MEXT: A Parameter-Free Oversampling Approach for Multi-Class Imbalanced Datasets

Chittima Chiamanusorn, Krung Sinapiromsaran
Department of Mathematics and Computer Science
Chulalongkorn University
Bangkok, Thailand, 10330

***Abstract*—Machine learning classifiers face significant challenges when confronted with class-imbalanced datasets, particularly in multi-class scenarios. The inherent skewness in class distributions often leads to biased model predictions, with classifiers struggling to accurately identify instances from underrepresented classes. This paper introduces MEXT, a novel parameter-free oversampling technique specifically designed for multi-class imbalanced datasets. Unlike conventional approaches that often rely on the one-against-all strategy and require manual parameter tuning for each class, MEXT addresses these limitations by simultaneously balancing all classes. By leveraging anomalous score analysis, MEXT automatically determines optimal locations for synthesizing new instances of minority classes, eliminating the need for manual parameter selection. The technique aims to achieve a balanced class distribution where each class has an equal number of instances. To evaluate MEXT's effectiveness, the experiments were conducted extensively on a collection of multi-class datasets from the UCI repository. The proposed MEXT algorithm was evaluated against a suite of state-of-the-art SMOTE-based oversampling techniques, including SMOTE, ADASYN, Safe-Level SMOTE, MDO, and DSRBF. All comparative algorithms were implemented within the one-against-all framework. Hyperparameter optimization for each algorithm was performed using grid search. An automated machine learning pipeline was employed to identify the optimal classifier-hyperparameter combination for each dataset and oversampling technique. The Wilcoxon signed-rank test was subsequently utilized to statistically assess the performance of MEXT relative to the other oversampling techniques. The results demonstrate that MEXT consistently outperforms the other methods in terms of average ranking of key evaluation metrics, including macro-precision, macro-recall, F_1-measure, and G-mean, indicating its superior ability to address multi-class imbalanced learning problems.**

Keywords*—*Class imbalance; classification; extreme anomalous; multiclass; oversampling; parameter-free

I. Introduction

Multiclass imbalanced learning poses a significant challenge in machine learning, particularly within real-world applications [1], [2], [3], [4], [5]. This challenge arises when training data exhibits a skewed class distribution, where one or more classes possess substantially fewer instances than others. This imbalance can detrimentally impact classifier performance due to two primary factors, which are data inadequacy and data ambiguity.

Data Inadequacy: When the number of instances from a minority class is insufficient, classifiers may struggle to recognize and accurately model the characteristics of that class, potentially leading to their misclassification as noise or outliers. For example, in a medical dataset with a rare disease, if the number of patients with the disease is very small, the classifier may not learn to accurately identify the disease, leading to misdiagnoses.

Data Ambiguity: When minority classes share significant characteristics with majority classes, classifiers may erroneously classify minority instances as belonging to the majority class. This ambiguity arises from the inherent limitations of traditional classification algorithms, which are often optimized for generalizability across the entire dataset rather than specifically addressing class imbalances. For instance, in a dataset of images containing different types of birds, a classifier might struggle to distinguish between rare bird species that share similar physical characteristics with more common species.

Over the past decades, imbalanced learning has garnered considerable attention within the machine learning research community, as evidenced by the increasing number of publications on this topic [6]. This has led to the development of various approaches to address this challenge, including algorithmic-level and data-level methods. Algorithmic-level methods aim to modify existing classification algorithms to better accommodate imbalanced data. However, their applicability is limited as they are often designed for specific classifiers. Conversely, data-level methods, which involve preprocessing the training data to address class imbalances, exhibit greater flexibility and can be applied to a wider range of classifiers.

One prominent data-level technique is the Synthetic Minority Over-sampling Technique (SMOTE) [7]. SMOTE addresses class imbalance by generating synthetic instances of minority class instances based on their feature similarity. This process involves creating new instances along the line segments connecting existing minority class instances within a defined region.

The success of SMOTE has spurred the development of numerous variations, each employing different strategies for identifying optimal synthesis regions. A comprehensive collection of 86 SMOTE variants is available in the open-source smote-variants package for Python [8]. While this package provides access to a wide array of oversampling techniques, many of these variants are specifically designed for binary classification problems and cannot directly handle multiclass imbalanced datasets.

