \dots, p)$ and $I(S)$ is the impurity measure of the output for a given S. Entropy, which is a measure of impurity or disorder in a set of data used to evaluate the quality of the split based on a certain feature, and Gini index which calculates the amount of probability of a specific feature that is classified incorrectly when selected randomly are two of the most popular impurity measures and are described in Equations 9 and 10:

$$I_{entropy}(S) = -\left(\frac{N_+}{N}\log\frac{N_+}{N}\right) - \left(\frac{N_-}{N}\log\frac{N_-}{N}\right) \quad (9)$$

$$I_{gini}(S) = \left[\frac{N_+}{N}\left(1 - \frac{N_+}{N}\right)\right] + \left[\frac{N_-}{N}\left(1 - \frac{N_-}{N}\right)\right], \quad (10)$$

N_+ represent the number of manipulated samples in each subset, and N represents the number of non-manipulated samples in each subset. In the current study the Gini index is used to calculate the amount of probability of a specific feature that is classified incorrectly when selected randomly. The process is repeated on the resulting nodes until it reaches a stopping criterion. That is, depending on the test outcome, the classification algorithm branches towards the appropriate child node where the process of test and branching repeat until it reaches the leaf node. The leaf or terminal nodes correspond to the decision outcomes [43].

The DT algorithm is a popular method for loan default prediction due to its simplicity, interpretability and its ability to handle large datasets with high dimensionality [44]. RF and DT classifiers are known for their ability to handle imbalanced datasets effectively by capturing complex decision boundaries and handling both minority and majority classes well [44]. The study by [44] also found that RF and DT classifiers outperformed other ML algorithms employed in their study, exhibiting robustness to class imbalance. Therefore, RF and DT are worth including in the current study, which is also about robustness of the ML classifiers to class imbalance.

3.2.3. NB Classifier

Naive Bayes is a probabilistic classifier which works based on the Bayes theorem to solve classification problems, by determining the probability of each feature occurring in each class and returning the most likely class. This classifier assumes that a particular feature in a class is not directly related to any other feature and that each feature makes an equal, individual contribution to the output, although features for that class could have interdependence among themselves [45]. Further stated that NB has a certain advantage over other classifiers as it requires only a small amount of training data. The Bayes rule is defined as follows [45]:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (11)$$

A and B represent class and features respectively, $P(A|B)$ represents the probability of belonging to class A with all given features of B(Likelihood), $P(B)$ denote the probability of all features used for normalisation (Predictor prior probability), $P(A)$is the class prior probability, and $P(B|A)$ represents the probability of belonging to B feature with all given classes of A. The stages of the NB algorithm in classifying datasets are as follows [46]: Read training data; Calculating probability in the following way; Calculates the average of each parameter with the following formula:

$$\mu = \frac{\sum_{i=1}^{n} x_i}{n} \quad (12)$$

μ represents mean x_i is the sample value i, and n is the number of samples.

Calculates the standard deviation of each parameter with the following formula:

$$\sigma^2 = \frac{1}{n-1}\sum(x_i - \mu)^2 \quad (13)$$

σ is the standard deviation, expresses the variance of all attributes, n is the amount of data in the same class, x_i is the value of attribute to i,μ is the mean.

 ISSN 2723-6471

Look for probability values using the following formula:

$$P(X_i = x_i|Y_i = y_i) = \frac{1}{\sqrt{2\pi\sigma_{ij}}} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}}, \quad (14)$$

σ is the standard deviation, which expresses the variance of all attributes, x_i is the value of attribute to i, μ : mean, X_i is attribute to i, $Y_i$is the class sought and, y_i is the Y sub-class searched.

Step 2 is repeated until the probability of all parameters is calculated. The calculation process will stop when the probability value of all parameters of each attribute has been calculated. NB is known to be a fast and space efficiency classifier which only requires a small amount of training data to estimate the parameters required for classification, as [47] explain. So, its inclusion in the current study was to enable the researcher to determine whether it will outperform the other classifiers under study for smaller sample sizes even when the data is imbalanced.

3.2.4. K-NN Classifier

The K-NN classifier is a simple algorithm that assigns the majority vote of k training samples that are most like the new sample [48]. The k is a limitation for adjusting the classification algorithms as stated in the study by [49]. The K-NN is commonly used with the Euclidean distance since the linear time complexity of the Euclidean distance $(O(n))$makes it an ideal choice for large datasets [50]. Therefore, the Euclidean distance is used in this study. The Euclidean distance is used to measure the distance between samples, and the inverse distance square weighting method is applied to calculate the sample weight [50]. The Euclidean distance is defined as follows [51]:

$$d = \sum_{i=0}^{k} (x_i - y_i)^2, \quad (15)$$

where k is the number of nearest neighbours, x_i is the data point, y_i is the neighbouring point and d is the distance.

The K-NN is implemented using data from an original sample class. K data is chosen, which is the closest neighbour to the new data to be decided which sample class it should be added. The distance of the new data to be included in any of the original sample class groups is taken from the data showing the K nearest neighbouring property [22]. The following are the steps followed in K-NN classification and are adapted from [51]. Consider the following to be the training set of data

$$\mathcal{L} = (X, Y)_{n\times(p+1)}, \quad (16)$$

$X_{n\times(p+1)}$ is a matrix with p features and n sample points and Y is a binary categorical response. Let $X^0_{1\times P}$ be a test sample point with p values and it is needed to predict the output class i.e. $\hat{Y}$ for $X^0_{1\times P}$. Suppose B bootstrap samples are drawn from the training data $\mathcal{L}$, each with a random subset of $p' \leq p$ features such that, $S^b_{n\times(p'+1)}$, where b=1,2,3,…,B and $X^0_{1\times p'}$ is a subset of p'≤p corresponding values from $X^0_{1\times P}$. The distance formula is given as:

$$\delta_b\left(X^{i-1}_{1\times p'}, X^i_{1\times p'}\right) min = \left[\sum_{j=1}^{p'} \left| X^{i-1}_j - X^i_j \right|^q\right]^{\frac{1}{q}}, \text{ i=1, 2, 3, …, k.} \quad (17)$$

In each base model, the distance formula given in Equation 16 is used to determine the sequence of distances as follows:

$\delta_b\left(X^0_{1\times p'}, X^1_{1\times p'}\right)$ min, $\delta_b\left(X^1_{1\times p'}, X^2_{1\times p'}\right)$ min, $\delta_b\left(X^2_{1\times p'}, X^3_{1\times p'}\right)$ min, . . . , $\delta_b\left(X^{k-1}_{1\times p'}, X^k_{1\times p'}\right)$ min

This sequence suggests that $X^i_{1\times p'}$ is the nearest observation to $X^{i-1}_{1\times p'}$ where, i=1, 2, 3, …, k. The corresponding response values of $X^1_{1\times p'}$, $X^2_{1\times p'}$, $X^3_{1\times p'}$, . . . , $X^k_{1\times p'}$ are y^1 , y^2 , y^3, . . . , y^k, respectively, and the predicted class of test point $X^0_{1\times P}$ for the b^{th} base model is $\hat{Y}^b$ is the majority vote of (y^1 , y^2 , y^3, . . . , y^k), where $b = 1, 2, 3, \ldots, B$. The final predicted class of the test observation $X^0_{1\times P}$ is $\hat{Y}$ is the majority vote of ($\hat{Y}^1$, $\hat{Y}^2$, $\hat{Y}^3$, . . ., $\hat{Y}^B$).

