IIUC Journal of Science and Engineering
Vol.-1, Issue-1, December 2023, pp. 61-82
ISSN: 3005-5873

Unleashing class imbalance problem in loan dataset through a novel oversampling approach based on FCM

Subrina Akter
Department of Computer Science and Engineering
International Islamic University Chittagong (IIUC), Bangladesh

Article info

Keywords
Decision tree
F-measure
G-mean
Imbalanced dataset
KNN
Loan approval
Machine learning

Abstract

Commercial companies highly depend on loan approval models trained by machine learning and statistical methods to predict loan status. However, imbalanced datasets present a key challenge in this sector. Addressing this issue, this paper proposes a new oversampling method based on Fuzzy C means clustering. This clustering algorithm assigns the instances to several groups by assigning a flexible degree of membership. K-Nearest Neighbor and the Decision Tree served as the basic classifiers in an extensive test on the Kaggle loan dataset. Three distinct imbalanced ratios—2.2, 4.2, and 8.45—were used in the experiment. The effectiveness of the recommended strategy was compared with SMOTE and WBOT using 5-fold CV. The outcomes showed that the proposed method outperformed both SMOTE and WBOT, obtaining higher average F-measure and G-mean values across the machine learning algorithms. These findings show how the suggested method may correct class imbalance while also enhancing prediction accuracy in the context of loan acceptance.

Corresponding author
Subrina Akter
Department of Computer Science and Engineering, International Islamic University Chittagong, Chattogram, Bangladesh - 4318. Email: subrinacse@iiuc.ac.bd

1. Introduction

Loan-granting is a crucial operation to increase bank's profitability and maintenance. Even slight advances in default prediction might result in large revenues for banks. However, deciding to approve a loan is difficult and dangerous since banks must correctly forecast defaults to prevent financial losses, particularly during economic crises. According to Thomas, Crook and Edelman (2017) they noticed various factors influencing the rate of default throughout a period, including the price of money (that is, interest rate),

credit availability and need, the status of the country's economy, and the cyclical fluctuations of credit through time. Aside from these factors, the accessibility of data, exactness, and dependability make the default forecast considerably more difficult than other domain-dependent categorization tasks. Furthermore, because the number of loan applicants is massive, it is difficult for bankers to calculate the appropriate risk-to-profitability ratio. When a customer requests a loan, lenders must assess the applicant's solvency before approving the loan. Because the bank's risk tolerance and financial objectives are tied to this issue. As a result, loan approval models play a vital role in assisting bankers in making sound decisions.

However, the real-world scenario is different. The number of classes in a real-world dataset is not equal. Because of these inequalities, the loan approval model cannot produce an accurate result, instead produces a biased output. Recently, ML researchers guided how to handle a data set with imbalances. Haixiang, Yijing, Shang, Mingyun, Yuanyue and Bing (2017) highlight two approaches commonly used in this context: Resampling and Ensemble methods. The resampling strategy focuses on class distribution, whereas the ensemble technique focuses on numerous poor classifiers that work together to create a more effective and sound model. Although these strategies are frequently employed, they are not without flaws. The most basic kind of over-sampling is to replicate random data of the minority group, thus might result in overfitting. The simplest under-sampling strategy includes eliminating arbitrary samples from the dominant class, therefore might result in information loss. On the other hand, previous research (Haixiang, Yijing, Shang, Mingyun, Yuanyue, & Bing, 2017) has shown that ensemble learning algorithms perform better than a single model in the case of skewed datasets. They have been widely applied to dealing with various real-world challenges, such as credit reporting and class imbalance problems. Bagging and boosting are two methods for incorporating basic classifiers into a powerful classifier. The bagging approach is a concurrent ensemble technique that creates the primary classifiers simultaneously whereas the boosting method is a sequential approach that builds the base classifiers sequentially, with the final classifiers impacted by the initial classifiers.

However, the integration of preprocessing methods, dynamic selection strategies, and group generators leads only to the work of De Melo Junior, Nardini, Renso and de Macêdo (2019). This particular study stands out because it incorporates a combination of techniques rather than evaluating individual methods in isolation. The authors of this work explored the effectiveness of preprocessing techniques, dynamic selection methods (automatically selecting the most competent classifiers for prediction), and

clustering generators (generation of diverse taxonomy). One of the important things is, that De Melo Junior, Nardini, Renso and de Macêdo (2019) mainly focused on the comparison of these techniques rather than the methodology of the techniques. Their study delivers very useful information rather than introducing new techniques.

Among the Data-level approaches, such as oversampling (Batista, Prati, & Monard, 2004) and the under-sampling methods (Zhang & Mani, 2003; Li, Zhu, & Wu, 2020) a popular oversampling method is the synthetic minority oversampling technique (SMOTE) (Chawla, Bowyer, Hall, & Kegelmeyer, 2002), which has many applications and also practice in different fields.