To address multiclass scenarios, the one-against-all (OAA) approach is commonly employed by decomposing the problem into a series of binary classification tasks. In each task, a single class is designated as the positive class, while all other classes are aggregated into a single negative class. Despite its conceptual simplicity, the OAA approach can exhibit inherent limitations. Notably, it can be susceptible to class imbalance issues, particularly when the number of classes increases. This imbalance, arising from a significant disparity between the number of instances in the positive and negative classes within each binary classification task, can bias the learned models towards the majority class, potentially compromising the accurate identification of instances from the minority class. Furthermore, the independent training of each binary classifier can result in inconsistent decision boundaries across different classification tasks. These inconsistencies can lead to ambiguous classifications for certain instances, where the predicted class may vary depending on the specific binary classifier employed. Consequently, these limitations can potentially diminish the overall accuracy and reliability of the OAA approach in multiclass classification scenarios.

Furthermore, many SMOTE variants require careful manual tuning of hyperparameters, which can be time-consuming and may necessitate domain expertise. These hyperparameters often control aspects such as the selection of minority class instances for synthesis and the determination of suitable synthesis regions. To mitigate the challenges associated with globally defined hyperparameters, several enhanced SMOTE variants [9], [10], [11], [12], [13], [14], [15] incorporate adaptive strategies. These techniques dynamically adjust key hyperparameters, such as those governing minority class categorization or the synthesis process, on an instance-by-instance basis. However, it is important to note that these adaptive methods often rely on a secondary layer of hyperparameters, whose values are not always explicitly exposed to the user, potentially increasing the complexity of the tuning process.

Despite the numerous variations of SMOTE proposed in recent years, the development of truly parameter-free implementations has received limited attention. While some research has explored parameter-free techniques for post-processing synthesized instances [16], these methods primarily focus on refining the output of existing SMOTE algorithms and do not address the fundamental issue of parameter dependence within the core SMOTE process. This necessitates the development of genuine parameter-free oversampling techniques that eliminate the need for manual hyperparameter tuning, thereby simplifying the application of SMOTE and its variants in real-world scenarios.

Despite these advances, there is a clear need for parameter-free oversampling technique. This paper introduces a novel parameter-free oversampling technique specifically designed to address the challenges of multiclass imbalanced learning. Building upon the foundational principles of the Extreme Anomalous Oversampling Technique (EXOT) [17], this research explores an enhanced framework that extends the capabilities of EXOT to effectively handle multiclass datasets.

The EXOT algorithm represents a significant departure from traditional SMOTE-based methods by eliminating the need for hyperparameter tuning. Unlike SMOTE, which heavily relies on the concept of nearest neighbors, EXOT leverages a set of three distinct anomalous scores to categorize minority instances and determine optimal synthesis regions. This innovative approach effectively circumvents the challenges associated with hyperparameter selection and tuning, which can often be time-consuming and require domain expertise.

This research aims to investigate the potential of parameter-free oversampling techniques in achieving optimal classifier performance across diverse datasets. A key component of this investigation involves integrating the proposed multiclass oversampling technique with automated machine learning (AutoML). AutoML, encompassing a suite of 15 distinct classifiers, will be employed to identify the most suitable classifier and its optimal hyperparameter configuration for each dataset after the application of the enhanced EXOT oversampling technique. This parameter-free approach, coupled with AutoML's ability to efficiently search through a diverse set of classifiers and their hyperparameter configurations, aims to achieve high classifier performance on multiclass imbalanced datasets while minimizing human intervention.

The paper make the following contributions:

- Multiclass imbalance: This research investigates and addresses the challenges posed by multiclass imbalanced learning, a prevalent issue in real-world applications, by acknowledging and overcoming the limitations of existing methods, particularly those associated with binary-class oversampling techniques and the complexities of hyperparameter tuning.
- Parameter-free method: This research introduces MEXT, a novel parameter-free oversampling technique specifically designed for multiclass imbalanced datasets, thereby addressing a critical need by eliminating the requirement for manual hyperparameter tuning, a significant bottleneck in many existing oversampling methods.
- Extension of EXOT: This paper investigates the properties of the anomalous scores utilized in the EXOT algorithm, providing a formal definition and extending its applicability to multiclass datasets by introducing the concept of the extreme anomalous score with respect to a dataset, enabling the MEXT algorithm to address multiclass imbalance without requiring class relabeling procedures.
- Use of anomalous score: MEXT leverages anomalous score analysis to identify optimal synthesis locations, departing from traditional neighbor-based approaches and offering a potentially more robust and effective solution for oversampling minority classes in imbalanced datasets.
- Extensive experiment over datasets and classifiers: This research encompasses an extensive experimental evaluation of the MEXT algorithm on a collection of multiclass datasets from the UCI repository, comparing its performance against several state-of-the-art oversampling algorithms and providing empirical evidence of its effectiveness.