 ISSN 2723-6471

The final label of sample x_i is obtained by applying the following decision rule:

$$f(x_i)=\begin{cases} 1, & \text{and if } \sum_{c=1}^{n_{k(x_i)}} y_i \geq 0 \\ -1, & \text{and if } \sum_{c=1}^{n_{k(x_i)}} y_i < 0, \end{cases} \quad (18)$$

$n_{k(x_i)}$ expresses the indexes of the k-nearest observations. The most similar samples are calculated by using various distance algorithms. K-NN is known to have higher possibilities of observations from the majority class of an imbalanced dataset [37], so the current study ought to determine whether this is impacted by the sample size or not.

3.2.5. RF Classifier

RF is a set of tree classifiers $\{h(x,\theta_k),k=1,2\ldots., n\}$, where $h(x,\theta_k)$ determines the growth of each decision tree, and x is the input vector of the classifier [52]. Compared with other classifiers, RF has the advantages of being less prone to overfitting and reducing the impact of outliers, leading to higher accuracy of the classification in many studies [26], hence it is considered in this study. The following steps are followed in training the RF classifier [41]: Start by creating a combination of trees which each will vote for a class, then let k be the number of sampling groups, n_i and m_i be the number of data and variables in a group where i= 1, 2...k. Each sampling group is as follows: n_i data where $n_i \leq N$ are selected randomly from N. m_i variables where $m_i \leq M$ are selected randomly from M. A tree is grown and gives a prediction class. After Step 1 to 3 are repeated for k times, these trees become a forest. Then the classification will be selected by a majority vote of all trees in the forest [45].

RF is the algorithm of interest because based on previous studies, it is the most recommended, and its classification performance for imbalanced data was high and more accurate, irrespective of sample sizes, as compared to other classification algorithms, outperforming other supervised classification approaches even at large sample sizes as explained [8], [53] and [54]. That is, previous studies showed that RF consistently achieved the highest accuracy across all contexts (different datasets), and previous studies showed that RF is unbiased with examples of such studies being [8] and [54].

3.2.6. Methodology of SMOTE-Tomek

Hybrid resampling methods has been proposed as a more effective way to handle imbalanced data [55]. Hybrid sampling achieves an optimal balance by removing examples in the majority class and replicating some examples in the minority class as a result it neither loses too much information from the under-sampling process nor does it overfit the classifier through the over-sampling process [56]. Therefore, due to these advantages of a hybrid sampling algorithm, the current study will use SMOTE-Tomek hybrid sampling to balance the data to determine the impact of sample size variations on the performance of ML classifiers when data is balanced compared with the classification performance of the ML classifiers when the data is imbalanced.

According to [57] SMOTE-Tomek combines SMOTE with Tomek links under-sampling technique to balance data [58]. SMOTE is first used to oversample the dataset [56]. Then to create better-defined class clusters, Tomek links can be applied to the over-sampled training set as a data cleaning method [59]. As stated by [60], the Tomek links can be formulated in a binary classification task as follows: X_{maj} and X_{min} denote a majority class and minority class sample respectively, and $d(X_{maj},X_{min})$ denotes the distance between them. If there is no observation y , which is any X_{maj} or X_{min} such that $d(X_{maj},z)<d(X_{maj},X_{min})$ or $d(z,X_{min})<d(X_{maj},X_{min})$, then $d(X_{maj},X_{min})$ is called a Tomek link. SMOTE +TOMEK LINKS then corrects SMOTE data by finding pairs of minimally distanced nearest neighbours of opposite classes. It then identifies and removes Tomek links to produce a balanced dataset with well-defined classes [61].

3.3. Evaluation of the Classifiers

The classification performance for all the algorithms used in the current study was assessed using overall classification accuracy, sensitivity/recall, specificity, precision and F1-score. These metrics have been frequently used to predict loan defaults by previous studies such as those by [14], [10], [11] and [13], while studies such as [1], [62] and [63] used

 ISSN 2723-6471

them on different datasets. The higher the value of the classification metrices, the better the performance of the model [8].

The five-classification metrics are computed from the confusion matrix. A confusion matrix is useful in computing model recall or sensitivity, specificity, accuracy and precision [64]. Considering the “Loan default” class as a (Positive) and “Loan payment” class being a (negative), the True Positives (TP) is the number of cases that are predicted as loan defaults" by the model and are indeed loan defaults In the dataset, True Negatives (TN) is the number of cases that are predicted as loan payments and are actually loan payments in the dataset, False Positives (FP) are cases that the model incorrectly predicted as loan defaults but are actually loan payments in the dataset whereas False Negatives (FN) are cases that are incorrectly predicted as loan payments, are actually loan defaults in the datasets. Therefore, the metrics can be calculated as shown in Equation 19 up to Equation 23.

$$\text{Overall Classification Accuracy}=\frac{(TP+FN)}{(TP+TN+FP+FN)}, \tag{19}$$

$$\text{Precision}=\frac{TP}{(TP+FP)}, \tag{20}$$

$$\text{Recall/sensitivity}=\frac{TP}{(TP+FN)}, \tag{21}$$

$$\text{F1-score}=\frac{2\times\text{Precision}\times\text{Recall}}{(\text{Precision}+\text{Recall})}, \tag{22}$$

$$\text{Specificity}=\frac{TN}{(TN+FP)}, \tag{23}$$

Further simplified the performance matrices as follows: when both recall and precision are high then it is a good model, when both recall and precision are low then it is a poor model, when recall is low and precision is high then the model cannot detect the classes, but it is highly trustable when it does, and when recall is high and precision is low then the model can detect the classes but includes points of other classes in it [63].

4. Data analysis and interpretation of results

4.1. Comparison of the competing classifiers across sample sizes

Figure 1 shows a comparison of classifiers based on the overall classification accuracy. The figure does not show a steady increase in the overall classification accuracy of DT, RF, K-NN and NB as sample size increases. However, SVM shows an increase in performance between sample sizes n = 15391 up until n = 26934 and at larger sample sizes as sample size increases. This imply that the results do not show any evidence that increasing the sample size can improve the classifiers’ ability to classify the cases (both loan payments and defaults) out of all the cases in the testing dataset except for SVM at some sample sizes. All the classifiers generally gave high values of overall classification accuracy of at least 74% and at most 100%. Figure 1 shows that in general, RF and DT classifiers are best performers, both achieving 100% across all sample sizes, followed by K-NN and NB while SVM achieved the lowest overall classification accuracy. The overall classification accuracy of DT and RF was stable at 100% across all sample sizes, this indicates that DT and RF correctly classified 100% of the cases (both loan defaults and loan payments) out of all the cases in the testing data.