The classic SMOTE method and its many extensions (Chawla, Lazarevic, Hall, & Bowyer, 2003; Han, Wang, & Mao, 2005; Sinapiromsaran & Lursinsap, 2009; Tang, Zhang, Chawla, & Krasser, 2008; Bai & Garcia, 2008; Prusty, Jayanthi, & Velusamy, 2017; Douzas, Bacao, & Last, 2018; Susan & Kumar, 2019; Cover, & Hart, 1967) uses the k-Nearest Neighbor (KNN) algorithm to generate new samples. These approaches attempt to aggregate minority class samples by computing the arbitrary difference between a chosen source sample and one of its nearest neighbor samples. Several improved variants have been proposed based on this principle, including borderline SMOTE (Han, Wang, & Mao, 2005), safe-level SMOTE (Bunkhumpornpat, Sinapiromsaran, & Lursinsap, 2009), ADASYN (He, Bai, Garcia, & Li 2008), SMOTE-IPF (Sáez, Luengo, Stefanowski, & Herrera, 2015) and k-means SMOTE (Douzas, Bacao, & Last, 2018). These variations introduced additional considerations to improve the efficiency of the SMOTE technique and overcome potential limitations. However, SMOTE is capable of generating noisy, finite, and repetitive patterns, which can affect the possibility of misclassification of the dominant class that makes up the original data. By using oversampling, Emu, Jahin, Akter, Patwary and Akter (2022) were able to increase the sample size for the minority while preserving the proportions of low classification errors for the group, which kept the majority class. Their method, however, has problems with how well weighting filters out minority data. Their weighting procedure proved to be inadequately effective, which can have a significant impact on assessing the importance of each minority sample. In terms of filtering and adaptive representation for the minority class instance, this strategy does not produce the best results. This constraint emphasizes the need to create reliable and effective methods for handling minority class samples in oversampling approaches.

Considering the aforementioned issues, especially those related to SMOTE and WBOT, this paper offers a novel strategy based on fuzzy

c-means clustering. This oversampling method makes use of fuzzy computing and clustering to boost composite sample production. It allocates data points to various clusters with variable degrees of membership. The proposed strategy tries to fabricate composite examples that are more representative, consequently limiting the effect on the misclassification pace of the ruling classes. An improvement in oversampling adequacy in case of class imbalance has been done by this innovative procedure.

The remaining sections of this study are organized as follows: The literature review is fully reported in Part II, which is followed by Part III on imbalanced data and Part IV on the suggested technique, and Part V on the findings. The work is concluded by part VI with suggestions for more investigation.

2. Related works

In a study, De Melo Junior, Nardini, Renso, and de Macêdo, (2019) conducted extensive research on credit scoring, focusing on four datasets with an imbalance ratio (IR) of 25%. The analysts pointed to judge the value of different methods in tending to class imbalance, counting under-sampling strategies such as Random Under-Sampling (RUS), oversampling methods like Smote and Random Over-Sampling, as well as ensemble strategies like homogeneous selection and dynamic selection. Additionally, they examined the performance of these approaches when combined with pool generators. In contrast to conventional credit valuation approaches, Serrano-Cinca and Gutirrez-Nieto, (2016) proposed an elective strategy that employments the inside rate of return (IRR) rather than the twofold IDs of the default advance. By considering the IRR as a preparation measure, the creators need to adjust the credit rating models with the profit-oriented objectives of monetary education. This procedure offers an exciting viewpoint because it centers on maximizing benefits and not only forecasting. Bastani, Asgari and Namavari (2019) presented a diverse point of view on credit evaluation by evaluating not only the probability of default but also profitability. They offer a two-phase approach that uses a comprehensive and in-depth learning strategy to associate dropout assessment with profitability. By counting productivity as a deciding figure, their approach gives a more comprehensive examination that considers both the risk and potential money related to loan approvals. Sousa, Gama and Brandão (2016) proposed a show for credit chance examination and stated that active models tend to surpass inactive models in terms of estimate exactness. However, despite the outstanding performance of dynamic models, the writers note that ordinary models are still more broadly utilized within the economic industry. This finding

highlights a discrepancy between the potential of dynamic modeling techniques and their actual adoption in practice, suggesting a need for greater recognition and adoption of dynamic approaches in credit risk assessment.

Recently, Song, Wang, Ye, Zaretzki and Liu (2023) suggests a multi-objective ensemble learning approach that is rating-specific for building potential predictions for loan subgroups with comparable default probability. A data augmentation technique was presented by Temraz and Keane (2022), which creates artificial, hypothetical cases in the minority class. This approach uses real feature values—rather than overlaying values across data points. However, this approach was binary class-specific.

More studies have focused on evaluating the competence of classification methods for loan approval datasets. Several papers, including Abellán and Castellano (2017), Ala'raj and Abbod (2016a), Ala'raj and Abbod (2016b), Feng, Xiao, Zhong, Qiu and Dong (2018), García, Marques and Sánchez (2019), He, Zhang, and Zhang (2018), Lessmann, Baesens, Seow and Thomas (2015), Sun, Lang, Fujita and Li (2018), Xia, Liu, Da and Xie (2018), Xia, Liu, Li and Liu (2017), Xiao, Xiao and Wang (2016) have assessed the predictive capabilities of different methods in this domain.

3. Imbalanced dataset and evaluation metric

In this section, we have started by introducing several solution approaches for balancing the imbalanced dataset. Subsequently, the specific metric for evaluating the performance of the imbalanced dataset is also introduced.

3.1 Theory of imbalanced dataset

A dataset is said to be imbalanced if the target class has an unbalanced distribution. In this case, the frequency of one class is much higher than the amount of another. This class imbalance can result in biased models that fare poorly with the minority class, which is often the class of interest. To address this issue, different algorithms have been created particularly for managing imbalanced data sets.

There exist many data-driven solutions that concentrate on data augmentation, in which artificial data from the minority class is manufactured using various approaches. SMOTE (Synthetic Minority Over-sampling approach) is a common data-driven approach that generates synthetic instances by interpolating between existing minority class samples. By increasing the representation of the minority class, data expansion balances the data set and improves the performance of the model in the minority class.

Another data-driven approach is transfer learning, which leverages knowledge from related tasks or datasets to improve performance relative to

the imbalanced dataset. In transfer learning, a model that has been pre-trained on a large and diverse data set is matched to the imbalanced data set to benefit from the learned representations. This transfer of knowledge can lead to more effective learning and better generalization of the minority class.

