Journal of Information & Optimization Sciences
ISSN 0252-2667 (Print), ISSN 2169-0103 (Online)
Vol. 47 (2026), No. 6, pp. 2319–2343
DOI : 10.47974/JIOS-2141

Boosting inspired overlap reduction : A novel approach for reducing overlap in imbalanced classification problems

Pooja Tyagi *
Jaspreeti Singh †
Anjana Gosain §
University School of Information, Communication and Technology
Guru Gobind Singh Indraprastha University
Dwarka 110078
New Delhi
India

Abstract

The classification of imbalanced datasets has garnered a significant research attention over the past decades. Although, numerous solutions have been proposed to address imbalanced classification problems, recent research suggests that class imbalance alone does not significantly degrade learning performance. Rather, class overlap, which has often been overlooked in previous studies has an even greater negative impact on the performance of learning algorithms. This paper proposes a Boosting-Inspired Overlap Reduction (BiOR) algorithm to handle the problem of class overlap in imbalanced datasets. BiOR employs a variance-based adaptive threshold mechanism, where the overlap threshold is dynamically determined as a function of the variance of margins in both majority and minority classes. By iteratively adjusting weights of samples, BiOR ensures selective removal of highly overlapping majority instances, thereby enhancing class separability without excessive data loss. The effectiveness of BiOR is evaluated on 20 benchmark real world datasets as well as 16 synthetic datasets, demonstrating its superior classification performance over existing resampling techniques.

Subject Classification: *68T01.*

Keywords: *Class imbalance, Class overlap, Machine learning, Classification, SMOTE, Random forest classifier.*

* *E-mail:* **pooja.tyagi94@gmail.com (Corresponding Author)**
† *E-mail:* **jaspreeti_singh@ipu.ac.in**
§ *E-mail:* **Anjana_gosain@ipu.ac.in**

1. Introduction

Classification is a cornerstone of machine learning widely employed in diverse application domains such as medical diagnosis, credit risk analysis, anomaly detection and fraud detection etc. [1]. A persistent challenge in classification tasks is class imbalance, where one or more classes are heavily underrepresented (the minority class) than others (the majority class) [2] as shown in Fig. 1(a). Conventional machine learning models tend to be biased toward the majority class (having more instances), potentially leading to inaccurate or unreliable predictions for the minority class. This imbalance can result in elevated false-negative rates, which may be detrimental especially in scenarios where the minority class signifies critical outcomes, such as diagnosing diseases, detecting frauds, fault prediction etc. [3] [4].

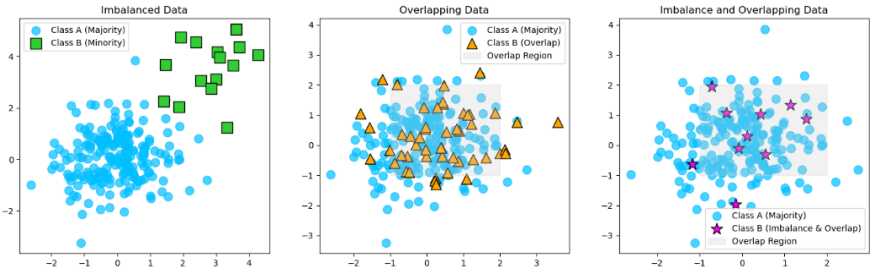

Figure 1

a) Representation of imbalanced data; b) Overlapping data and c) Imbalanced and overlapped data

The majority of solutions tackling this issue typically fall into three broad categories namely data level solutions, algorithm level techniques and ensemble methods [5]. At the data level various sampling techniques like undersampling, oversampling and hybrid sampling are employed to rebalance the dataset [6]. Undersampling techniques remove the majority class instances, oversampling techniques increase the minority class instances while the hybrid methods combine these approaches [7][8]. Algorithm level techniques adapt the learning algorithm itself to account for imbalance by assigning higher misclassification penalties to minority instances [9-11]. Lastly, ensemble methods build multiple classifiers to improve generalization, with techniques like boosting and bagging tailored to handle imbalanced data by iteratively assigning different weights to the majority and the minority samples [12].

While class imbalance is commonly assumed to cause significant degradation in classifier performance, it has been convinced in the literature that apart from class imbalance, presence of overlap between the classes is another factor that degrades the classification performance [2] [13-17]. A dataset is considered "overlapped" when instances from different classes occupy the same

region in the feature space [18-20] as shown in Fig 1(b). Although these instances may share similar characteristics, they belong to different classes, posing a significant challenge for classification tasks during both training and inference. Overlapping data not only complicates the definition of decision boundaries but also reduces the efficacy of traditional class imbalance-handling methods [21-23]. In such overlapping regions as shown in Fig.1(c), the majority class tends to dominate because it appears more frequently to the learning algorithm[24]. Consequently, the decision boundary shifts toward the majority class, often resulting in the misclassification of minority-class instances situated near the overlap—an outcome that is undesirable in practical applications [25].

Most of the existing research typically tackles the problem of class imbalance and class overlap separately. However, in numerous real-world scenarios (e.g., credit card fraud detection, fault detection, etc.), these issues frequently appear simultaneously. While the traditional boosting algorithms focus on adjusting the weights of misclassified instances to improve overall accuracy, they generally do not explicitly target the removal or mitigation of class overlap. In [26] an overlap based undersampling was proposed where the majority class instances in the overlapped region were identified for elimination. However, this algorithm may cause excessive sample elimination, leading to information loss [27]. An another research [28] introduced the Random Forest Cleaning Rule (RFCL), an undersampling technique designed to address the challenges of both class imbalance and overlap in binary classification. By leveraging the concept of margins derived from a random forest model, RFCL identifies and removes majority class instances crossing a newly determined classification boundary, optimized for maximizing F1-score. This research also devised a metric to quantify the overlapping degree. However, the algorithm performed poorly on datasets with low overlap, where it provided no significant improvement in comparison to no under-sampling. While the RFCL algorithm and its margin threshold selection method help identify and remove overlapping instances, the threshold is determined globally for the entire dataset in a static and single-round fashion. This may not accommodate local variations within the dataset, where different regions of the feature space might benefit from different thresholds.

In order to address these gaps in the literature, this study proposes Boosting-Inspired Overlap Reduction (BiOR) algorithm which leverages an iterative, adaptive and margin-based framework to effectively manage overlapping regions in imbalanced datasets. The core principle of BiOR lies in its margin based overlap detection combined with boosting inspired weight updation mechanism [29, 42]. It repeatedly re-estimates margins using random forest as classifier and selectively removes the majority class instances that contribute to the overlap [30][31]. The instances are removed based on the overlapping degree (OD), which is compared against a dynamically computed threshold in each iteration. This iterative refinement process allows BiOR to

gradually enhance class separability, leading to improved classification performance.

This work makes the following key contributions:

1. Addresses the problems in classification caused by both overlapping of classes and the imbalance between them.
2. Proposes a boosting driven approach called, Boosted Overlap Reduction (BiOR), that iteratively refines margins, dynamically updates sample weights, and selectively eliminates highly overlapping majority-class instances over multiple training rounds.
3. Evaluates the classification performance of BiOR against 13 state-of-art sampling techniques on 20 real-world datasets and 16 synthetic datasets varying in imbalance and overlap degrees. The experimental results demonstrate the effectiveness of the proposed technique in terms of AUC and F1-measure.

The remainder of this paper is organized as follows: Section 2 provides an overview of related work in class imbalance and overlap reduction. Section 3 details the proposed BiOR algorithm. Section 4 presents the materials and methods used in this study followed by the experimental setup and results and discussion in the later sections. Finally, the conclusion and future directions have been outlined in the last section.