 ISSN 2723-6471

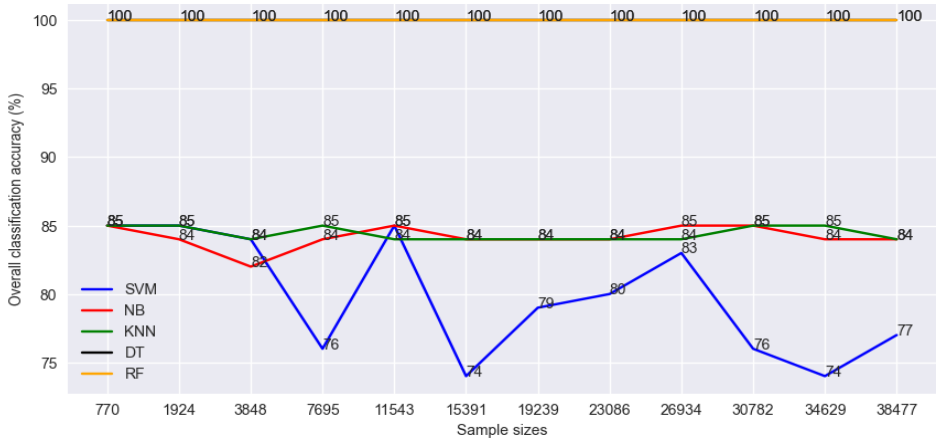

Figure 1. A Comparison of Classifiers Based on the Overall Classification Accuracy

Figure 2 shows a comparison of classifiers based on the sensitivity/recall. The figure does not show a steady increase in the sensitivity of all competing classifiers as the sample size increases. This implies that the results do not show any evidence that increasing the sample size can improve the classifiers' ability to correctly classify the positives (loan defaults) out of all the positive cases in the testing dataset. All classifiers generally gave high sensitivity of at least 84%. This implies that all the classifiers classify the loan defaults (positives) quite well, even though loan default is a minority class. Figure 2 shows that between n=11543 and n=38477 the sensitivity/recall of K-NN and SVM is equal. Figure 2 also shows that in general, the DT and RF classifier are the best performers in terms of sensitivity/recall since the two classifiers share the highest values of 100% across all sample sizes, followed by NB and K-NN while SVM achieved the lowest classification performance and was outperformed by K-NN in two sample sizes. Sensitivity/recall of DT and RF was stable at 100% which implies that DT and RF correctly classified all (100%) of the positives (loan defaults) out of all the loan defaults that are there in the testing dataset.

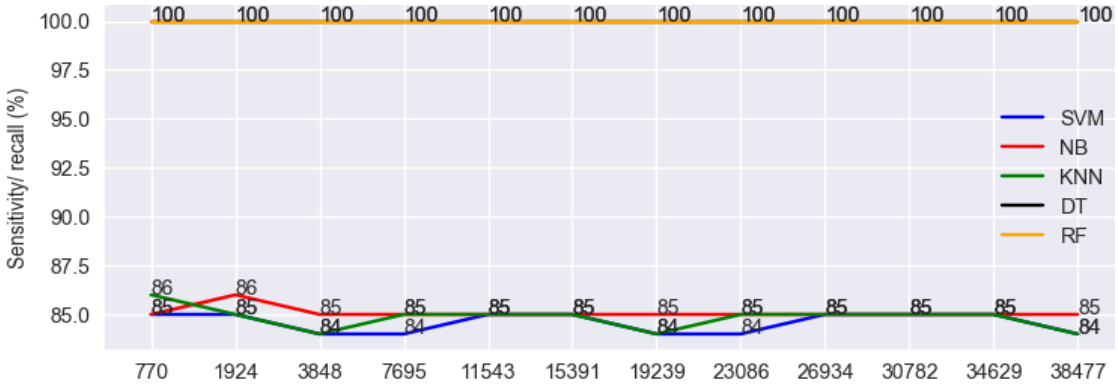

Figure 2. A Comparison of Classifiers Based on the Sensitivity/Recall

Figure 3 shows a comparison of classifiers based on the specificity. The figure shows that there is a slow increasing trend for NB and K-NN, which indicates that the specificity of NB and K-NN can be improved with an increase in sample size. This implies that the results show evidence that increasing the sample size can improve the classification ability of NB to correctly classify the negatives (payments) out of all the negative cases in the testing dataset. RF and DT are the best performing classifiers in terms of specificity due to their high values, both achieving 100% specificity across all sample sizes which implies that both RF and DT correctly classified all (100%) of the individuals who made payments on their loans which was the majority class. Therefore, DT and RF were the best performing classifiers followed by NB and K-NN while SVM achieved the lowest classification performance across all sample sizes.

 ISSN 2723-6471

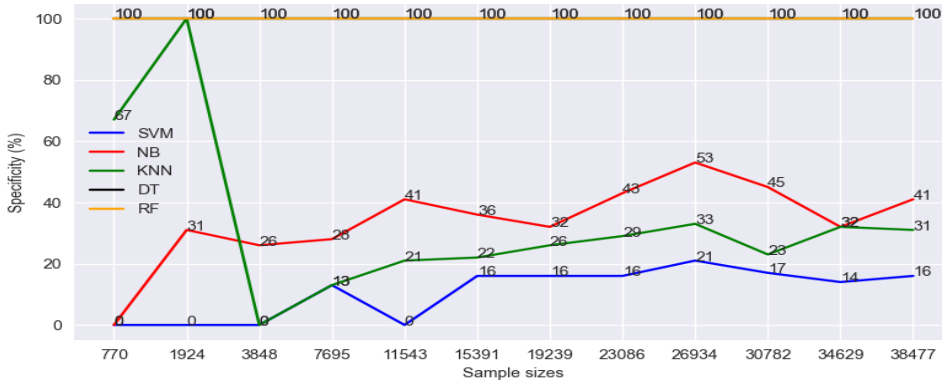

Figure 3. A Comparison of Classifiers Based on Specificity

Figure 4 shows a comparison of classifiers based on the F1-score. The figure does not show a steady increase in the F1-score for DT, RF, K-NN and NB as the sample size increases. However, SVM's F1-score showed an increase with an increase in sample sizes between n=15391 up to n=26934. This implies that the results do not show any evidence that increasing the sample size can improve the classifiers' ability to correctly classify the positives (loan defaults) out of all the positive cases in the testing dataset except for SVM. All classifiers generally gave high values of F1-score of at least 85%. This implies that all the classifiers classify the loan defaults (positives) quite well, even though loan default is a minority class. Figure 4 shows that in general, the DT and RF classifiers are the best performers in terms of F1-score since the two classifiers share the highest value of 100% F1-score, followed by K-NN and NB while SVM has the lowest values of F1-score on most of the sample sizes. RF and DT are the most stable classifiers in terms of F1-score since they have a value of 100% across all the sample sizes.

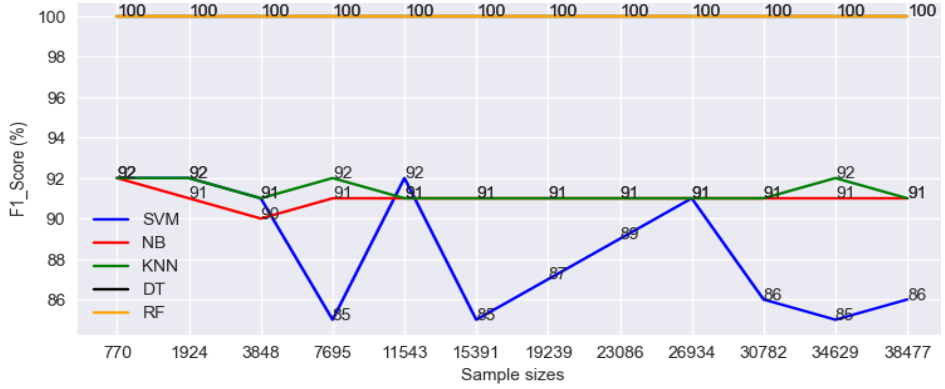

Figure 4. A Comparison of Classifiers Based on the F1-Score

Figure 5 shows a comparison of classifiers based on precision. The figure does not show a steady decrease or increase in the precision of DT, RF, K-NN and NB as the sample size increases. However, SVM's precision showed an increase with an increase in sample sizes between n=15391 up to n=26934. This implies that the results do not show any evidence that increasing the sample size can improve the classifiers' ability to correctly classify the positives (loan defaults) out of all the positive cases in the testing dataset except for SVM. All classifiers generally gave high values of precision of at least 85%. This implies that all the classifiers classify the loan defaults (positives) quite well, even though loan default is a minority class. Figure 5 shows that in general, RF and DT are the best performers in terms of precision since both models yielded 100% precision across all sample sizes, followed by K-NN and NB while SVM yielded the lowest precision. RF and DT are the most stable classifiers in terms of precision since its classification ability was stable across all sample sizes while NB was constant from n=23086 up until the largest sample size.