The algorithm used for learning is either updated or built to accommodate imbalanced data sets in algorithm-level solutions. Another useful option for imbalanced classification is cost-sensitive learning, which focuses on first allocating various costs to the many sorts of misclassification rates that might occur, and then applying specialized algorithms to compensate for those costs. The concept of a cost matrix helps to understand the varied misclassification costs.

In this work, we focused on a data-level approach to addressing the problem of class imbalance. The analysis in this context is concentrated on datasets containing the categories "yes" and "no." The class with a "yes" label stands in for the majority class, whereas the class with a "no" label symbolizes the minority class with fewer cases.

The dataset's imbalanced ratio (IR), which reflects the unequal distribution of occurrences among several categories, is calculated by equation 1. It shows the discrepancy in sample sizes between the majority and minority groups. The imbalanced ratio, for instance, would be 9:1 if a dataset had 1,000 instances with 900 samples from the majority group and 100 samples from the minority group.

During the empirical investigation, the minority class's size intentionally decreased by a factor of 4.22 and 8.45. This reduction resulted in the creation of three distinct datasets from the initial one. Thus, these shortened datasets are used to assess the viability of different calculations and techniques in dealing with the issue of class unevenness.

Two existing techniques as well as the suggested approach were applied to the imbalance loan dataset, which was segmented into three imbalanced ratios, and the performance of each method for each dataset was recorded. The results of this analysis can assist us in making a loan approval model that will be more robust and precise in dealing with a class imbalance.

$$IR = \frac{Total\ samples\ of\ majority\ class}{Total\ samples\ of\ minority\ class} \quad (1)$$

3.2 Evaluation metric

When dealing with imbalanced datasets, accuracy is hardly a valid metric in the context of imbalanced data sets because it does not discriminate between

the amount of properly identified samples of various categories. When one category is substantially more numerous than another in these sorts of datasets, a framework that merely predicts the dominant category for all cases can nonetheless reach excellent accuracy while not adequately identifying the minority class. This is the reason why supplementary measures, such as precision and recall, are so useful in this type of network monitoring. These metrics are important for quickly acquiring knowledge of core patterns throughout a dataset. Observing them periodically will allow us to have a greater understanding of the system's overall efficacy as well as the variables influencing its dependability and usefulness. The specific formulas for these metrics can be found in the following references (He, Bai, Garcia, & Li, 2008; Douzas, Bacao & Last, 2018; Pan, Zhao, Wu, & Yang, 2020; Li & Zhu, 2019; Tang, Zhang, Chawla, & Krasser, 2008; Guo & Viktor, 2004).

Table I
Confusion Matrix

	Predictive Negative	Predictive Positive
Actual Negative	True Negative (TN)	False Positive (FP)
Actual Positive	False Negative (FN)	True Positive (TP)

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{4}$$

F-measure: F-measure offers a balanced metric that incorporates both precision and recall into a single statistic. It has a score between 0 and 1, with 1 reflecting flawless precision and recall. As in prior work (He, Bai, Garcia, & Li, 2008; Douzas, Bacao & Last, 2018; Pan, Zhao, Wu & Yang, 2020), prediction accuracy (ACC), the F-measure with b = 1 and G-mean were employed as assessment measures.
F-measure is calculated as:

$$F - measure = \frac{(1 + \beta) * recall * precision}{\beta^2 * recall * precision} \tag{5}$$

G-mean: G-mean compares performance in both favorable and adverse factors, whereas F-measure combines precision and recall.
The score of G-mean can be calculated as:

$$G - mean = \sqrt{recall * specificity} \tag{6}$$

A higher G-mean score designates balanced performance between the minority and majority classes, while a higher F-measure value shows a greater

accuracy in predicting the minority class.

4. Proposed method

There are three phases in the suggested technique. These steps involve creating an unlabeled dataset, using Fuzzy C means clustering and classification to oversample the original dataset, and then verifying the balanced dataset using the Decision Tree and K-Nearest Neighbor method. The overall methodology is shown in the following figure:

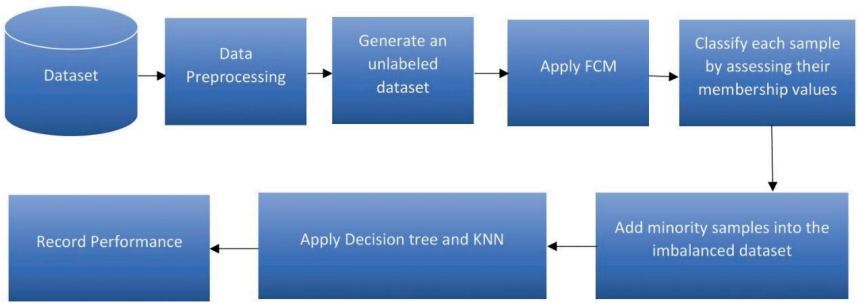

Figure 1
Proposed method

4.1 Generate an unlabeled dataset

In this paper, to generate an unlabeled dataset we applied the method studied by (Emu, Jahin, Akter, Patwary, & Akter, 2022; Jahin, Emu, Akter, Patwary, Bhuiyan, & Miraz, 2021). The unlabeled dataset was created by combining all possible combinations of unique values from different characteristics. This method is designed to generate a diverse and comprehensive set of samples that represent the different combinations of attribute values present in the original dataset.

To illustrate this method, let's consider two features of the loan dataset: Education and Self-employed. Education has two unique values - Graduate and Not Graduate, while Self-employed also has two unique values - Yes and No. Both features have a cardinality of 2, meaning they each have two distinct values.

To create the unlabeled dataset, the authors multiplied the cardinalities of these features, resulting in $2 \times 2 = 4$ possible combinations. The dataset, therefore, consisted of four samples, each represented by a combination of values from Education and Self-employed. The samples in this example would look like this: [(Graduate, Yes), (Graduate, No), (Not Graduate, Yes), (Not Graduate, No)].