The remainder of this paper is separated into seven sections. Section II provides a foundational understanding of the anomalous scoring concept employed in the EXOT algorithm, contrasting it with the neighbor-based and clustering approaches utilized in other SMOTE variants. Next, Section

III generalizes the EXOT concept by introducing the notion of an "extreme anomalous score with respect to a dataset" and comparing it to the three anomalous scores employed in the original EXOT algorithm. Section IV presents the proposed MEXT algorithm, a novel multi-class extreme anomalous oversampling technique. Section V details the experimental setup and methodology employed in this study. The experimental results are presented and discussed in Section VI and Section VII, respectively. Finally, the essences of this work are ultimately summarized in Section VIII.

II. Preliminary Knowledge

Anomalous scores quantify the degree of abnormality exhibited by individual instances within a dataset relative to their surrounding instances. In Euclidean space, dissimilarity between instances is typically quantified by Euclidean distance. Consequently, instances with greater distances to their nearest neighbors are generally considered more anomalous.

The Extreme Anomalous Score (EAS) is a metric specifically designed for numeric datasets to quantify the degree of isolation of an individual instance. Originally proposed for outlier detection, EAS has subsequently been employed in various applications, including clustering [18] and imbalanced classification [17].

Within the application of imbalanced classification, EAS plays a pivotal role in the EXOT algorithm, which is the parameter-free oversampling algorithm. EAS is defined for all instances independent of their classes. Formally, EAS for a given instance is defined as the radius of the largest open ball centered on that instance that contains no other instances [17]. In addition to EAS, the EXOT algorithm utilizes two class-dependent anomalous scores: the Negative Anomalous Score (NAS) and the Positive Anomalous Score (PAS). These scores are defined based on class labels, where the positive class typically represents the minority class in imbalanced classification problems. NAS of any instance is the largest radius of an open ball centered at that instance containing no other negative instances, while PAS is the largest radius of an open ball centered at that instance containing no other positive instances [17]. By leveraging these three distinct anomalous scores, the EXOT algorithm effectively circumvents the challenges associated with hyperparameter tuning, a common limitation encountered in many traditional SMOTE-based oversampling techniques.

The original SMOTE algorithm operates within the Euclidean space, necessitating the use of numerical attributes. It generates synthetic minority instances by interpolating between pairs of existing minority class instances. For each minority instance, SMOTE identifies its k nearest neighbors within the minority class. A new synthetic instance is then created along the line segment connecting the original minority instance to one of its randomly selected k-nearest neighbors.

The process of generating a synthetic instance can be mathematically expressed as follows:

$$\mathbf{x}_{syn} = \mathbf{x}_i + \gamma \cdot (\mathbf{x}_j - \mathbf{x}_i). \tag{1}$$

In (1), $\mathbf{x}_{syn}$ represents a synthetic minority instance, $\mathbf{x}_i$ denotes an original minority instance under consideration, $\mathbf{x}_j$ represents a randomly selected instance from the k nearest minority neighbors of $\mathbf{x}_i$, and γ is a uniformly distributed random number within the interval [0, 1]. The sole hyperparameter within the original SMOTE algorithm is the number of nearest neighbors, k.

For each synthesizing step, $\mathbf{x}_i$ is like the core of the synthesizing region, The vector $\mathbf{x}_j - \mathbf{x}_i$ defines the direction of synthesis, while the scalar γ (a random value between 0 and 1) determines the position of the synthesized instance $(\mathbf{x}_{syn})$ along this vector. The region to be densified depends on the $\mathbf{x}_i$ selection. The broadening of the minority region depends on the conditions to select $\mathbf{x}_j$, and γ. Variations of SMOTE diverge primarily in their strategies for selecting $\mathbf{x}_i$, $\mathbf{x}_j$, and the range of permissible γ values.