2. Literature Review

Class imbalance and class overlap are critical challenges in machine learning that significantly degrade classifier performance, particularly in complex data domains. Garcia et al. [2] were among the first to analyse the impact of class overlap in conjunction with class imbalance. Their study revealed that class overlap has a more pronounced effect on classification performance compared to imbalance, particularly in boundary regions where the distinction between classes becomes unclear. Later, in 2008, García, Mollineda, and Sánchez [32] conducted an in-depth analysis of the k-Nearest Neighbors (k-NN) classifier under class imbalance and overlap conditions. Their study demonstrated that k-NN struggles when the majority class dominates overlapping regions, leading to the misclassification of minority instances.

Furthering this line of search, Borsos, Lemnaru, and Potolea [33] introduced the Augmented R-value, a novel metric designed to quantify the compounded effects of imbalance and overlap. This metric, combined with a meta-learning approach, underscored the need for classifiers specifically tailored to overlap-prone regions. Expanding on decomposition-based strategies, Sáez, Galar, and Krawczyk, [34] introduced the One-vs-One (OVO) decomposition strategy to mitigate class overlap in multi-class classification. Their approach involved breaking down multi-class classification into multiple binary sub-problems, effectively reducing the impact of overlapping instances in individual classifiers.

Guzmán-Ponce et al. [35] also contributed to the field by proposing a two-stage under-sampling technique leveraging DBSCAN clustering and a Minimum Spanning Tree (MST) algorithm to improve the identification and handling of overlapping data points. Similarly, Vuttipittayamongkol and Elyan [26] proposed the Overlap Based Undersampling (OBU) method that utilized soft clustering, i.e., Fuzzy C-Means to detect and eliminate the overlapped instances. In their another research in 2020 [27], the authors introduced a neighborhood-based undersampling framework using k-NN to remove majority class instances in overlapping regions while enhancing minority class visibility. Their approach significantly improved sensitivity and reduced classification bias toward the majority class. Further in [36] the authors introduced Adaptive-Threshold OBU (AdaOBU) and Boosted OBU (BoostOBU) as improvements over OBU. AdaOBU dynamically adjusted the elimination threshold based on the dataset's overlap degree, while BoostOBU enhanced the minority-class representation before majority-class removal. In another research, Nwe and Lynn [37] developed K-US, a KNN-based under-sampling method that removes majority class samples from overlapping regions while preserving the structural integrity of the minority class.

Recognizing the need for classifier specific strategies, Yuan et al. [38] introduced the Overlap and Imbalance Sensitive Random Forest (OIS-RF), which integrated a Hard-to-Learn (HTL) coefficient to prioritize difficult instances and used weighted bootstrap sampling to address both overlap and imbalance. In 2022, the authors Zhang and Han in [39] introduced the Imbalanced Overlapping Random Forest (IORF), an ensemble-based method combining weighted bootstrap sampling and Gaussian perturbation.

Seoane Santos et al. [13] conducted a comprehensive review of two decades of research on class imbalance and overlap, presenting taxonomies for characterizing class overlap and imbalance. Their work highlighted the limitations of the existing methods, discussed about the classifier biases and the need for a unified framework to address these issues. Their findings reinforced that class overlap is a more pressing issue than imbalance in many classification tasks. Later, in their another research in 2023, the authors refined previous taxonomies of overlap complexity measures, reaffirming that class overlap is a more critical issue than imbalance in real-world high-dimensional data. Their study emphasized the need for standardized definitions and tools for quantifying overlap [14].

A more recent development by Islam et al. [40] is the Credit Card Anomaly Detection (CCAD) model. This ensemble learning approach is designed for fraud detection in imbalanced and overlapping credit card transaction data. The model integrated multiple outlier detection algorithms with XGBoost to improve fraud detection in highly imbalanced and overlapping datasets. Further, Kumar et al. [15] conducted a comprehensive survey on class overlap handling methods in imbalanced datasets. They categorized existing

approaches into data-level, algorithm-level, ensemble-based, and hybrid methods, summarizing their respective strengths and limitations. Later, Jubair et al. [41] introduced the Generated Overlapping Samples (GOS) technique, which leveraged an overlapping degree metric and a transformation matrix to generate synthetic samples while preserving boundary information. Their method showed significant improvements in accuracy and F1-score.

3. Proposed Method

This section introduces the Boosting-Inspired Overlap Reduction (BiOR) algorithm which mitigates the problem of class overlap thereby improving the classification performance under imbalanced conditions. For the purpose of clarity, this paper focusses on the binary classification. Figure 2 illustrates the framework of BiOR algorithm. The novelty of BiOR lies in its fusion of boosting-like iterative refinement with a margin-based strategy to detect and remove overlapping majority-class instances. Unlike conventional boosting algorithms that focus on correcting misclassified instances, BiOR enhances the decision boundaries by selectively removing overlapping majority class instances and dynamically adjusting the weights of the instances.

Standard boosting algorithms focus primarily on misclassified instances to improve overall accuracy but do not explicitly address the overlap between classes. BiOR algorithm overcomes this limitation by iteratively identifying and mitigating overlap. Instead of repeatedly increasing the weights of all misclassified minority instances, BiOR measures the modified margin (called margin henceforth in the text) of each instance, which quantifies the classifier's confidence in its predictions as given in Equation 1 [28].

For each instance x, the margin is defined as :

$$M_i = \frac{v_1 - v_0}{v_1 + v_0} \tag{1}$$

where v_1, v_0 are votes for class 0 and 1 respectively

In this study, the margin lies in the range [-1,+1]. Margin quantifies how strongly the model favours the true class over the opposite class. A margin close to +1 implies the model is very confident that the instance belongs to the true class while a margin close to -1 implies the model is confident in the opposite class which would typically mean a significant misclassification or overlapping instance. Thus, margin helps in detecting overlapping instances and guide their removal for improving minority class recognition. After computing margins for all instances, the algorithm considers the distribution of margins in each class. By leveraging margin-based quantiles, BiOR uses the metric of overlapping degree (OD) defined in Equation 2 [28] to quantify overlap in each iteration. OD summarizes how well-separated the classes are based on where their margin distributions lie relative to each other. In this study, on the basis of OD, datasets have been grouped in the following categories as:

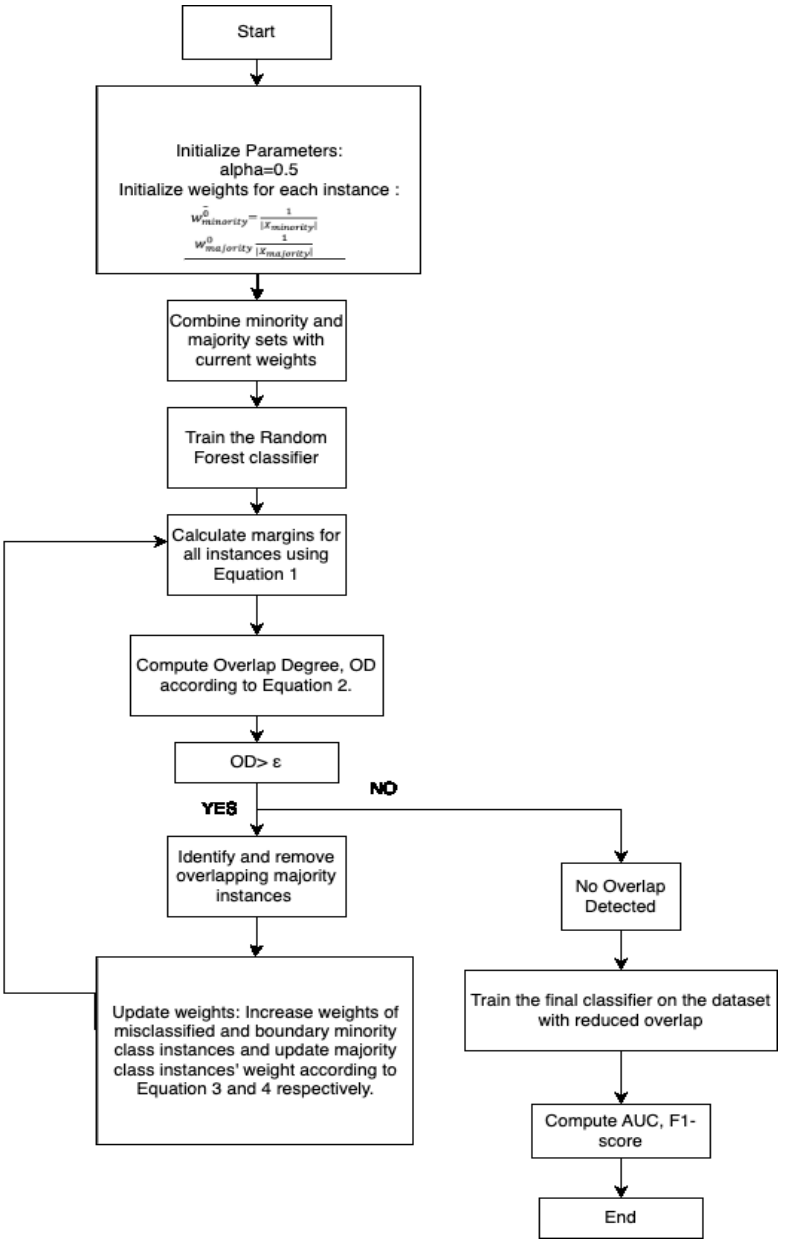

Figure 2
An overview of the proposed BiOR technique.

a) Significantly Overlapped Datasets: with absolute values of OD greater than or equal 1, i.e., $|OD| \geq 1$.
b) Moderately Overlapped Datasets where $0.5 \leq |OD| < 1$.
c) Lowly Overlapped Datasets where $0.1 \leq |OD| < 0.5$.
d) Rarely Overlapped Datasets where $0.1 < |OD|$.

If OD exceeds the dynamically computed overlap threshold (ϵ), the algorithm deems the classes to be overlapping in certain regions. In response, the most overlapping instances in the majority class are then selectively removed from the training set, thereby reducing confusion in the decision boundary and allowing the classifier to more effectively learn minority class characteristics.

$$OD_0 = Q3_{minority} - Q1_{majority} \quad (2)$$

where Q3 is the third quartile of minority margins and Q1 is first quartile of majority margins

The overlap threshold (ϵ) is adaptively defined according to Equation 3. It is dynamically computed using α and serves as a stopping criterion for the algorithm. α is a key parameter that controls ϵ, hence regulating the extent of removal of majority class instances.

$$\epsilon_m = \alpha \times (\sigma^2_{minority} + \sigma^2_{majority}) \quad (3)$$

where α is a parameter controlling the degree of overlap allowed before instance removal is initiated.

$\sigma^2_{minority}$: is the variance of minority class

$\sigma^2_{majority}$: is the variance of majority class

It scales the sum of variance of minority and majority class margins. A higher α results in a more aggressive removal of overlapping instances, while a lower α retains more data. In this study, the selection of α is based on empirical evaluation across multiple datasets to achieve an optimal F1 score which was found to be 0.5. The number of iterations is determined dynamically by an early stopping criterion i.e., the process terminates when the overlap degree becomes sufficiently small compared to ϵ_m [40]. This ensures that the algorithm only runs for as many iterations as necessary to reduce overlap, preventing unnecessary computations.

The weights for minority instances are increased for misclassified samples or those near the decision boundary (low or negative margin) as given in Equation 4.

$$w^{m+1}_{\text{minority}} = w^{m}_{\text{minority}} \times e^{-M_m(y=0)} \quad (4)$$

The weights of majority class instances are increased for samples causing overlap (positive margin) according to Equation 5. This adaptive changing of

weights ensures that instances causing confusion due to class overlap are progressively eliminated, refining the class separation over multiple iterations k. Algorithm 1 explains BiOR in detail.

$$w_{\text{majority}}^{m+1} = w_{\text{majority}}^{m} \times e^{-M_m(y=1)} \quad (5)$$

The time complexity of computing the margins for all instances, for calculating the percentiles and for removal and updation of weights is at most O(N) which is very small in comparison to the cost for training the Random Forest i.e., O (d*NlogN), where d is the dimensionality and N is the number of instances. As these steps are repeated for k iterations, the overall complexity of BiOR is approximately given as O(k*d*NlogN).

4. Algorithm 1: Boosting Inspired Overlap Reduction (BiOR)
Input:
$X_{minority}$: Minority class samples
$X_{majority}$:Majority class samples
y: Binary class labels, where $y_i \in \{0,1\}$ and 0 denotes the minority class, 1 denotes the majority class.
k: Number of boosting iterations
ϵ: Overlap threshold
α: Scaling factor for overlap reduction threshold
Base classifier C (e.g., Random Forest)
Output:
$X^*_{majority}$: Adjusted majority class samples with reduced overlap

Procedure:	
1.	Initialization
	1.1 Let $X_{minority}=\{x_i \mid y_i=0\}$ and $X_{majority}=\{x_i \mid y_i=1\}$.
	1.2 Initialize weights for each instance: $w^0_{minority} = \frac{1}{\lvert X_{minority}\rvert}$ $w^0_{majority} \frac{1}{\lvert X_{majority}\rvert}$
	1.3 Initialize the overlap set $O = \emptyset$
2.	Compute initial Overlap Degree (OD) :
	2.1 Train Random Forest, the base classifier, f on the combined dataset: $X = X_{minority} \cup X_{majority}$, $y = \{0\}^{\lvert X_{minority}\rvert} \cup \{1\}^{\lvert X_{majority}\rvert}$
	2.2 Compute the modified margin for each sample: $M_i = \frac{v_1 - v_0}{v_1 + v_0}$, where v_1, v_0 are votes for class 0 and 1 respectively.

	2.3 Let: $M_{minority} = \{margin(x) \mid x \in X_{minority}\}$, $M_{majority} = \{ margin(x) \mid x \in X_{majority} \}$
	2.4 Compute the third quartile (Q3) of minority margins ($M_{minority}$) and first quartile (Q1) of majority margins ($M_{majority}$):
	$Q3_{minority} = Percentile (M_{minority,} 75)$
	$Q1_{majority} = Percentile (M_{majority},25)$
	2.5 Compute the Initial Overlap degree (OD_0):
	$OD_0 = Q3_{minority} - Q1_{majority}$
3.	Iterative Overlap Reduction using Boosting-inspired framework
	For m=1 to k do:
	3.1 Train a new classifier f_m using sample-weighted training:
	$f_m \leftarrow Train (X, y, w_m)$
	3.2 Compute modified margins M_m for all samples.
	3.3 Compute variance of majority and minority samples:
	$\sigma^2_{minority} = Var(M_m \mid y=0)$, $\sigma^2_{majority} = Var(M_m \mid y=1)$
	3.4 Determine the adaptive overlap threshold:
	$\epsilon_m = \alpha \times (\sigma^2_{minority} + \sigma^2_{majority})$
	3.5 Recompute overlap degree OD_m :
	$Q3^m_{minority} = Percentile (M_{minority,} 75)$
	$Q1^m_{majority} = Percentile (M_{majority},25)$
	$OD_m = Q3^m_{minority} - Q1^m_{majority}$
	3.6 Early Stopping Condition: If $OD_m < \epsilon_m$, terminate the iteration and proceed to step 4.
	3.7 Identify overlapping instances:
	$O_m = \{ x_i \in X_{majority} \mid M_m(x_i) \leq 0\} \cup \{ x_j \in X_{minority} \mid M_m(x_j) \geq 0\}$
	3.7 Update the overlap set:
	$O \leftarrow O \cup O_m$
	3.8 Update Weights for Next Iteration
	$w^{m+1}_{minority} = w^m_{minority} \times e^{-M_m(y=0)}$
	$w^{m+1}_{majority} = w^m_{majority} \times e^{-M_m(y=1)}$
	3.9 Normalize weights:
	$w^{m+1}_{minority} = \frac{w^{m+1}_{minority}}{\sum w^{m+1}_{minority}}$, $w^{m+1}_{majority} = \frac{w^{m+1}_{majority}}{\sum w^{m+1}_{majority}}$
4.	Construct the final adjusted dataset:
	Remove the identified overlapping majority samples
	$X^*_{majority} = X_{majority} \setminus O$
5.	Return the modified datasets $X_{minority}$, $X^*_{majority}$

5. Materials and Methods

This section discusses the datasets and the benchmark sampling techniques used in this study.