Taking into consideration all conceivable combinations of attribute

values, this newly constructed unlabeled dataset provides a complete representation of the samples and allows researchers to study various possibilities and obtain a broad diversity of combinations.

4.2 Apply fcm on unlabeled dataset

Fuzzy C means clustering is then employed to the unlabeled dataset to assign membership values to each data point. Then data is classified based on the highest membership value obtained from the clustering process. This classification helps in identifying the most suitable class for each data point. From these artificially generated data points minority samples are then included in the actual dataset, thereby increasing the representation of the minority class. The formula for calculating the Fuzzy membership value of each sample is:

$$\mu = \frac{1}{\left[\sum_{1}^{n} (d^2ki/d^2kj)^{\frac{1}{m}-1}\right]} \quad (7)$$

Where,

μ= Membership value

n= Number of patterns

m= Fuzzifier value

d_{ik}/d_{kj} signifies Euclidean distance value between cluster and data point.

An example of this process is shown in Table II including 10 samples.

Table II

Predicted class for artificial data using FCM

Data	Membership value for 10 samples		Assigned class
	Class 1(yes/1)	Class 2(NO/0)	
1.	0.9470201	0.0529799	Yes/1
2.	0.9589875	0.0410125	Yes/1
3.	0.0648357	0.9351643	No/0
4.	0.0732844	0.9267157	No/0
5.	0.410113	0.589887	No/0
6.	0.8902665	0.1097335	Yes/1
7.	0.866684	0.1333161	Yes/1
8.	0.097001	0.902999	No/0
9.	0.9216201	0.07838	Yes/1
10.	0.0716716	0.9283285	No/0

Table II shows 10 samples, and for each sample, the membership values for two classes, "Yes"(1) and "No,"(0) are provided. For instance, considering the 1st sample, the membership value for the "Yes" class is 0.9470201, while the membership value for the "No" class is 0.0529799. To fix the assigned class

label for individual data, the membership values are compared. In this case, since the membership score of the "Yes" class (0.9470201) is larger than the membership degree of the "No" class (0.0529799), the assigned class for the 1st sample would be "Yes" or 1.

On the remaining data points in the Table, the same procedure was used. Each sample's membership values for the "Yes" and "No" classes are compared, and the data point was assigned to the cluster with the greater membership value. The original dataset is then updated with the samples from the minority classes. Through this procedure, the minority class's presence in the dataset is effectively increased by the fabrication of synthetic data points that represent it.

The algorithm's stages are as follows:

Algorithm: Fuzzy C-means-based Oversampling Algorithm

Input: Imbalanced loan dataset

Output: Balanced dataset

Steps:

1. Generate an unlabeled dataset (X_unlabeled) by combining unique attribute values.
2. Apply Fuzzy C-means clustering to X_unlabeled and obtain cluster assignments (cluster_labels).
3. Label the unlabeled dataset by assigning majority class labels within each cluster.
4. Oversample the minority class in X_labeled by creating artificial instances based on cluster information.
5. Validate the balanced dataset using DT and KNN classifiers.
6. Return the balanced loan dataset (X_balanced, y_balanced).

End

Figure 2 represents the flowchart of the algorithm.

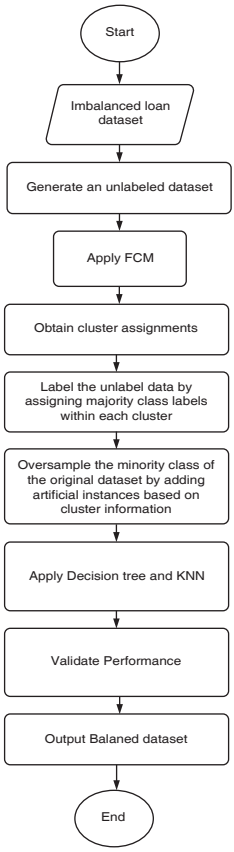

Figure 2
Flow chart of the proposed algorithm

Time Complexity: The time complexity for preparing and testing the classifiers relies upon the calculations utilized and their execution.The time complexit of each step and the total time complexity are shown below:

Step 1: Generate an unlabeled dataset(X_unlabeled) by combining unique attribute values.

Time complexity: O(n * m), where m is the attribute and n is the total data points

Step 2: Apply Fuzzy C-means clustering to X_unlabeled and obtain cluster assignments.

Time Complexity: O(k * n * m * max_iter) , k is the cluster

Step 3: Label the unlabeled dataset by assigning majority class labels within each cluster.

Time Complexity: O(n)

Step 4: Oversample the minority class in X_labeled by creating artificial instances based on cluster information.

Time Complexity: O(n_synthetic * m), n_synthetic is the number of artificial data.

Step 5: Validate the balanced dataset using DT and KNN classifiers.

Time Complexity: O(k * n * m)

Overall Time Complexity:

$$O(n * m) + O(k * n * m * max\_iter) + O(n) + O(n\_synthetic * m) + O(k * n * m) \approx O(k * n * m * max\_iter)$$

4.3 Validation of the newly created balanced dataset

To validate the balanced dataset, two classification algorithms, namely Decision Tree (DT) and K-nearest neighbors (KNN), are used. These algorithms are chosen as they are commonly used for classification tasks and can effectively perform on categorical and numerical data.

The decision tree constructs a classification framework in the shape of a tree. It progressively divides a dataset into progressively smaller sections while also developing a linked decision tree. The outcome of this process is a tree containing decision points and terminal nodes (Jiao, Song & Liu, 2020). Furthermore, KNN acts by locating the K nearest points throughout the training dataset and using their class to forecast the category of a fresh query instance. Because of having capability to handle complicated data and its simple nature, KNN has grown into a prominent tool in the field of machine learning (Cover & Hart, 1967).