The original SMOTE and neighbor-based SMOTE variants such as Borderline-SMOTE [19] and Safe-Level SMOTE [20] define the synthesis region based on the k-nearest neighbors of each minority instance. These methods operate under the assumption that synthesizing new instances along the lines connecting neighboring minority instances will likely generate instances within the minority class region. The selection of the neighboring instance $(\mathbf{x}_j)$ for synthesis is typically performed randomly from the set of k nearest minority neighbors of $\mathbf{x}_i$.

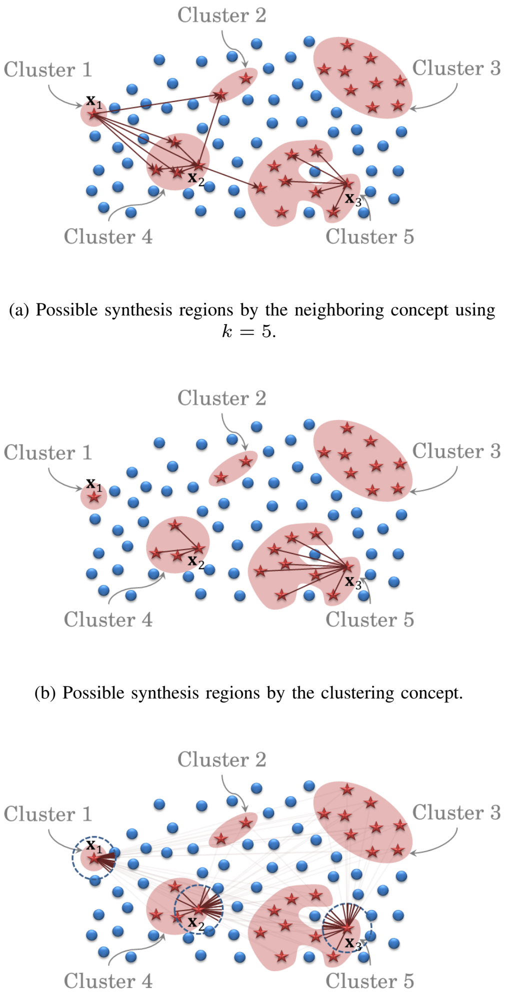
Clustering-based SMOTE variants, such as cluster-SMOTE, CE-SMOTE, DE-oversampling, kmeans-SMOTE, MWMOTE, and DDSC-SMOTE [21], [22], [23], [24], [25], [26], leverage clustering algorithms to identify dense regions within the minority class distribution. These methods synthesize new instances within these localized clusters. Specifically, the neighboring instance $(\mathbf{x}_j)$ for synthesis is selected randomly from the set of minority instances belonging to the same cluster as the original minority instance $(\mathbf{x}_i)$.

In the EXOT algorithm, the neighboring instance $(\mathbf{x}_j)$ serves solely to establish the unit direction vector emanating from the original minority instance $(\mathbf{x}_i)$. Consequently, the selection of $\mathbf{x}_j$ is not restricted to a specific neighborhood; any minority instance within the dataset, excluding $\mathbf{x}_i$ itself, can be utilized to define the direction of synthesis.

Most conventional SMOTE-based methods constrain the synthesized instance $(\mathbf{x}_{syn})$ to lie within the linear subspace defined by the original minority instance $(\mathbf{x}_i)$ and its selected neighbor $(\mathbf{x}_j)$. This constraint typically restricts the range of the interpolation parameter (γ) to the interval [0, 1]. When $\gamma = 0.5$, the synthesized instance $(\mathbf{x}_{syn})$ is equidistant from the original instance $(\mathbf{x}_i)$ and its neighbor $(\mathbf{x}_j)$. To generate instances closer to the original instance, γ is typically sampled from the interval [0, 0.5). Conversely, to generate instances closer to the neighboring instance, γ is sampled from the interval (0.5, 1].