 ISSN 2723-6471

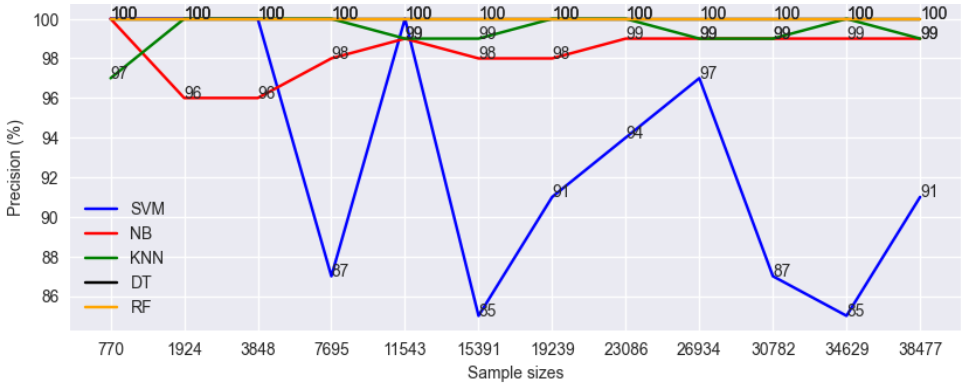

Figure 5. A Comparison of Classifiers Based on Precision

Figure 6 shows a comparison of classifiers based on the average classification ability. It is noticeable from figure 6 that SVM has the lowest average classification ability for all samples, followed by K-NN which has the second lowest average value of classification metrices from the third smallest sample size (n = 3848) onwards based on the averages of classification accuracy, precision, specificity, sensitivity/recall and F1-Score. RF and DT have the highest average classification ability across all sample sizes, so the two models are the best classifiers on average, followed by NB which has the highest average for ten out of twelve samples. Therefore, the classifiers can be listed in ascending order of average classification performance as follows: RF and DT, NB, K-NN, SVM. It is also worth noting that a slow upward trend in the average performance of NB, K-NN and SVM is evident as the sample size increases, implying that increasing the sample size slightly improved the average classification performance of NB, K-NN and SVM.

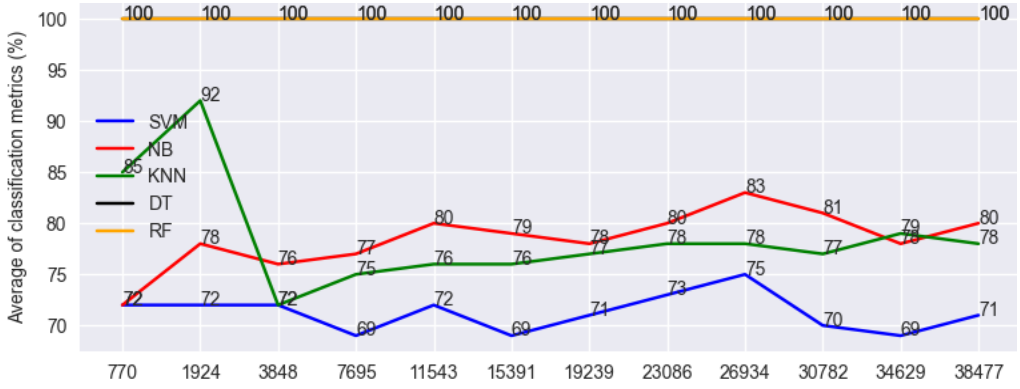

Figure 6. Comparison of the Classifiers Based on the Average Classification Ability

4.2. Classification evaluation of the classifiers trained with SMOTE-Tomek balanced dataset

Table 2 shows that the classification performance of the SVM when data was balanced using SMOTE-Tomek decreased in terms of overall classification accuracy, F1-score and precision by the negative values across all sample sizes. However, it is evident from the output that specificity, although low, showed an improvement between 1 and 20% at some sample sizes when data was balanced using SMOTE-Tomek. Also, in terms of sensitivity/recall, the classification performance of SVM improved with 1 to 4%, leading to sensitivity values of 82 to 89%. This indicates that the application of SMOTE-Tomek improved the classification performance of SVM with regards to sensitivity/recall and specificity, but it decreased overall classification accuracy, specificity, F1-score and precision.

Table 2. Comparison of classification metrics for SVM and SVM with SMOTE-Tomek, [balanced – imbalanced]

Sample size	Overall classification accuracy (%)	Sensitivity or recall (%)	specificity (%)	F1 Score (%)	Precision (%)
806	59 [-26]	89 [4]	20 [20]	71 [-21]	59 [-41]
1974	54 [-31]	85 [0]	15 [15]	68 [-24]	59 [-44]
3984	46 [-38]	82 [-2]	14 [14]	59 [-32]	46 [-54]

 ISSN 2723-6471

8108	49 [-27]	83 [-1]	14 [1]	62 [-23]	50 [-37]
12262	63 [-22]	86 [1]	17 [17]	63 [-29]	50 [-50]
16310	51 [-23]	85 [0]	15 [-1]	64 [-21]	51 [-34]
20284	52 [-27]	86 [2]	17 [1]	65 [-22]	52 [-39]
24740	51 [-29]	85 [1]	16 [0]	64 [-25]	51 [-43]
28516	51 [-32]	85 [0]	16 [-5]	64 [-27]	51 [-46]
32724	52 [-24]	85 [0]	15 [-2]	65 [-21]	52 [-35]
36854	49 [-25]	84 [-1]	15 [1]	62 [-23]	49 [-36]
41242	51 [-26]	85 [1]	16 [0]	45 [-41]	52 [-39]

Table 3 compares the classification metrics for NB and NB trained with data that has been balanced with SMOTE-Tomek. The table shows that like SVM trained with that data that has been balanced using SMOTE-Tomek, the classification performance of the NB after balancing the data with SMOTE-Tomek decreased in terms of overall classification accuracy, F1-score and precision across all sample sizes. However, in terms of specificity, at smaller sample sizes the classification performance improved (n =806 and n=3984) while the rest of the sample sizes showed a decrease in classification performance of at most 33%. There was an improvement in sensitivity/recall of 7 to 11% across all sample sizes which imply that SMOTE-Tomek improved the performance of NB with regards to sensitivity/recall.

Table 3. Comparison of classification metrics for NB and NB with SMOTE-Tomek, [balanced – imbalanced]

Sample size	Overall classification accuracy (%)	Sensitivity or recall (%)	specificity (%)	F1 Score (%)	Precision (%)
806	49 [-36]	92 [7]	20 [20]	59 [-33]	43 [-57]
1974	51 [-33]	94 [8]	21 [-10]	61 [-30]	45 [-51]
3984	67 [-15]	96 [11]	30 [4]	77 [-13]	64 [-32]
8108	58 [-26]	95 [10]	24 [-4]	68 [-23]	53 [-45]
12262	47 [-38]	93 [8]	20 [-21]	57 [-34]	41 [-58]
16310	45 [-39]	95 [10]	20 [-16]	53 [-38]	37 [-61]
20284	66 [-18]	95 [10]	29 [-3]	76 [-15]	63 [-35]
24740	53 [-31]	95 [10]	23 [-20]	63 [-28]	47 [-52]
28516	44 [-41]	94 [9]	20 [-33]	53 [-38]	37 [-66]
32724	54 [-31]	95 [10]	22 [-23]	64 [-27]	48 [-51]
36854	47 [-37]	94 [9]	20 [-12]	56 [-35]	40 [-59]
41242	50 [-34]	95 [10]	22 [-19]	59 [-32]	43 [-56]

Table 4 compares the classification metrics for K-NN and K-NN with SMOTE-Tomek (K-NN on a balanced dataset). The table shows that the K-NN trained with data that has been balanced with SMOTE-Tomek decreased in terms of overall classification accuracy, F1-score and precision across all sample sizes. At n=3984 and n=8108, the specificity improved by 19% and 2% however experienced a decrease for the rest of the sample sizes. There was an improvement in sensitivity/recall of at most 2%. Most of the sample sizes showed an improvement in classification performance with regards to sensitivity/recall while some sample sizes experienced no difference, which is an indication that SMOTE-Tomek generally only improved the sensitivity/ recall and specificity in relatively few samples, and decreased the other classification metrics, so it did not assist to mitigate the impact of data imbalances on the K-NN.