5.1 *Datasets*

16 binary synthetic datasets consisting of 1000 observations each are generated from bivariate normal distributions. To investigate the classification performance under different overlap conditions, the minority class's mean vector is fixed at [0.00, 0.00]. The majority class's mean vector is varied at [0.05, 0.05], [0.20, 0.20], [0.30, 0.30], and [0.50, 0.50] to represent four distinct overlap scenarios. Specifically, a mean vector of [0.05, 0.05] signifies close class centres resulting in significant overlap, while [0.30, 0.30] shows widely separated class centres with minimal overlap. For each category of overlapping degree, the imbalance ratio was varied as 1:2, 1:4, 1:9 and 1:19 by varying the sample sizes of the two classes. The description of the synthetic datasets is given in Table 1.

Table 2 summarizes the 20 real-world datasets used in the experiment. These datasets are retrieved from UCI Machine Learning Repository. The datasets vary in several aspects, including imbalance ratios (*1.30-17.54*), dimensionality(*3-168*) and the total number of instances *(173-4839)*.

Table 1

Description of Synthetic Datasets used in this study

Datasets	# Positive instances	#Negative instances	Imbalance ratio	Two centres (minority, majority)	Overlapping Degree
P1	333	667	1:2	(0.00, 0.05)	Significant Overlap
P2	200	800	1:4	(0.00, 0.05)	
P3	100	900	1:9	(0.00, 0.05)	
P4	50	950	1:19	(0.00, 0.05)	
Q1	333	667	1:2	(0.00, 0.20)	Moderate Overlap
Q2	200	800	1:4	(0.00, 0.20)	
Q3	100	900	1:9	(0.00, 0.20)	
Q4	50	950	1:19	(0.00, 0.20)	
R1	333	667	1:2	(0.00, 0.30)	Low Overlap
R2	200	800	1:4	(0.00, 0.30)	
R3	100	900	1:9	(0.00, 0.30)	
R4	50	950	1:19	(0.00, 0.30)	
S1	333	667	1:2	(0.00, 0.50)	Rare Overlap
S2	200	800	1:4	(0.00, 0.50)	
S3	100	900	1:9	(0.00, 0.50)	
S4	50	950	1:19	(0.00, 0.50)	

Table 2

Description of datasets used in this study. IR represents the imbalance ratio

S.No	Dataset	#Instances	#Features	IR	OD	Overlap
D1	QSAR	1055	41	1.96	0.159	Low
D2	Breast Cancer	569	33	1.68	0.105	Low
D3	Haberman	306	3	2.78	-0.198	Low
D4	glass0	214	9	2.06	0.830	Moderate
D5	vehicle3	846	19	2.99	0.01	Rare
D6	German	1000	20	2.33	-0.129	Low
D7	ILPD	583	10	2.49	-0.220	Low
D8	Blood Transfusion	748	4	3.2	1.780	Significant
D9	musk1	476	168	1.3	0.325	Low
D10	Tic-Tac-Toe	958	9	1.89	0.069	Rare
D11	ionosphere	351	34	1.78	-0.035	Rare
D12	seismic-bumps	2584	19	14.2	-1.280	Significant
D13	ecoli4	335	7	15.75	-0.360	Low
D14	Wilt	4839	5	17.54	-0.164	Low
D15	glass5	214	10	22.77	-1.33	Significant
D16	glass4	214	10	15.46	-1.395	Significant
D17	glass6	185	29	6.37	0.03	Rare
D18	cleveland-0_vs_4	173	14	12.31	0.919	Moderate
D19	yeast3	1484	9	8.1	1.08	Significant
D20	yeast-2_vs_4	514	9	9.07	0.740	Moderate

5.2 *Benchmark Algorithms*

a. *Random Undersampling(RUS) :* It is one of the most straightforward and widely adopted strategies for addressing imbalanced datasets where samples from the majority class are randomly discarded until the classes are balanced. However, this approach is susceptible to loss of potentially valuable data[2][10][43].

b. *Cluster-Based Undersampling (CBUS):* A cluster-based under-sampling method was proposed to enhance the classification accuracy of the positive class [43][44]. They segmented the training data into clusters and selected representative samples from each cluster for the negative class, ensuring a balanced proportion relative to the positive class.

c. *Tomek Link(TL) :* A Tomek link is defined as a pair of nearest neighbors (x, y), where one of them belongs to the majority class and the other to the minority class, such that no other instance z exists where the distance between x and z or y and z is smaller than the distance between x and y. If two samples form a TL, they are situated at the class boundary, indicating potential noise or overlapping instances. The objective of TL undersampling technique is to eliminate majority class instances in each TL pair, thereby reducing overlap, and improving class separability [45][46].

d. *Condensed Nearest Neighbor (CNN):* It is an undersampling technique that selectively removes majority class instances near the decision boundary, as they are considered less critical for learning, while preserving instances farther from the decision boundary, as they are deemed to provide valuable information for training the classifier [47].

e. *One Sided Selection(OSS) :* It is an undersampling technique that combines TL and CNN rule to reduce the size of the majority class while preserving essential classification patterns. It first applies CNN rule to select a subset A from the dataset T ensuring that A correctly classifies T using 1-Nearest Neighbor (1-NN). The process begins with a set S, which includes all minority class instances and one representative from each majority class. Misclassified majority instances are iteratively added to S until no further misclassifications occur. After CNN filtering, OSS uses TL to remove noisy and Borderline majority class instances from the dataset to enhance class separability [48-50].

f. *Edited Nearest Neighbor (ENN):* It is an undersampling technique that works by analysing the neighborhood of the given majority class instance x. If this instance is misclassified by its k-nearest neighbors, it is removed from the original dataset [46].

g. *Neighborhood Cleaning Rule (NC) :* It follows a similar approach to OSS, focussing on data cleaning rather than data reduction. While it preserves the class of interest (C), the remaining data (O) is filtered to remove noise. It applies ENN rule to remove noisy instances (A_1) in O, and adds majority samples misclassifying C to A_2 , provided their class size is greater than or equal to 50% of C. The final dataset is obtained by removing A_1 and A_2 from the original dataset T[51].

h. *Near Miss (NM-1, NM-2 and NM-3) :* The Near Miss algorithm has three variations for under sampling the majority class by selecting samples based on their proximity to minority class instances. NM-1 selects majority class samples that are closest to minority class samples, specifically those with the smallest average distance to their three nearest minority class samples. NM-2 retains majority class samples that have the smallest average distance to the three farthest minority samples. NM-3 removes a specified

number of the closest majority class samples for each minority class sample [43][52].

i. *SMOTE:* As an oversampling technique, SMOTE generates synthetic instances of the minority class by interpolating between several existing k-nearest minority class instances instead of merely replicating them. However, a limitation of SMOTE is its potential to create overlap between classes, particularly when there is no clear boundary between them in the input space [2].

j. *Borderline SMOTE (BDS):* It is an extension of SMOTE that focuses on improving the classification of samples near the decision boundary as these are more likely to be misclassified in comparison to those farther away. Instead of using all the minority class instances, it identifies the k-nearest neighbors for a minority class instance x and applies SMOTE on those at higher risk of misclassification[46][50].

k. *RFCL:* It is an undersampling technique aimed designed to address class imbalance and overlap by leveraging random forest ensemble classifiers. It introduced an overlapping degree metric to quantify overlap. This technique removes majority class instances that cross a classification boundary, determined by maximizing the F1-score, to reduce overlap while preserving important boundary information. It leverages random forest margins for accurate instance selection and demonstrates superior performance compared to other under-sampling techniques [28].