The Decision Tree and K-nearest neighbor algorithms both act as validation approaches for determining the efficacy of the altered data set. They show how the dataset was assessed in terms of classification accuracy, precision, recall, and other key parameters.

5. Experimental setup

This part examines the experimental outcomes of the loan dataset in detail, as well as the classification performance of two ML algorithms named Decision Tree and K-Nearest Neighbor with imbalanced data sets. A table of comparisons among SMOTE, WBOT, and the suggested approach is also illustrated with proper justification.

5.1 Data and experimental settings

The tests were performed on a PC running Windows 10, 64-bit, with an Intel Core i5-4590 CPU running at 3.30 GHz and 4 GB of RAM, as well as Python 3.8.8 software. Table III shows the features of the data set used in

this study. The dataset is readily accessible through Kaggle (https://www.kaggle.com/datasets/vikasukani/loan-eligible-dataset). The quantity of dominant samples is 422 while the quantity of minority samples is 192. Since the data set employed has a sufficient number of observations, it was divided into a training (80%) and a test (20%) set. This test set will stay unaltered in the analysis phase. We have employed the DT and KNN as a core classifier in our experiments. In general, we conducted three trials, which are described briefly below:

Trial A: Performance of the classifiers with the imbalanced datasets (Table IV).

Trial B: Comparison of the oversampling techniques in terms of Evaluation metric (Table V).

Trial C: Computational efficacy (Table VI)

Table III

Characteristics of the loan dataset

Attribute	Characteristics of the attribute
Loan_ID	Categorical
Gender	Categorical
Married	Categorical
Dependents	Ordinal
Education	Ordinal
Self_Employed	Categorical
ApplicantIncome	Numerical
CoapplicantIncome	Numerical
LoanAmount	Numerical
Loan_Amount_Term	Numerical
Credit_History	Numerical
Property_Area	Ordinal
Loan_Status	Categorical

Table III describes the features of a loan dataset. The client-based loan eligibility procedure is covered in detail in the dataset. The attributes of this dataset include Loan_ID, Gender, Married, Dependents, Education, Self_Employed, ApplicantIncome, CoapplicantIncome, LoanAmount, Loan_Amount_Term, Credit_History, Property_Area and Loan_Status. 'Loan_ID' is the unique identification number for each loan, and 'Gender' denotes the loan applicant's gender. It might be a male or female. The married column represents the loan applicant's marital status while the feature 'Dependent' refers to the number of ordinal dependents of the loan applicant. The educational background of the loan applicants is also included in the dataset's 'Education' column. 'Self-employed' signifies the loan applicant's employment status. 'Applicant Income' reflects the loan

applicant's income (in thousands), while 'Coapplicant Income' denotes the co-applicant's income (in thousands). 'LoanAmount' refers to the loan amount sought (in thousands of dollars) and 'Loan_Amount_Term' refers to the loan's term (in months). Credit History of the loan applicants is also required which is recorded in the 'Credit_history' column. 'Property_Area' is used to represent the location type of the loan applicant's property. 'Loan_Status' is the approval status of the loan, which is yes or no. Five of the attributes are of categorical type, five are numeric, and three features are ordinal.

5.2 Preprocessing

Preprocessing the Loan eligibility prediction dataset entails various processes to organize the data for constructing predictive models. Here's a quick rundown of the pre-processing processes that are often used with the Loan eligibility prediction dataset:

1) For numerical variables, missing values were handled using the mean or median, and for categorical variables, the mode.
2) Outliers were identified using IQR and addressed by the capping technique.
3) The label encoder was used to transform categorical values into numerical values.
4) The feature Loan_ID was removed from the dataset since it is only an ID and does not affect the prediction of loan status.

The dataset is now ready to utilize after preprocessing, with an imbalanced ratio of 2.20. However, for this empirical investigation, the minority class was intentionally lowered by a ratio of 4.22 to 8.45. As a consequence of this elimination, three data sets for the actual data set were formed. The outcomes of these assessments ought to give us a clear picture of the overall impact of the evaluation metric on both minority and majority groups. Every trial employed 5-fold cross-validation.

5.3 Experimental results

Trial A: Performance of the classifiers with the imbalanced datasets.

This section presents the classifiers' performances. The dataset was first separated into three imbalanced ratios: 8.45, 4.22, and 2.2. Following the classifiers' application to these datasets, Table IV presents the classifiers' performance.

Table IV
Result of experiment A

IR	Classifiers	Recall	Precision	F-measure	G-mean	Accuracy
8.45	DT	0.102	0.820	0.181	0.353	0.86
8.45	KNN(k=5)	0.126	0.684	0.212	0.34	0.59
4.22	DT	0.126	0.756	0.216	0.380	0.72
4.22	KNN(k=11)	0.156	0.69	0.254	0.373	0.62
2.2	DT	0.452	0.788	0.574	0.595	0.80
2.2	KNN(k=18)	0.162	0.704	0.263	0.380	0.69

In Table IV, when the Imbalanced Ratio was 8.45, the recall of the DT was very poor (0.102) but its precision and accuracy were pretty excellent, 0.820 and 0.86, respectively. This is because the frequency of data points from the majority group was significantly more than the frequency of data points from the minority group. However, the lower F-measure (0.181) and G-mean (0.353) values indicate that the accuracy of the minority class is relatively poor, and the classifier failed to perform well at all on minority and majority class data. Except for poor precision (0.684) and accuracy (0.59), the same situation was provided in the case of KNN. These are the result of having the fewest number of nearest neighbors (k=5).