Table 4. Comparison of classification metrics for K-NN and K-NN with SMOTE-Tomek

Sample size	Overall classification accuracy (%)	Sensitivity or recall (%)	specificity (%)	F1 Score (%)	Precision (%)
806	61 [-24]	86 [0]	17 [-50]	73 [-19]	64 [-35]

ISSN 2723-6471

1974	62 [-23]	86 [1]	16 [-84]	75 [-17]	67 [-33]
3984	63 [-21]	86 [2]	19 [19]	75 [-16]	66 [-34]
8108	64 [-21]	85 [0]	15 [2]	77 [-15]	70 [-30]
12262	66 [-18]	85 [0]	17 [-4]	79 [-12]	73 [-26]
16310	66 [-18]	86 [1]	17 [-5]	78 [-13]	72 [-27]
20284	66 [-18]	85 [1]	17 [-9]	78 [-13]	72 [-28]
24740	67 [-17]	86 [1]	19 [-10]	79 [-12]	73 [-27]
28516	67 [-17]	85 [0]	18 [-15]	79 [-12]	73 [-26]
32724	67 [-18]	86 [1]	17 [-6]	79 [-12]	73 [-26]
36854	66 [-19]	86 [1]	18 [-14]	78 [-14]	72 [-28]
41242	66 [-18]	85 [1]	18 [-13]	78 [-13]	72 [-27]

5. Discussion of Result

The aim of this study was to determine the impact of sample size variations on the robustness of ML classification algorithms (SVM, RF, NB, DT and K-NN) to data imbalance, as well as to identify the ultimate ML classifier for loan default prediction from the five. The classification ability of the five ML algorithms was compared in terms of overall classification accuracy, sensitivity/recall, specificity, F1-score and precision. [15] found that effects of sample sizes differ with the different classifiers which was evident in this study as the classification performance of the classifiers was impacted differently by sample size variations. This study found that as sample size increased, for both specificity and on average classification of NB, SVM (at larger sample sizes) and K-NN showed a slow increasing trend in classification performance however showed no trend that increasing the sample size improved the sensitivity/recall of the classifiers. When SMOTE-Tomek was applied, both K-NN and NB exhibited an upward trend in terms of overall classification accuracy and the F1-score of SVM also increased with an increase in sample size. These findings by the study are similar to those made by [19] where the authors found that NB was among the classifiers whose classification criteria improved as sample size increased while [18] found that SVM and K-NN had accuracy advantages for larger references datasets, while [15] found that the classification performance of the classifiers improved with an increase in sample size.

This study found that on average, sample size increments did not improve classification performance of most of the classifiers, in most of the sample sizes. Based on these findings, it can be noted that the results did not show that increasing the sample size while the imbalance ratio is constant can improve the performance of the models. So, the study recommends that increasing the sample size does not improve the ML classifiers' robustness to data imbalances. However, it is worth noting that this recommendation is based on the results of the empirical analysis conducted in this study, which also has some limitations. For instance, only stratified random sampling was implemented whereas other sampling methods such as systematic random sampling or cluster sampling could have been used. This is because a comparison of sampling methods was not an objective of the current study, and the use of stratified sampling for ML is justified. For example, [65] explain that finding representative samples is made easy by stratification which ensures that the number of samples for each class is balanced and that the variation of the data within each class is taken into account to maintain a balance in the number of samples for the majority and minority classes which aids in maintaining the original data structure feature information. The authors add that stratified sampling works best when data is distributed unevenly, which was the case in the current study. Another scope limitation is that one dataset was used, so future studies on the impact of sample sizes on the robustness of ML classifiers to data imbalances should consider using various datasets simultaneously to compare the performance of the classifiers across different datasets. The current study also was limited to 12 sample sizes, with a 10% difference so a future study with a lower percentage difference between the samples (meaning more samples than the 12) may enable the authors to better observe the trend in the performance of the ML classifiers as the sample size increases.

 ISSN 2723-6471

DT and RF achieved 100% classification performance across all sample sizes when the minority class was loan_default = 1 on all classification metrics similar to the results obtained by [13] where RF was among the classifiers trained on an imbalanced loan default dataset and the author found that RF was the best performer across all metrics, contrary to the findings by [10] where the authors found that RF outperformed DT in terms of classification ability. Although studies by [21] and [66] employed different datasets to the one used in this study which is loan default dataset, the results obtained in this study in terms of the best performing models are supported by studies by [21] and [66] where both studies also found that RF consistently achieved the highest performances and provided the best classification ability, while [15] found RF to be the best all-rounder classifier outperforming competing classifiers in their study similar to the results obtained in this study. Therefore, this is an indication that RF has superior classification ability even when trained on different types of datasets both balanced and imbalanced datasets.

[23] and [15] where both studies found RF had the highest overall classification accuracy, RF achieved the highest overall classification accuracy while [11] found that when trained on an imbalanced loan default dataset, RF achieved the highest precision like the results obtained by this study. [4] found that when trained on a credit default dataset, RF and DT gave good performances emphasizing their robustness to class imbalance when credit default prediction was the objective which is what the results of this study has proven. [10] also found that RF was a better option for loan default prediction. The best performance of RF in the current study could be because it is known to have a good classification performance for imbalanced data like in previous studies such as the ones conducted by [8], [53] and [54]. Therefore, based on these results, this study recommends the use of RF and DT classifiers for loan default detection, mainly due to the classifiers' superior classification performance regardless of sample sizes when the data was imbalanced across all sample sizes. Future studies should further explore their performance by conducting a comparison study to determine the best performer between the two classifiers when data is imbalanced.

The classifiers can be listed based on average classification performance from best performing to the worst performer as follows; RF and DT, K-NN, NB and lastly SVM. Contrary to the results obtained by [64] which showed that SVM had the highest classification efficiency and was the best classifier for highly imbalanced data as well as a study by [19] where the authors found that SVM outperformed across all sample sizes, this study found that on average SVM was the worst performing classifier. These results obtained by this study are supported by those obtained by [21] where the authors found that SVM significantly achieved lower performances. SVM is also known to be sensitive to imbalanced dataset and decision boundary have bias towards minority class [37] and this could be the reason for its poor performance in the current study. This study also found that K-NN was the second worst performing classifier on average classification.

These findings by the study are similar to those made by [64] and [19] where although trained on a different dataset to the one used in the current study, the studies found that K-NN achieved the least classification performances and these findings where further supported by [13] where the author found that when trained on a credit default dataset K-NN was the worst performing classifier, contradictory to the results by [6] where K-NN was found to be the best model for credit prediction due to its high classification performance. Therefore, this study recommends further research on the impact of sample sizes on the classification performance of ML classifiers to imbalanced data, future studies can extend the scope of the current study by performing a comprehensive hyperparameter tuning for their classifiers in an attempt to maximize the performance of each classifier instead of being limited to default classifiers as well as considering decreasing the imbalance ratio of the data that would be employed for their studies, and testing the classifiers' performance asymptotically in terms of sample size.