6. Experiments

This section outlines the experimental settings and the metrics employed for performance evaluation. This research encompasses two distinct experiments. The first experiment assesses the performance of the proposed method BiOR on 16 synthetic datasets. Subsequently, the second experiment validates its effectiveness across 20 real-world datasets. For robust evaluation of classifiers in class imbalance scenarios, the AUC and F1 score are employed as evaluation metrics.

Experimental Configurations

All the experiments have been conducted on a system featuring an Apple M2 chip and 16 GB RAM, with implementations in Python 3.7. Standard scores (z-scores) are applied for normalization to maintain consistent scaling across all experiments. Each dataset is split into training and testing subsets in a 70:30 ratio. For reliable model selection and evaluation, 5-fold cross-validation is employed during the training phase[10]. The performance of BiOR is evaluated and compared against 12 benchmark and 1 latest (RFCL) sampling techniques as detailed in Section 4.2.

1) Undersampling techniques: RUS, CBUS, TL, OSS, ENN, NC, NM-1, NM-2, NM-3, CNN and RFCL.
2) Oversampling Techniques: SMOTE and BSMOTE.

These techniques have been selected for comparison as they represent a diverse range of strategies to address class imbalance, ensuring comprehensive evaluation [47]. Random Forest is selected as the learning algorithm as it is the standard machine learning model which focuses on maximizing overall classification accuracy and is widely used for handling imbalanced classification problems [17]. Moreover it is also robust against overfitting which makes it suitable for datasets with different sample sizes [28].

Evaluation Metrics

Metrics such as Area Under the Receiver Operating Characteristic Curve (AUC) and F-Measure (F1-score) have been utilized in this study as they are widely adopted and appropriate measures that can effectively assess the model's ability to distinguish between imbalanced classes [15]. The aim of using two performance measures is to gain a more reliable understanding of the learning algorithms. AUC provides a single scalar value to summarize the model's performance across all thresholds. AUC value closer to 1 denotes enhanced model effectiveness, reflecting its stronger discriminative power between different classes. Furthermore, the F1 score, which is the harmonic mean of sensitivity and precision, is calculated according to Equation 6 [10, 27].

$$\text{F-Measure} = (2 \times \text{sensitivity} \times \text{precision}) / (\text{sensitivity} + \text{precision}) \quad (6)$$

Statistical Tests

Given that most datasets in this study are ordinal, a non-parametric Friedman test, as recommended in the literature [51, 52, 53, 54], is employed for statistical validation. This test compared the performance rankings of BiOR against 13 existing sampling techniques. The null hypothesis, stating no difference in the average ranks of classifiers across the data sets, is tested against the alternate hypothesis which states that such a difference does exist A p-value below 0.05 is set as the threshold for rejecting the null hypothesis, thereby indicating a statistically significant difference in classifier rankings.

7. Results and Discussion

The focus of this study is to evaluate the effectiveness of BiOR algorithm in handling classification challenges stemming from class overlap in imbalanced datasets. To accomplish this, two sets of experiments have been conducted. Initially, BiOR's performance is assessed on 16 synthetic datasets, designed to simulate varying degrees of overlap and imbalance ratios. Subsequently, the

algorithm's validation is extended to 20 real-world datasets, ensuring that BiOR's effectiveness is demonstrated in practical classification scenarios.

Tables 3 and 4 present the experimental results of BiOR alongside 13 existing sampling techniques, evaluated in terms of AUC and F1 score respectively. The best values for each dataset have been highlighted. In datasets with such as *Blood Transfusion, yeast3, seismic-bumps, glass4 and glass5* the IR increases progressively from left to right. These datasets exhibit significant overlap as indicated in Table 2. While under such conditions BiOR achieves competitive results in terms of AUC values, the extreme class overlap limits its ability to optimize the F1 score compared to datasets with moderate or low overlap. This is because the high degree of overlap hinders accurate distinction between classes as F1 score is sensitive to both precision and recall leading to a decline in its performance.

In moderately overlapped datasets including *glass0, yeast-2_vs_4* and *cleveland0_vs_4,* BiOR outperforms the benchmark algorithms in terms of both AUC and F1 score. Similarly, in datasets with low overlap as categorized in Table 2, it consistently performs better than other sampling techniques. In datasets with rare overlap such as *ionosphere, tic-tac-toe, vehicle3* and *glass,* BiOR maintains robust performance irrespective of the imbalance ratio.

In synthetic datasets with high overlap (e.g. P1, P2, P3 and P4) even as the imbalance ratio increases from P1 to P4, the results indicate that BiOR consistently delivers competitive AUC and F1 scores irrespective of the imbalance ratio. In datasets with moderate overlap (Q1 to Q4), while BiOR's AUC values demonstrate the strong performance, however in terms of F1 scores RFCL lowly outperforms BiOR as the imbalance increases. However it still surpasses many existing techniques like RUS, CBUS, TL, ENN,NM-1, NM-2 and BSMOTE. Furthermore, in datasets with low to rare overlap, BiOR's performance closely aligns with that of other methods underscoring its overall robustness. Figure 3&4 depicts the graphical representation of AUC values of synthetic datasets with different degrees of overlap.

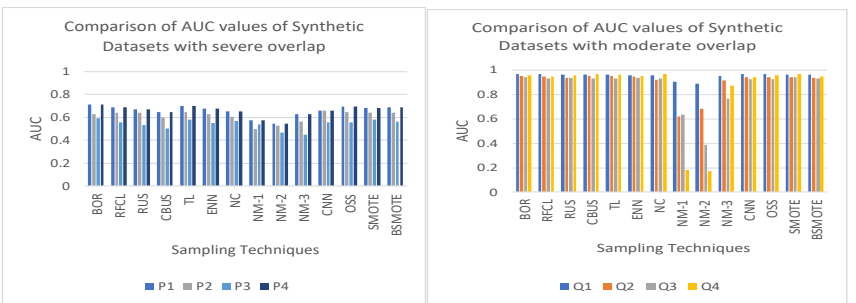

Figure 3
Graphical representation of comparison of AUC values of synthetic datasets with significant and moderate overlap respectively.

The results demonstrate that BiOR consistently achieved remarkable performance across most real-world datasets, substantially enhancing the classification performance by minimizing the degree of overlap between the classes, thereby improving class separability and achieving better AUC and F1 scores.

In order to assess the statistical significance among the comparative methods, Friedman test is applied to AUC and F1 metrics. Since our experiments involve 14 methods, the Friedman statistic follows the F-distribution with 13 degrees of freedom. The p-value is observed to be less than 0.05 (refer Table 6), which means the null hypothesis is rejected. Therefore, there is a statistically significant difference in the average ranks between AUC and F1 scores. Table 5 presents the Friedman's ranking of the two evaluation metrics. In general, the algorithm with a higher average rank performed better than the other algorithms with a lower average rank. Here, BiOR achieved the highest average rank of 10.89 followed by OSS (9.16) and SMOTE (9.1).