When the IR is 4.22 then the value F-measure and G-mean of both classifiers were increased, indicating that the outcomes of the models were marginally enhanced.

In case of the original dataset having IR of 2.2 then the score of the F-measure and G-mean for both classifiers were increased but the performance remains unsatisfactory. Following these studies, three oversampling approaches were applied to the original dataset with an IR of 2.2 and recorded the performance of the base classifiers in Table V.

Trial B: Comparison of the oversampling techniques in terms of Evaluation metric

Following the documentation of the classifiers' performance on the imbalanced data sets, the original dataset with an IR of 2.2 underwent three oversampling techniques. The experiments were done using 5-fold cross-validation and the average accuracy is recorded in Table V.

Table V
Result of experiment B

Evaluation metric	DT_SMOTE	DT_WBOT	DT_Proposed method	KNN_SMOTE	KNN_WBOT	KNN_proposed method
F-measure	0.861	0.861	0.941	0.856	0.876	0.916
G-mean	0.861473	0.841881	0.951	0.858	0.879	0.889
Accuracy	0.855	0.846	0.902	0.878	0.870	0.892

In Table V, although SMOTE and WBOT performed equally in terms of F-measure and G-mean, SMOTE's average accuracy was somewhat higher than WBOT's. Because the number of majority classes in WBOT is always larger than the number of minority classes. However, all performance metrics show that the accuracy of the minority group grew greatly, whereas the accuracy of the majority class's accuracy declined in SMOTE due to the issue of noisy data (Bunkhumpornpat, Sinapiromsaran, & Lursinsap, 2009). Furthermore, Table V demonstrates that the suggested model's average Accuracy, F-measure, and G-mean were greater than those of SMOTE and WBOT. The findings show that this strategy worked well for oversampling an imbalanced data set.

The high score of G-mean implies that the classifier works well on the data of minority and majority groups. Furthermore, a high F-measure value signifies greater accuracy in the minority class.

Figure 3 and Figure 4 compare the performance of the proposed method against previous methods.

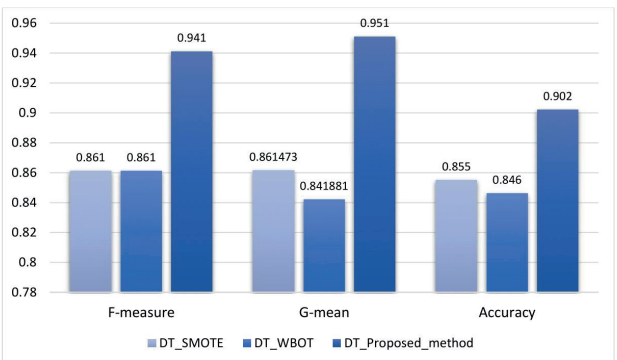

Figure 3

Performance comparison of the proposed method with previous methods based DT.

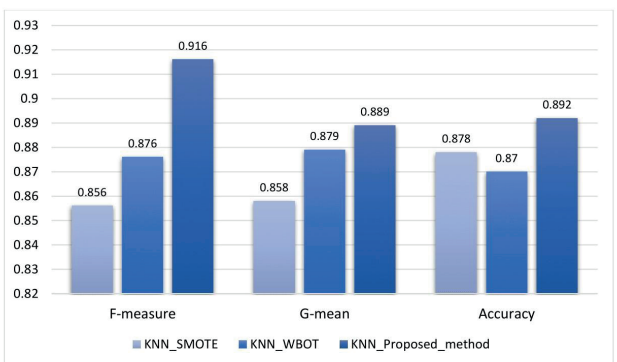

Figure 4

Performance comparison of the proposed method with previous methods based on KNN.

From both figures, it has been observed that in both cases the proposed algorithm works well.

Experiment C: Computational Efficiency

To assess the computational effectiveness of each classifier, we utilized the average running time of five runs. The decision tree was pruned using the cost complexity pruning approach. As a result, the depth of the tree was lowered which reduced the running time. When it comes to the K-Nearest Neighbor, additional time was required to obtain the best value of k. Table VI demonstrates the average running time of the classifiers employing three oversampling strategies.

Table VI

Average running time of the methods(sec)

DT_SMOTE	DT_WBOT	DT_proposed method	KNN_SMOTE	KNN_WBOT	KNN_proposed method
0.15	0.14	0.12	0.23	0.20	0.20

Table VI shows that the proposed model takes less time than the previous model for completing the 5-fold cross-validation. Figure 5 and Figure 6 show the comparison of the run time of the proposed method against previous methods. In both cases, KNN_proposed method takes less time than KNN_SMOTE and KNN_WBOT and the DT_proposed method takes less time than DT_SMOTE and DT_WBOT. That is, the average run time of the proposed method require less time than that of others.

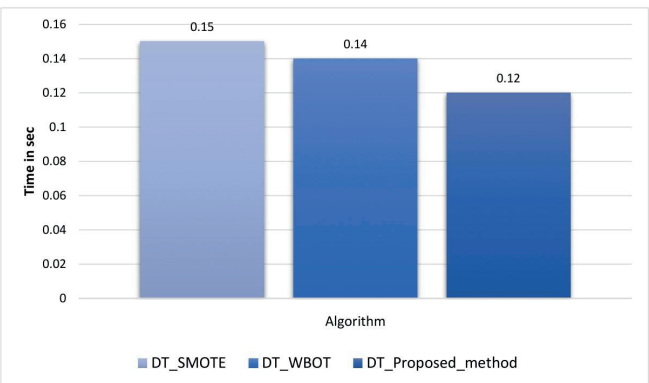

Figure 5

Comparison of the run time of the proposed method with other methods based on DT.