The results of this study showed that SMOTE-Tomek negatively affected the average performance of the models compared to when the data was imbalanced. This study found that SMOTE-Tomek only improved the sensitivity/recall of SVM by 1 to 4%, and the specificity of this classifier improved by 1 to 20%, but noticeably decreased overall classification accuracy, F1-score and Precision for this classifier. Sensitivity for NB only increased by 7 to 10% across all sample sizes, and the specificity by 4 and 20% for only two sample sizes after SMOTE-Tomek, but all other metrics decreased noticeably. Sensitivity for K-NN remained unchanged in some samples, increased by 1% in most samples and the specificity by 2 and 19% for only two sample sizes after balancing the data with SMOTE-Tomek whereas other classification metrics decreased noticeably similar to the finding by [59] where the authors noticed that the ML models

used in their study experienced a decrease in performance when implementing the class imbalance correction techniques.

[1] and [8] indicated that the use of sampling methods may improve the performance of the classifiers, i.e. oversampling followed by under sampling methods while [55] and [67] found that hybrid sampling methods (SMOTE-Tomek, SMOTE-RUS, SMOTE-ROS and SMOTE-ENN) showed the best classification performance among ML techniques and both studies found that when combined with SMOTE-Tomek the classification performance of the classifiers improved and [59] found hybrid techniques to give better competitive results, while [68] found that SMOTE with a combination of ENN technique was the best algorithm. Therefore, future studies should extent the scope of this study by employing hybrid sampling methods such as SMOTE+ENN with the aim of improving the classification performances of the ML algorithms and comparing the findings to SMOTE-Tomek results of the current study on a loan default dataset.

As sample size increased, SVM experienced the most fluctuations in classification performance in most metrics than the other four competing classifiers, therefore although it wasn't the best performing classifier, it was more impacted by sample sizes than NB, DT, RF and K-NN. SVM, NB and K-NN performed well in terms of Sensitivity/recall of at least 84% across all the samples but they performed poorly with regards to specificity, which was 0% across all the samples for SVM, but ranged from 0 to 53% for NB whereas in ranged between 0 to 100% for K-NN. This was an indication that when trained using imbalanced data, SVM, NB and K-NN correctly classified positives (those who indeed defaulted on their loans as loan defaults which was the minority group) which was the, minority class but poorly classified negatives (those who indeed made payments on their loans as having made payments on their loans which was the majority group) which was the minority class (sensitivity/recall was higher than specificity).

6. Recommendations

The results of this study RF and DT showed superior performance regardless of the sample size so we recommend that for loan default detection, these two ML classifiers are the most robust to the data imbalance in this dataset and should be used instead of SVM, K-NN and NB. This study was limited to exploring the classification ability of only five ML classifiers (RF, SVM, NB, DT and K-NN) on loan default dataset, therefore future researchers should use a different set of ML classifiers particularly ones which outperformed RF and DT in previous studies when predicting loan defaults such as Extreme trees classifier which outperformed all the classifiers used in the study when trained with loan default data [11] and XGBoost which outperformed RF, LR, K-NN, DT when trained on a loan default dataset [12] so as to determine the classifier with the most predictive powers or add to the ones used in the current study. As [10] found that combining ensemble methods with hybrid sampling techniques produces optimal classification results. Also, the scope of the current study was limited to traditional ML classifies and eminent studies including deep learning techniques such as Recurrent Neural Networks (RNNs) in their empirical analysis of the impact of sample size on the performance of these algorithms when trained with imbalanced data may be conducted to extend the scope of the current study. Since SMOTE-Tomek decreased the performance of the ML classifiers under study, it is recommended that feature studies should consider other data imbalance handling methods such as the Balanced Bagging Classifier, Adaptive Synthetic Sampling (ADASYN), combining SMOTE-N with Edited Nearest Neighbours (SMOTEENN), Bi-phasic SMOTE (BP-SMOTE) and compare it with SMOTE-Tomek for datasets that are similar to the ones used in the current study. The current study did not use advanced feature selection methods, and the base parameters were used to train the ML classifiers. Therefore, feature studies may explore various feature selection and hyperparameter tuning approaches when evaluating the performance of the ML classifiers when trained with imbalanced data. A combination of samples size, data balancing methods and other ML and deep learning methods recommended herein can also be explored in a future study.

7. Declarations

7.1. Author Contributions

Conceptualization: B.T.K., T.V.M.; Methodology: B.T.K., T.V.M.; Software: B.T.K.; Validation: T.V.M.; Formal Analysis: B.T.K.; Investigation: B.T.K.; Resources: T.V.M.; Data Curation: B.T.K.; Writing – Original Draft

Preparation: B.T.K.; Writing – Review and Editing: T.V.M.; Visualization: B.T.K.; All authors have read and agreed to the published version of the manuscript.

7.2. Data Availability Statement

The secondary data presented in this study are available through the link provided under the Data section of the paper.

7.3. Funding

The article processing charge (APC) was paid for by the North West University (NWU).

7.4. Institutional Review Board Statement

Not applicable.

7.5. Informed Consent Statement

Not applicable.

7.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] S. Ashraf and T. Ahmed, "Machine Learning shrewd approach for an imbalanced dataset conversion samples," *Journal of Engineering and Technology*, vol. 11, no. 1, pp. 1–22, Jun. 2020.

[2] V. López, A. Fernández, S. García, V. Palade, and F. Herrera, "An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics," *Information Sciences*, vol. 250, no. 7, pp. 113–141, Jul. 2013.

[3] W. Zheng and M. Jin, "The effects of class imbalance and training data size on classifier learning: an Empirical study," *SN Computer Science*, vol. 1, no. 2, pp. 17-29, Feb. 2020.

[4] L. Dube and T. Verster, "Enhancing classification performance in imbalanced datasets: A comparative analysis of machine learning models," *Data Science in Finance and Economics*, vol. 3, no. 4, pp. 354–379, Jan. 2023.

[5] P. Vuttipittayamongkol, E. Elyan, and A. Petrovski, "On the class overlap problem in imbalanced data classification," *Knowledge-Based Systems*, vol. 212, no. 106631, pp. 17-29, Nov. 2020.

[6] W. T. Loo, K. W. Khaw, X. Chew, A. Alnoor, and S. T. Lim, "Predicting the loan default using machine learning algorithms: a case study in India," *Journal of Engineering and Technology (JET)*, vol. 14, no. 2, pp. 17-27, Dec. 2023.

[7] S. M. A. Elrahman and A. Abraham, "A Review of Class Imbalance Problem," *Journal of Network and Innovative Computing*, vol. 1, no. 10, pp. 332–340, Oct. 2013.

[8] A. Singh, R. K. Ranjan, and A. Tiwari, "Credit Card Fraud Detection under Extreme Imbalanced Data: A Comparative Study of Data-level Algorithms," *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 34, no. 4, pp. 571–598, Apr. 2021.

[9] M. Madaan, A. Kumar, C. Keshri, R. Jain, and P. Nagrath, "Loan default prediction using decision trees and random forest: A comparative study," *IOP Conference Series Materials Science and Engineering*, vol. 1022, no. 1, pp. 12-24, Jan. 2021.