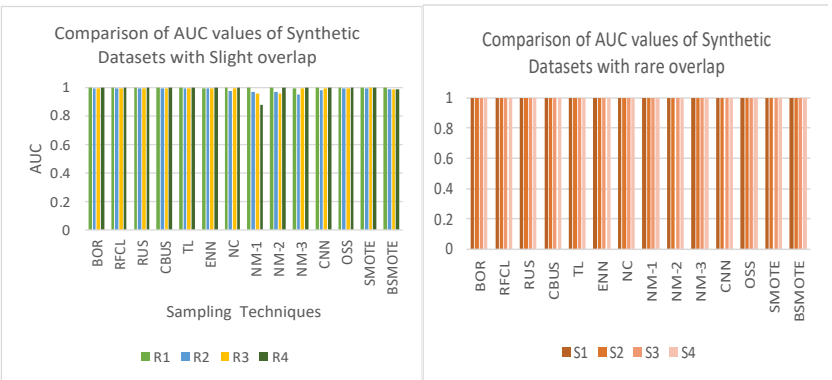

Figure 4

Graphical representation of comparison of AUC values of synthetic datasets with low and rare overlap respectively.

Summary

The experimental findings demonstrated that BiOR is a promising approach for addressing classification challenges in imbalanced datasets with overlapping instances. It consistently outperformed traditional sampling techniques across both synthetic and real-world datasets, achieving higher AUC and F1 scores. The statistical analysis using the Friedman test confirmed the significant difference in performance rankings among the evaluated methods. Its ability to enhance class separation while preserving essential data characteristics makes it a valuable addition to existing imbalance-handling methodologies.

Table 3
Results of AUC values

Data-sets	BiOR	RFCL	RUS	CBUS	TL	ENN	NC	NM-1	NM-2	NM-3	CNN	OSS	SMOTE	BDS
D1	0.9199	0.9309	0.9323	0.9357	0.9296	0.9327	0.92	0.8688	0.8705	0.883	0.9104	0.9225	0.9322	**0.9406**
D2	**0.9997**	0.9951	0.998	0.998	0.9957	0.9977	0.9987	0.9938	0.9892	0.9836	0.9977	0.9957	0.9984	0.9993
D3	**0.731**	0.6861	0.697	0.6562	0.6875	0.7242	0.7188	0.6529	0.5639	0.534	0.6562	0.6739	**0.731**	0.6766
D4	**0.9602**	0.9403	0.9403	0.9403	0.9375	0.9517	0.929	0.8608	0.8977	0.8409	0.9233	0.9347	0.9261	0.9119
D5	**0.8912**	0.8333	0.8757	0.8583	0.859	0.8838	0.8511	0.7502	0.692	0.7733	0.8212	0.8638	0.8767	0.8798
D6	0.6713	0.6782	0.6869	0.6656	0.6849	0.6867	0.6852	**0.7243**	0.6272	0.6638	0.6629	0.6777	0.6853	0.6616
D7	**0.7813**	0.7431	0.7491	0.7427	0.7322	0.7738	0.7423	0.5154	0.6965	0.6613	0.7479	0.7543	0.7438	0.7404
D8	**0.7114**	0.6907	0.7071	0.6734	0.6879	0.6912	0.6752	0.5511	0.4848	0.5123	0.6943	0.6842	0.6785	0.6696
D9	**1**	0.9991	0.9965	**1**	0.9948	0.9844	0.9861	0.9931	0.9901	0.9987	0.9995	0.9943	0.9995	1
D10	0.9231	0.9402	0.9173	0.8651	0.9325	0.909	0.8386	**0.9608**	0.8923	0.9213	0.9733	0.9204	0.9023	0.8995
D11	**0.9599**	0.9452	0.9301	0.9361	0.9545	0.9537	0.9377	0.8998	0.9023	0.9234	0.8627	0.9553	0.9419	0.9444
D12	0.7516	0.7603	0.777	0.6022	0.7614	**0.7761**	0.7507	0.7128	0.279	0.3303	0.7191	0.7669	0.7413	0.7488
D13	**1**	0.9973	0.9758	**1**	0.9919	0.9973	**1**	0.8925	0.8898	0.9919	**1**	**1**	**1**	**1**
D14	**0.9826**	0.9801	0.9817	0.9713	0.9754	0.981	0.9807	0.9122	0.8666	0.9718	0.9738	0.9816	0.9826	0.9822
D15	**0.9964**	0.9821	0.9679	0.9946	0.9929	0.9929	0.9929	0.9464	0.9375	0.9821	0.9536	0.9893	0.9929	0.9893
D16	0.9936	0.9744	0.8622	0.9038	**1**	0.9808	0.9936	0.9615	0.9615	0.984	**1**	0.9744	**1**	**1**
D17	**1**	0.9667	0.9291	0.9916	0.9667	**1**	**1**	0.8708	**1**	0.9667	0.8916	0.9833	**1**	**1**
D18	**1**	**1**	**1**	0.9705	**1**	**1**	**1**	0.9117	0.9705	**1**	**1**	**1**	**1**	0.9705
D19	**0.9773**	0.9767	0.9728	0.9803	0.9753	0.9779	0.9781	0.9225	0.9051	0.9676	0.9808	0.9814	0.9812	0.9732
D20	**0.9995**	0.9941	0.9802	0.9991	0.9941	0.9941	0.9961	0.9961	0.9901	0.9881	0.9911	0.9961	0.9991	0.9901

Table 4

Results of F1 Scores

Datasets	BiOR	RFCL	RUS	CBUS	TL	ENN	NC	NM-1	NM-2	NM-3	CNN	OSS	SMOTE	BDS
D1	0.8881	0.8993	0.9	0.8815	0.8973	0.8905	0.8699	0.8612	0.85	0.8662	0.8718	0.8963	**0.9085**	0.9058
D2	0.9722	0.9722	0.9787	0.8815	0.9650	0.9722	**0.9859**	0.9718	0.9722	0.965	0.964	0.9722	0.9718	0.9859
D3	**0.8485**	0.8043	0.6914	0.9718	0.8163	0.7073	0.7561	0.7	0.5429	0.5753	0.7229	0.8283	0.7229	0.7381
D4	**0.7692**	0.7619	**0.7692**	0.5075	0.7500	0.7143	0.7143	0.7407	**0.7692**	0.7143	0.7143	0.8	0.7692	0.7521
D5	0.3673	0.3556	0.5686	0.6452	0.5079	0.5870	0.551	0.4419	0.3876	0.4348	0.4752	0.4375	0.5952	0.5745
D6	**0.8235**	0.8127	0.7063	0.5376	0.7852	0.7287	0.7244	0.7287	0.6475	0.7209	0.6748	0.817	0.7640	0.7424
D7	**0.8316**	0.7771	0.731	0.6498	0.7953	0.6866	0.6866	0.5672	0.6232	0.7143	0.7034	0.8208	0.7600	0.7582
D8	0.36	0.4444	0.4286	**0.6715**	0.4667	0.4938	0.4318	0.4228	0.4189	0.3738	0.4706	0.4746	0.4595	0.4773
D9	**0.9894**	0.9671	0.9677	0.4821	0.9677	0.9375	0.9375	0.9591	0.9401	0.9215	0.9494	0.9473	0.9787	0.9894
D10	**0.9286**	0.8921	0.878	0.9791	0.8945	0.8803	0.808	0.8955	0.8368	0.873	0.7685	0.8746	0.8812	0.8769
D11	**0.9233**	0.9132	0.9111	0.7593	0.9232	0.9232	0.9232	0.8101	0.8292	0.9111	0.6944	0.8913	0.9011	0.9231
D12	**0.9762**	0.9696	0.822	0.9011	0.9660	0.9658	0.8594	0.5242	0.1741	0.3785	0.9597	0.966	0.8919	0.8831
D13	**0.992**	0.9841	0.9412	0.4098	0.9841	0.9841	0.992	0.4691	0.9231	0.9836	0.9836	0.9841	0.9920	0.9841
D14	**0.9987**	0.9887	0.9622	0.9667	0.9881	0.9881	0.9564	0.7924	0.5209	0.9704	0.9863	0.9892	0.9715	0.9715
D15	**0.9722**	0.9577	0.9118	0.9111	0.9722	0.9714	0.9714	0.8571	0.8438	0.9714	0.9254	**0.9722**	0.9577	0.9577
D16	0.975	0.975	0.8857	0.9714	0.9750	0.9744	0.987	0.8529	0.9459	0.9459	**1**	0.9744	1.0000	1.0000
D17	0	0	0.4615	0.7419	0.000	0.0000	**0.8**	0.3333	0.4	0.4615	0.3636	0	0.0000	0.0000
D18	**1**	**1**	0.6666	0.4285	1.0000	1.0000	1	0.25	0.4	**1**	**1**	0	1.0000	0.0000
D19	0.7272	0.7076	0.7708	0.3333	0.7272	0.7886	0.8045	0.5245	0.3238	0.7714	**0.8219**	0.7462	0.8048	0.7954
D20	**0.9091**	0.8695	0.7096	0.6923	0.8695	0.8801	0.8148	0.5522	0.4682	0.8333	0.8803	0.8181	0.7587	0.7586