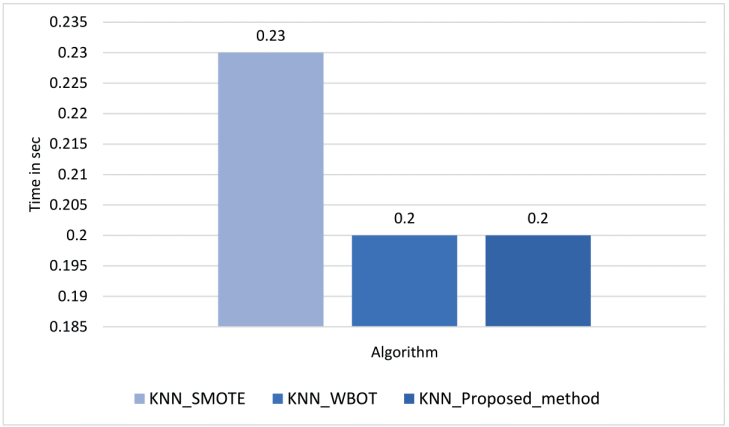

Figure 6

Comparison of the run time of the proposed method with other methods based on KNN.

6. Conclusion

The suggested solution eliminates the class imbalance problem of a loan dataset through a series of phases. First, an unlabeled dataset is created by combining all possible combinations of the unique value of each attribute. Fuzzy C means assigns membership values to each sample. The minority samples are chosen based on their membership values. The artificially generated data points representing the minority class are then merged into the original dataset. The modified dataset is then tested for performance and accuracy using two well-known classifiers, Decision Tree (DT) and K-Nearest Neighbours (KNN). Utilizing appropriate metrics like accuracy, F-measure, G-mean, and other pertinent estimates, the effectiveness of the classifiers is evaluated. To prove the recommended method's effectiveness, its performance is contrasted with that of two popular oversampling techniques, SMOTE and WBOT. According to the study's findings, SMOTE and WBOT are not as effective as the suggested oversampling strategy. This demonstrates that the FCM-based approach provides a more effective way of coping with imbalanced data sets and boosts the classifiers' overall accuracy and predictive ability. To ensure that the provided method can be used for large datasets without a lot of computational overhead, the computational efficiency of the method is also evaluated. Using a wider variety of credit datasets with various sets of imbalanced ratios, future studies could examine the generalizability and efficacy of the proposed methods in other contexts.

References

Abellán, J., & Castellano, J. G. (2017). A comparative study on base classifiers in ensemble methods for credit scoring. *Expert Systems with Applications*, *73*, 1-10. doi.: http/doi.org/10.1016/j.eswa.2016.12.020

Ala'raj. M., & Abbod, M. F. (2016a). A new hybrid ensemble credit scoring model based on classifiers consensus system approach. *Expert Systems with Applications*, *64*, 36-55. https://doi.org/10.1016/j.eswa.2016.07.017

Ala'raj. M., & Abbod, M. F. (2016b). Classifiers consensus system approach for credit scoring. *Knowledge-Based Systems*, *104*, 89-105. https://doi.org/10.1016/j.knosys.2016.04.013

Bastani, K., Asgari, E., & Namavari, H. (2019). Wide and deep learning for peer-to-peer lending. *Expert Systems with Applications*, *134*, 209-224. https://doi.org/10.1016/j.eswa.2019.05.042

Batista, G. E. A. P. A., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter, 6*(1), 20–29.

Bunkhumpornpat, C., Sinapiromsaran, K., & Lursinsap, C. (2009, April 27-30). Safe-level-SMOTE: S*afe-level-synthetic minority over-sampling technique for handling the class imbalanced problem.* Paper presented at the Pacific Asia Conference on Knowledge Discovery and Data Mining, Bangkok, Thailand.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority oversampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953

Chawla, N. V., Lazarevic, A., Hall, L. O., & Bowyer, K. W. (2003, September 22-26). *SMOTEBoost: Improving Prediction of the Minority Class in Boosting.* Paper presented at the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases, Cavtat-Dubrovnik, Croatia.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory, 13*(1), 21–27. https://doi.org/10.1109/TIT.1967.1053964

De Melo Junior, L. S., Nardini, F. M., Renso, C., & de Macêdo, J. A. F. (2019, December 16-19). *An empirical comparison of classification algorithms for imbalanced credit scoring datasets.* Paper presented at 2019 18th IEEE International Conference on machine learning and Applications (ICMLA), Boca Raton, FL, USA.

Douzas, G., Bacao, F., & Last, F. (2018). Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE. *Information Science*, *465*, 1–20. https://doi.org/10.1016/j.ins.2018.06.056

Emu, I. J., Jahin, D., Akter, S., Patwary, M. J., & Akter, S. (2022, February 26-27). *A novel technique to solve class imbalance problem.* Paper presented in 2022 International Conference on Innovations in Science, Engineering and Technology (ICISET), Chittagong, Bangladesh.

Feng, X., Xiao, Z., Zhong, B., Qiu, J., & Dong, Y. (2018). Dynamic ensemble classification for credit scoring using soft probability. *Applied Soft Computing*, *65*, 139-151. https://doi.org/10.1016/j.asoc.2018.01.021

García, V., Marques, A. I., & Sánchez, J. S. (2019). Exploring the synergetic effects of sample types on the performance of ensembles for credit risk and corporate bankruptcy prediction. *Information Fusion*, *47*, 88-101. https://doi.org/https://doi.org/10.1016/j.inffus.2018.07.004

Guo, H., & Viktor, H. L. (2004). Learning from imbalanced data sets with boosting and data generation: the databoost-im approach. *ACM Sigkdd Explorations Newsletter*, *6*(1), 30-39 https://doi.org/10.1145/1007730.100773

Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., & Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, *73*, 220-239. https://doi.org/10.1016/j.eswa.2016.12.035

Han, H., Wang, W. Y., Mao, B. H. (2005). Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. In: Huang, DS., Zhang, XP., Huang, GB. (eds) Advances in Intelligent Computing. ICIC 2005. Lecture Notes in Computer Science, vol 3644. Springer, Berlin, Heidelberg. Retrieved from https://doi.org/10.1007/11538059_91

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008, June 1-8). *ADASYN: Adaptive synthetic sampling approach for imbalanced learning.* Paper presented at the 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), Hong Kong, (pp. 1322-1328).