[10] M. Pamuk, R. O. Grendel, and M. Schumann, "Towards ML-based Platforms in Finance Industry – An ML Approach to Generate Corporate Bankruptcy Probabilities based on Annual Financial Statements," *ACIS 2021 Proceedings. 8*, vol. 2021, no. 12, pp. 11-22, Dec. 2021.

[11] M. Anand, A. Velu, and P. Whig, "Prediction of Loan Behaviour with Machine Learning Models for Secure Banking," *Journal of Computer Science and Engineering (JCSE)*, vol. 3, no. 1, pp. 1–13, Feb. 2022.

[12] D. Shokeen, V. Grover, and V. Verma, "Mitigating Loan Default Risk in the Banking Sector: Machine Learning Solutions and Comparative Performance Analysis," vol. 2023, no. 6, pp. 10-20, Jun. 26, 2023.

[13] S. Fati, "A loan default prediction model using machine learning and feature engineering," *ICIC Express Letters*, vol. 18, no. 1, pp. 27–37, 2024.

[14] K. Andric, D. Kalpic, and Z. Bohacek, "An insight into the effects of class imbalance and sampling on classification accuracy in credit risk assessment," *Computer Science and Information Systems*, vol. 16, no. 1, pp. 155–178, Nov. 2018.

[15] C. A. Ramezan, T. A. Warner, A. E. Maxwell, and B. S. Price, "Effects of training set size on supervised Machine-Learning Land-Cover classification of Large-Area High-Resolution remotely sensed data," *Remote Sensing*, vol. 13, no. 3, pp. 27 Jan. 2021.

[16] I. B. A. Peling, I. N. Arnawan, I. P. A. Arthawan, and I. G. N. Janardana, "Implementation of data mining to predict period of students study using Naive Bayes algorithm," *International Journal of Engineering and Emerging Technology*, vol. 2, no. 1, pp. 53-57, Sep. 2017.

[17] B. Mahesh, "Machine learning algorithms - a review," *International Journal of Science and Research (IJSR) ResearchGate Impact Factor*, vol. 9, no. 1, pp. 381-386, Jan 2018.

[18] S. S. Heydari and G. Mountrakis, "Effect of classifier selection, reference sample size, reference class distribution and scene heterogeneity in per-pixel classification accuracy using 26 Landsat sites," *Remote Sensing of Environment*, vol. 204, no. 10, pp. 648–658, Oct. 2017.

[19] S. K. Punia, M. Kumar, T. Stephan, G. G. Deverajan, and R. Patan, "Performance analysis of machine learning algorithms for big data classification," *International Journal of E-Health and Medical Communications*, vol. 12, no. 4, pp. 60–75, Apr. 2021.

[20] Z. Chen, C. Li, and W. Sun, "Bitcoin price prediction using machine learning: An approach to sample dimension engineering," *Journal of Computational and Applied Mathematics*, vol. 365, no. 112395, pp. 13-26, Aug. 2019.

[21] J. Hartmann, J. Huppertz, C. Schamp, and M. Heitmann, "Comparing automated text classification methods," *International Journal of Research in Marketing*, vol. 36, no. 1, pp. 20–38, Oct. 2018.

[22] G. Karatas, O. Demir, and O. K. Sahingoz, "Increasing the performance of Machine Learning-Based IDSs on an imbalanced and Up-to-Date dataset," *IEEE Access*, vol. 8, no. 2, pp. 32150–32162, Feb. 2020.

[23] Q. Li, C. Zhao, X. He, K. Chen, and R. Wang, "The impact of partial balance of imbalanced dataset on classification performance," *Electronics*, vol. 11, no. 9, pp. 17-34, Apr. 2022.

[24] E. F. Swana, W. Doorsamy, and P. Bokoro, "Tomek Link and SMOTE Approaches for Machine Fault Classification with an Imbalanced Dataset," *Sensors*, vol. 22, no. 9, pp. 21-34, Apr. 2022.

[25] H. Hairani, A. Anggrawan, and D. Priyanto, "Improvement Performance of the Random Forest Method on Unbalanced Diabetes Data Classification Using Smote-Tomek Link," *JOIV : International Journal on Informatics Visualization*, vol. 7, no. 1, pp. 258–264, Feb. 2023.

[26] Q. Guo, "Urban tree classification based on Object-Oriented Approach and Random Forest Algorithm using Unmanned Aerial Vehicle (UAV) multispectral imagery," *Remote Sensing*, vol. 14, no. 16, pp. 17-34, Aug. 2022.

[27] A. B. Parsa, H. Taghipour, S. Derrible, and A. Mohammadian, "Real-time accident detection: Coping with imbalanced data," *Accident Analysis and Prevention*, vol. 129, no. 8, pp. 202–210, Aug. 2019.

[28] M. Sheykhmousa, M. Mahdianpari, H. Ghanbari, F. Mohammadimanesh, P. Ghamisi, and S. Homayouni, "Support Vector Machine Versus Random Forest for Remote Sensing Image Classification: A Meta-Analysis and Systematic Review," *IEEE Journal Of Selected Topics In Applied Earth Observations And Remote Sensing*, vol. 13, no. 9, pp. 6308-6325, Sep. 2020.

[29] A. A. Pinheiro, I. M. Brandao, and C. Da Costa, "Vibration analysis in turbomachines using machine learning techniques," *European Journal of Engineering and Technology Research*, vol. 4, no. 2, pp. 12–16, Feb. 2019.

[30] Y. Fuqing, U. Kumar, and D. Galar, "A comparative study of artificial neural networks and support vector machine for fault diagnosis," *International Journal of Performability Engineering*, vol. 9, no. 1, pp. 49–60, Jan. 2013.

[31] V. Flores and C. Leiva, "A comparative study on supervised machine learning algorithms for copper recovery quality prediction in a leaching process," *Sensors*, vol. 21, no. 6, pp. 20-32, Mar. 2021.

[32] B. S. Saljoughi and A. Hezarkhani, "A comparative analysis of artificial neural network (ANN), wavelet neural network (WNN), and support vector machine (SVM) data-driven models to mineral potential mapping for copper mineralizations in the Shahr-e-Babak region, Kerman, Iran," *Applied Geomatics*, vol. 10, no. 3, pp. 229–256, Jun. 2018.

[33] F. Zhang, M. Petersen, L. Johnson, J. Hall, and S. E. O'Bryant, "Hyperparameter Tuning with High Performance Computing Machine Learning for Imbalanced Alzheimer's Disease Data," *Applied Sciences*, vol. 12, no. 13, pp. 11-22, Jul. 2022.

[34] J. Wainer and P. Fonseca, "How to tune the RBF SVM hyperparameters? An empirical evaluation of 18 search algorithms," *Artificial Intelligence Review*, vol. 54, no. 6, pp. 4771–4797, May 2021.

[35] S. S. Heydari and G. Mountrakis, "Effect of classifier selection, reference sample size, reference class distribution and scene heterogeneity in per-pixel classification accuracy using 26 Landsat sites," *Remote Sensing of Environment*, vol. 204, no. 1, pp. 648–658, Oct. 2017.

[36] S. K. Punia, M. Kumar, T. Stephan, G. G. Deverajan, and R. Patan, "Performance analysis of machine learning algorithms for big data classification," *International Journal of E-Health and Medical Communications*, vol. 12, no. 4, pp. 60–75, Apr. 2021.

[37] P. Kumar, R. Bhatnagar, K. Gaur, and A. Bhatnagar, "Classification of Imbalanced Data:Review of Methods and Applications," *IOP Conference Series Materials Science and Engineering*, vol. 1099, no. 1, pp. 8-16, Mar. 2021.