Table 5

Results of Friedman Test on AUC values and F1-Scores

Metrics	AUC	F1Score	Average
BiOR	11.06	10.72	10.89
RFCL	8.4	9.78	9.09
RUS	7.94	6.19	7.065
Cluster-Based US	7.86	4.99	6.425
TL	9.01	9.71	9.36
ENN	8.71	7.71	8.21
NC	7.53	7.06	7.295
NM-1	3.69	4.58	4.135
NM-2	3.21	4.69	3.95
NM-3	4.33	6.81	5.57
CNN	7.07	7.08	7.075
OSS	8.97	9.35	9.16
SMOTE	9.69	8.51	9.1
BSMOTE	7.51	7.82	7.665

Table 6

Friedman Test Statistics

	AUC	F1-Score
N	36	36
Chi-Square	160.404	115.605
Degree of freedom	13	13
p-value	<0.001	<0.001

8. Conclusion and Future Work

This study presented the Boosting-Inspired Overlap Reduction (BiOR) algorithm, a novel approach designed to address the dual challenges of class imbalance and class overlap in classification problems. Unlike conventional resampling techniques, BiOR employed an adaptive, margin-based overlap reduction strategy integrated within a boosting-inspired framework which iteratively refined the decision boundaries and selectively removed the overlapping majority class instances to improve classification performance. By systematically reducing overlap and dynamically adjusting sample weights, BiOR offered a principled approach to improve classifier performance in highly complex imbalanced settings.

Extensive experimentation on 16 synthetic and 20 real-world datasets demonstrated BiOR's superiority over 13 state-of-the-art sampling techniques. The results indicated that BiOR consistently achieved higher AUC and F1 scores, particularly in datasets with high overlap, where traditional methods failed to mitigate decision boundary ambiguity. The statistical analysis using

Friedman test further validated that the observed improvements are statistically significant.

Future research directions may include extending BiOR to multi-class problems, optimizing computational efficiency for large-scale datasets, and integrating deep learning-based classifiers to further enhance its applicability to real-world datasets.

References

[1] H. Guo, Y. Li, J. Shang, M. Gu, Y. Huang, and B. Gong, "Learning from class-imbalanced data: Review of methods and applications," *Expert Syst. Appl.*, vol. 73, pp. 220–239 (2017).

[2] V. García, R. Alejo, J. S. Sánchez, J. M. Sotoca, and R. A. Mollineda, "Combined effects of class imbalance and class overlap on instance-based classification," in Proc. 7th Int. Conf. Intell. Data Eng. Autom. Learn. (IDEAL), Burgos, Spain, pp. 371–378 (2006).

[3] A. A. Khan, O. Chaudhari, and R. Chandra, "A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation," E*xpert Syst. Appl.*, vol. 244, p. 122778 (2024).

[4] Z. Xu, D. Shen, T. Nie, and Y. Kou, "A hybrid sampling algorithm combining M-SMOTE and ENN based on random forest for medical imbalanced data," *J. Biomed. Inform.*, vol. 107, p. 103465 (2020).

[5] P. Tyagi, J. Singh, and A. Gosain, "Handling class imbalance problem using feature selection techniques: A review," in Proc. Int. Conf. Innovations Comput. Intell. Comput. Vis., Singapore (2022).

[6] K. Matloob, M. A. Khan, S. Abbas, M. T. Khan, and S. Khan, "A comparative performance analysis of data resampling methods on imbalance medical data," *IEEE Access,* vol. 9, pp. 109960–109975 (2021).

[7] A. More, "Survey of resampling techniques for improving classification performance in unbalanced datasets," arXiv preprint, arXiv:1608.06048 (2016).

[8] R. Ghorbani and R. Ghousi, "Comparing different resampling methods in predicting students' performance using machine learning techniques," *IEEE Access*, vol. 8, pp. 67899–67911 (2020).

[9] F. Feng, Y. Xu, Q. Li, Z. Huang, and X. Zheng, "Using cost-sensitive learning and feature selection algorithms to improve the performance

of imbalanced classification," *IEEE Access*, vol. 8, pp. 69979–69996 (2020).

[10] B. Pes, "Learning from high-dimensional biomedical datasets: The issue of class imbalance," *IEEE Access,* vol. 8, pp. 13527–13540 (2020).

[11] M. Bach and A. Werner, "Cost-sensitive feature selection for class imbalance problem," in *Information Systems Architecture and Technology*, Springer (2018).

[12] K. M. Hasib, M. M. Rahman, M. S. Hossain, and A. K. M. Masum, "A survey of methods for managing the classification and solution of data imbalance problem," arXiv preprint, arXiv:2012.11870 (2020).

[13] M. S. Santos, P. H. Abreu, N. Japkowicz, A. Fernández, C. Soares, S. Wilk, and J. Santos, "On the joint-effect of class imbalance and overlap: A critical review," *Artif. Intell. Rev.,* vol. 55, no. 8, pp. 6207–6275 (2022).

[14] M. S. Santos, P. H. Abreu, N. Japkowicz, A. Fernández, and J. Santos, "A unifying view of class overlap and imbalance: Key concepts, multi-view panorama, and open avenues for research," *Inf. Fusion,* vol. 89, pp. 228–253 (2023).

[15] A. Kumar, D. Singh, and R. S. Yadav, "Class overlap handling methods in imbalanced domain: A comprehensive survey," *Multimedia Tools Appl.,* vol. 83, no. 23, pp. 63243–63290 (2024).

[16] S. Wojciechowski and S. Wilk, "Difficulty factors and preprocessing in imbalanced data sets: An experimental study on artificial data," *Found. Comput. Decis. Sci*., vol. 42, no. 2, pp. 149–176 (2017).

[17] P. Vuttipittayamongkol, E. Elyan, and A. Petrovski, "On the class overlap problem in imbalanced data classification," *Knowl.-Based Syst*., vol. 212, p. 106631 (2021).