He, H., Zhang, W., & Zhang, S. (2018). A novel ensemble method for credit scoring: Adaption of different imbalance ratios. *Expert Systems with Applications*, *98*, 105-117. Doi: 10.1016/j.eswa.2018.01.012

Zhang, J., & Mani, I. (2003). *KNN approach to unbalanced data distributions: a case study involving information extraction*. Paper presented in Proceedings of the workshop on learning from imbalanced datasets , ICML, Washington, DC.

Jahin, D., Emu, I. J., Akter, S., Patwary, M. J., Bhuiyan, M. A. S., & Miraz, M. H. (2021, December 9-10). *A novel oversampling technique to solve class imbalance problem: A case study of students' grades evaluation.* Paper presented in 2021 International Conference on Computing, Networking, Telecommunications & Engineering Sciences Applications (CoNTESA), Tirana, Albania.

Jiao, S. R., Song, J., & Liu, B. (2020, August 21-23). *A review of decision tree classification algorithms for continuous variables.* Paper presented in the 2020 second International Conference on Artificial Intelligence Technologies and Application (ICAITA) Dalian , China.

Sáez, J. A., Luengo, J., Stefanowski, J., & Herrera, F. (2015). SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering. *Information Sciences, 291,* 184-203. https://doi.org/10.1016/j.ins.2014.08.051

Lessmann, S., Baesens, B., Seow, H.V., & Thomas, L.C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research, 247*(1), 124-136. https://doi.org/10.1016/j.ejor.2015.05.030

Li, J., &Zhu, Q. (2019). A semi-supervised self-training method based on an optimum-path forest. *IEEE Access, 7*, 36388-36399. https://doi.org/10.1109/ACCESS.2019.2903839

Li, J., Zhu, Q., & Wu, Q. (2020). A parameter-free hybrid instance selection algorithm based on local sets with natural neighbors. *Applied Intelligence, 50*(5), 1527–1541.https://doi.org/10.1007/s10489-019-01598-y

Pan, T., Zhao, J., Wu, W., & Yang, J. (2020). Learning imbalanced datasets based on SMOTE and gaussian distribution. *Information Sciences*, *512*, 1214-1233. https://doi.org/10.1016/j.ins.2019.10.048

Prusty, M. R., Jayanthi, T., & Velusamy, K. (2017). Weighted-SMOTE: A modification to SMOTE for event classification in sodium-cooled fast reactors. *Progress in Nuclear Energy*, *100*, 355–364. https://doi.org/10.1016/j.pnucene.2017.07.015

Serrano-Cinca, C., & Gutiérrez-Nieto, B. (2016). The use of profit scoring as an alternative to credit scoring systems in peer-to-peer (P2P) lending.

Decision Support Systems, *89*, 113-122. https://doi.org/10.1016/j.dss.2016.06.014

Sousa, M. R., Gama, J., & Brandão, E. (2016). A new dynamic modeling framework for credit risk assessment. *Expert Systems with Applications*, *45*, 341-351. https://doi.org/10.1016/j.eswa.2015.09.055

Sun, J., Lang, J., Fujita, H., & Li, H. (2018). Imbalanced enterprise credit evaluation with DTE-SBD: Decision tree ensemble based on SMOTE and bagging with differentiated sampling rates. *Information Sciences*, *425*, 76-91. https://doi.org/10.1016/j.ins.2017.10.017

Susan, S., & Kumar, A. (2019). SSOMaj-SMOTE-SSOMin: Three-step intelligent pruning of majority and minority samples for learning from imbalanced datasets. *Applied Soft Computing, 78*, 141–149. https://doi.org/10.1016/j.asoc.2019.02.028

Tang, Y., Zhang, Y. Q., Chawla, N. V., & Krasser, S. (2008). SVMs modeling for highly imbalanced classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *39*(1), 281-288. https://doi.org/10.1109/TSMCB.2008.2002909

Thomas, L., Crook, J., & Edelman, D. (2017). *Credit scoring and its applications*. [Review of the book of Society for Industrial and Applied Mathematics].

Xia, Y., Liu, C., Da, B., & Xie, F. (2018). A novel heterogeneous ensemble credit scoring model based on bstacking approach. *Expert Systems with Applications*, *93*, 182-199. https://doi.org/10.1016/j.eswa.2017.10.022

Xia, Y., Liu, C., Li, Y., & Liu, N. (2017). A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. *Expert Systems with Applications*, *78*, 225-241. https://doi.org/10.1016/j.eswa.2017.02.017

Song, Y., Wang, Y., Ye, X., Zaretzki, R., & Liu, C. (2023). Loan default prediction using a credit rating-specific and multi-objective ensemble learning scheme. *Information Sciences*, *629*, 599-617. https://doi.org/10.1016/j.ins.2023.02.014

Temraz, M., & Keane, M. T. (2022). Solving the class imbalance problem using a counterfactual method for data augmentation. *Machine Learning with Applications*, *9*, 100375. https://doi.org/10.1016/j.mlwa.2022.100375