[38] A. S. Eesa, Z. Orman, and A. M. A. Brifcani, "A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems," *Expert Systems With Applications*, vol. 42, no. 5, pp. 2670–2679, Nov. 2014.

[39] J. Liang, Z. Qin, S. Xiao, L. Ou, and X. Lin, "Efficient and secure decision tree classification for Cloud-Assisted online diagnosis services," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 4, pp. 1632–1644, Jun. 2019.

[40] F.-J. Yang, "An Extended Idea about Decision Trees," *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, vol. 2019, no. 12, pp. 349-354 Dec. 2019.

[41] N. Nai-Arun and R. Moungmai, "Comparison of classifiers for the risk of diabetes prediction," *Procedia Computer Science*, vol. 69, no. 1, pp. 132–142, Jan. 2015.

[42] A. Dey, "Machine Learning Algorithms: A review," *International Journal of Computer Science and Information Technologies*, vol. 7, no. 3, pp.1174-1179, 2016.

[43] B. Charbuty and A. Abdulazeez, "Classification based on decision tree algorithm for machine learning," *Journal of Applied Science and Technology Trends*, vol. 2, no. 01, pp. 20–28, Mar. 2021.

[44] L. Dube and T. Verster, "Enhancing classification performance in imbalanced datasets: A comparative analysis of machine learning models," *Data Science in Finance and Economics*, vol. 3, no. 4, pp. 354–379, Jan. 2023.

[45] J. O. Awoyemi, A. O. Adetunmbi and S. A. Oluwadare, "Credit card fraud detection using machine learning techniques: A comparative analysis," *2017 International Conference on Computing Networking and Informatics (ICCNI)*, Lagos, Nigeria, vol. 2017, no. 10, pp. 1-9, Oct. 2007.

[46] A. R. Safitri and M. A. Muslim, "Improved accuracy of naive Bayes classifier for determination of customer churn uses SMOTE and genetic algorithms," *Journal of Soft Computing Exploration*, vol. 1, no. 1, pp. 70-75, Sep. 2020.

[47] I. B. A. Peling, I. N. Arnawan, I. P. A. Arthawan, and I. G. N. Janardana, "Implementation of data mining to predict period of students study using Naive Bayes algorithm," *International Journal of Engineering and Emerging Technology*, vol. 2, no. 1, pp. 53-57, Sep. 2017.

[48] N. Sateesh, E, S. Bhanusri, k, K. M. Pasha, S. K. Sameer, P. G. krishna, and Sunitha.A, "Crop Recommendation system using machine learning algorithm", *UGC Care Group I*, vol. 13, no. 20, pp. 184-188, 2023.

[49] Rahman, A.K.M., F.M. Javed Mehedi Shamrat, Z. Tasnim, J. Roy, and S.A. Hossain, "A comparative study on liver disease prediction using supervised machine learning algorithms," *International Journal Of Scientific and Technology Research*, vol. 8, no. 11, pp. 419–420, Nov. 2019.

[50] Q. Wang, S. Wang, B. Wei, W. Chen, and Y. Zhang, "Weighted K-NN Classification Method of Bearings Fault Diagnosis with Multi-Dimensional Sensitive Features," *IEEE Access*, vol. 9, no. 1, pp. 45428–45440, Jan. 2021.

[51] K. H. Foysal, H. J. Chang, F. Bruess, and J. W. Chong, "SmartFit: smartphone application for garment fit detection," *Electronics*, vol. 10, no. 1, pp. 15-27, Jan. 2021.

[52] Z. Qu, H. Li, Y. Wang, J. Zhang, A. Abu-Siada, and Y. Yao, "Detection of electricity theft behavior based on improved synthetic minority oversampling technique and random forest classifier," *Energies*, vol. 13, no. 8, pp. 20-32, Apr. 2020.

[53] J. Hartmann, J. Huppertz, C. Schamp, and M. Heitmann, "Comparing automated text classification methods," *International Journal of Research in Marketing*, vol. 36, no. 1, pp. 20–38, Oct. 2018.

[54] C. A. Ramezan, T. A. Warner, A. E. Maxwell, and B. S. Price, "Effects of training set size on supervised Machine-Learning Land-Cover classification of Large-Area High-Resolution remotely sensed data," *Remote Sensing*, vol. 13, no. 3, pp. 27-41, Jan. 2021.

[55] T. Kimura and Hosei University, "Customer churn prediction with hybrid resampling and ensemble learning," *Journal of Management Information and Decision Sciences*, vol. 25, no. 1, pp. 1-23, Feb. 2022.

[56] G. Otoo, "Analysis of credit card fraud detection methods," B.Sc. Computer Science, Ashesi University College, vol 2021, no. 5, pp. 48-56, May 2021.

[57] 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," *SIGKDD Explorations*, vol. 6, no. 1, pp. 20-29, Jun. 2003.

[58] N. A. A. Khleel and K. Nehéz, "A novel approach for software defect prediction using CNN and GRU based on SMOTE Tomek method," *Journal of Intelligent Information Systems*, vol. 60, no. 3, pp. 673–707, May. 2023.

[59] G. Nath, R. Luthra and R. Chellani, "A Review On Class Imbalanced Correction Techniques: A Case Of Credit Card Default Prediction On A Highly Imbalanced Dataset," *Proceedings of 7th International Conference of Business Analytics and Intelligence*, Indian Institute of Management, Bangalore, India, vol. 2019, no. 12, pp. 6-18, 2019.

[60] Z. Zhao, T. Cui, S. Ding, J. Li, and A. G. Bellotti, "Resampling Techniques Study on class imbalance problem in credit risk prediction," *Mathematics*, vol. 12, no. 5, pp. 27-39, Feb. 2024.

[61] A. Namvar, M. Siami, F. Rabhi, and M. Naderpour, "Credit risk prediction in an imbalanced social lending environment," *International Journal of Computational Intelligence Systems*, vol. 11, no. 1, pp. 925-935, Jan. 2018.

[62] S. Chowdhury and M. P. Schoen, "Research Paper Classification using Supervised Machine Learning Techniques," *In 2020 Intermountain Engineering, Technology and Computing (IETC)*, vol. 2020, no. 10, pp. 1–6, Oct. 2020.

[63] V. Kumar, "Addressing Binary Classification over Class Imbalanced Clinical Datasets Using Computationally Intelligent Techniques," *Healthcare*, vol. 10, no. 7, pp. 28-42, Jul. 2022.

[64] S. W. Juma, "Robust statistical learning for optimal classification of imbalanced data," MSc Thesis, Strathmore University, 2021.

[65] J. Sadaiyandi, P. Arumugam, A. K. Sangaiah, and C. Zhang, "Stratified Sampling-Based Deep Learning Approach to increase prediction accuracy of unbalanced dataset," *Electronics*, vol. 12, no. 21, pp. 16-28, Oct. 2023.

[66] Z. S. Rubaidi, B. B. Ammar, and M. B. Aouicha, "Fraud detection using large-scale imbalance dataset," *International Journal of Artificial Intelligence Tools*, vol. 31, no. 08, pp. 23-39, Sep. 2022.

[67] M. E. F. Milli, S. Aras, and İ. D. Kocakoç, "Investigating the effect of class balancing methods on the performance of machine learning techniques: Credit Risk application," *İzmir Yönetim Dergisi*, vol. 5, no. 1, pp. 55–70, Jun. 2024.

[68] M. R. Ali, "Prediction Accuracy of Financial Data-Applying Several Resampling Techniques," MSc dissertation, Computer Science North Dakota State University of Agriculture and Applied Science Fargo, North Dakota, vol. 2020, no. 10, pp. 36-42, Oct. 2020.