[18] G. H. Fu, Y. J. Wu, M. J. Zong, and L. Z. Yi, "Feature selection and classification by minimizing overlap degree for class-imbalanced data in metabolomics," *Chemom. Intell. Lab. Syst.,* vol. 196, p. 103906 (2020).

[19] A. Kumar, D. Singh, and R. S. Yadav, "Entropy and improved k-nearest neighbor search-based undersampling method to handle class overlap in imbalanced datasets," *Concurrency Comput.: Pract. Exp.,* vol. 36, no. 2, p. e7894 (2024).

[20] L. Gong, H. Zhang, J. Zhang, M. Wei, and Z. Huang, "A comprehensive investigation of the impact of class overlap on software defect

prediction," *IEEE Trans. Softw. Eng.*, vol. 49, no. 4, pp. 2440–2458 (2022).

[21] C. Liu, Y. Jin, H. Chen, and X. Zhang, "Constrained oversampling: An oversampling approach to reduce noise generation in imbalanced datasets with class overlapping," *IEEE Access,* vol. 10, pp. 91452–91465 (2022).

[22] H. K. Lee and S. B. Kim, "An overlap-sensitive margin classifier for imbalanced and overlapping data," *Expert Syst. Appl.,* vol. 98, pp. 72–83 (2018).

[23] D. Devi, S. K. Biswas, and B. Purkayastha, "Learning in presence of class imbalance and class overlapping using one-class SVM and undersampling technique," *Connection Sci.,* vol. 31, no. 2, pp. 105–142 (2019).

[24] F. Grina, Z. Elouedi, and E. Lefevre, "Evidential undersampling approach for imbalanced datasets with class-overlapping and noise," in Proc. 18th Int. Conf. Modeling Decisions Artif. Intell. (MDAI), pp. 181–192 (2021).

[25] E. Alogogianni and M. Virvou, "Handling class imbalance and class overlap in machine learning applications for undeclared work prediction," *Electronics,* vol. 12, no. 4, p. 913 (2023).

[26] P. Vuttipittayamongkol, E. Elyan, A. Petrovski, and C. Jayne, "Overlap-based undersampling for improving imbalanced data classification," in Proc. IDEAL, Madrid, Spain, pp. 689–697 (2018).

[27] P. Vuttipittayamongkol and E. Elyan, "Neighbourhood-based undersampling approach for handling imbalanced and overlapped data," *Inf. Sci.,* vol. 509, pp. 47–70 (2020).

[28] R. Zhang, Z. Zhang, and D. Wang, "RFCL: A new under-sampling method of reducing the degree of imbalance and overlap," *Pattern Anal. Appl.,* vol. 24, pp. 641–654 (2021).

[29] W. Wang and D. Sun, "The improved AdaBoost algorithms for imbalanced data classification," *Inf. Sci.,* vol. 563, pp. 358–374 (2021).

[30] R. Patil and K. Vanjerkhede, "Random forest machine learning approach for efficient optimization of Sierpinski carpet fractal antenna," *J. Inf. Optim. Sci.,* vol. 45, no. 4, pp. 851–861 (2024).

[31] A. Upadhyay, M. Singh, and V. K. Yadav, "Improvised number identification using SVM and random forest classifiers," *J. Inf. Optim. Sci.,* vol. 41, no. 2, pp. 387–394 (2020).

[32] V. García, R. A. Mollineda, and J. S. Sánchez, "On the k-NN performance in a challenging scenario of imbalance and overlapping," *Pattern Anal. Appl.*, vol. 11, pp. 269–280 (2008).

[33] Z. Borsos, C. Lemnaru, and R. Potolea, "Dealing with overlap and imbalance: A new metric and approach," *Pattern Anal. Appl.*, vol. 21, pp. 381–395 (2018).

[34] J. A. Sáez, M. Galar, and B. Krawczyk, "Addressing the overlapping data problem in classification using the one-vs-one decomposition strategy," *IEEE Access*, vol. 7, pp. 83396–83411 (2019).

[35] A. Guzmán-Ponce, R. M. Valdovinos, J. S. Sánchez, and J. R. Marcial-Romero, "A new under-sampling method to face class overlap and imbalance," *Appl. Sci*., vol. 10, no. 15, p. 5164 (2020).

[36] P. Vuttipittayamongkol and E. Elyan, "Improved overlap-based undersampling for imbalanced dataset classification with application to epilepsy and Parkinson's disease," *Int. J. Neural Syst.*, vol. 30, no. 8, p. 2050043 (2020).

[37] M. M. Nwe and K. T. Lynn, "KNN-based overlapping samples filter approach for classification of imbalanced data," in *Software Engineering Research, Management and Applications*, pp. 55–73 (2020).

[38] B. W. Yuan, Z. L. Zhang, X. G. Luo, Y. Yu, X. H. Zou, and X. D. Zou, "OIS-RF: A novel overlap and imbalance-sensitive random forest," *Eng. Appl. Artif. Intell*., vol. 104, p. 104355 (2021).

[39] Y. Zhang and F. Han, "An improved ensemble classification algorithm for imbalanced data with sample overlap," in Proc. Int. Conf. Neural Comput. Adv. Appl., pp. 454–468 (2022).

[40] M. A. Islam, M. A. Uddin, S. Aryal, and G. Stea, "An ensemble learning approach for anomaly detection in credit card data with imbalanced and overlapped classes," *J. Inf. Secur. Appl.*, vol. 78, p. 103618 (2023).

[41] S. Jubair, J. Yang, and B. Ali, "Overlap to equilibrium: Oversampling imbalanced datasets using overlapping degree," *Inf. Process. Manage.*, vol. 62, no. 2, p. 103975 (2025).

[42] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Front. Neurorobot.*, vol. 7, p. 21 (2013).

[43] S. J. Yen and Y. S. Lee, "Cluster-based under-sampling approaches for imbalanced data distributions," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 5718–5727 (2009).

[44] M. A. U. H. Tahir, S. Asghar, A. Manzoor, and M. A. Noor, "A classification model for class imbalance dataset using genetic programming," *IEEE Access*, vol. 7, pp. 71013–71037 (2019).

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

[46] P. Tyagi, J. Singh, and A. Gosain, "Comprehensive empirical investigation for prioritizing the pipeline of using feature selection and data resampling techniques," *J. Intell. Fuzzy Syst.*, vol. 46, no. 3, pp. 6019–6040 (2024).

[47] X. Tao, Q. Li, H. Zhang, and J. Li, "Real-value negative selection oversampling for imbalanced dataset learning," *Expert Syst. Appl.*, vol. 129, pp. 118–134 (2019).

[48] M. Kubat and S. Matwin, "Addressing the curse of imbalanced training sets: One-sided selection," in Proc. Int. Conf. Mach. Learn. (ICML), Nashville, TN, USA, p. 179 (1997).

[49] S. Fotouhi, S. Asadi, and M. W. Kattan, "A comprehensive data-level analysis for cancer diagnosis on imbalanced data," *J. Biomed. Inform.*, vol. 90, p. 103089 (2019).

[50] V. H. Barella, L. P. Garcia, M. C. de Souto, A. C. Lorena, and A. C. de Carvalho, "Assessing the data complexity of imbalanced datasets," *Inf. Sci.*, vol. 553, pp. 83–109 (2021).

[51] J. Laurikkala, "Improving identification of difficult small classes by balancing class distribution," in Proc. 8th Conf. Artif. Intell. Med. Eur. (AIME), Cascais, Portugal, pp. 63–66 (2001).

[52] I. Mani and I. Zhang, "kNN approach to unbalanced data distributions: A case study involving information extraction," in Proc. Workshop Learn. Imbalanced Datasets, Washington, DC, USA, pp. 1–7 (2003).

[53] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30 (2006).

[54] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of non-parametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18 (2011).

Received March, 2025