In contrast, the EXOT algorithm extends the synthesis region beyond this linear subspace. EXOT allows for the generation of instances within a "safe region" surrounding the original instance $(\mathbf{x}_i)$, defined by the radii of two open balls. The first one is the Extreme Anomalous Ball (EAB): the largest open ball centered at $\mathbf{x}_i$ that contains no other instances. Its radius corresponds to the Extreme Anomalous Score (EAS) of $\mathbf{x}_i$. The second one is the Negative Anomalous Ball (NAB): the largest open ball centered at $\mathbf{x}_i$ that contains no instances from the majority class. Its radius corresponds to the Negative Anomalous Score (NAS) of $\mathbf{x}_i$. The interpolation parameter

(γ) is sampled from the interval $(0, Rad_{\mathbf{x}_i})$, where $Rad_{\mathbf{x}_i}$ represents the radius of either the EAB or the NAB, depending on the specific conditions defined within the EXOT algorithm.



(a) Possible synthesis regions by the neighboring concept using $k = 5$.

(b) Possible synthesis regions by the clustering concept.

(c) Possible synthesis regions by the anomalous scoring concept.

Fig. 1. Possible synthesis regions for $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$.

Fig. 1 illustrates possible synthesis regions for three representative minority instances regions ($\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$) using three different concepts: the neighboring concept, the clustering concept, and the anomalous scoring concept. In this visualization, majority class instances are represented by dots, while minority class instances are represented by stars. The minority instances in this figure exhibit a clustered distribution, forming small subclusters near one another within the feature space. The instances $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$ are representative of this clustered distribution, each belonging to a distinct cluster of varying size.

Fig. 1a demonstrates the application of the k-nearest neighbors approach (with $k = 5$) in defining synthesis regions. This approach, employed in the original SMOTE algorithm and its variants, utilizes the number of nearest neighbors as a global hyperparameter to determine the extent of the synthesis region. In this example, the arrows emanating from each instance ($\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$) indicate their five nearest minority neighbors. Instance $\mathbf{x}_3$ belongs to a relatively dense cluster (cluster 5) containing more than five instances. Therefore, its five nearest neighbors are all members of the same cluster. In contrast, instances $\mathbf{x}_1$ and $\mathbf{x}_2$ belong to smaller clusters. Consequently, some of their nearest neighbors may belong to other clusters. This reliance on nearest neighbors can introduce potential challenges. For instance, if the algorithm synthesizes instances along the line segment connecting a minority instance in one cluster to its nearest neighbor in another cluster, the synthesized instances may inadvertently fall within the majority class region. This can lead to misclassification issues, where the classifier erroneously labels majority class instances as belonging to the minority class. Furthermore, setting an appropriate value for the hyperparameter k can be challenging. Using an excessively large value for k may result in the generation of synthetic instances within the majority class region. Conversely, using a small value for k (e.g., $k = 1$) can lead to overfitting, as it essentially duplicates existing minority instances.

To address these challenges, adaptive approaches have been proposed that dynamically adjust the value of k for each instance. Additionally, clustering-based methods have been developed to mitigate the risk of synthesizing instances within the majority class region [26].

Fig. 1b illustrates potential synthesis regions for clustering-based SMOTE variants. To mitigate the risk of synthesizing instances within the majority class region, these methods partition the minority class into distinct clusters prior to the oversampling process. For instance $\mathbf{x}_1$, which belongs to a singleton cluster, the synthesis process cannot be directly applied due to the absence of neighboring minority instances within the same cluster. For instance $\mathbf{x}_2$, synthesis can proceed by selecting neighboring instances from within its respective cluster (cluster 4) without the risk of encroaching upon the majority class region. In contrast, synthesizing instances for $\mathbf{x}_3$, which belongs to a cluster containing two majority class instances, carries a higher risk of generating synthetic instances within the majority class region.

Fig. 1c illustrates the application of the EXOT algorithm, which utilizes anomalous scores to define the synthesis region. A key advantage of EXOT is its ability to incorporate all minority instances, including isolated instances such as $\mathbf{x}_1$, into the synthesis process. In this figure, the potential synthesis regions for each minority instance ($\mathbf{x}_1, \mathbf{x}_2$, and $\mathbf{x}_3$) are represented by the area within their respective Extreme Anomalous Balls (EABs) or Negative Anomalous Balls (NABs). In this specific example, the NAS of each instance is equal to its EAS, resulting in a single dashed circle representing both the EAB and NAB for each instance. This approach allows for the generation of synthetic instances in a more diverse range of positions within the feature space. The extent of the synthesis region for each instance is dynamically determined by its corresponding anomalous score, ensuring that the expansion of the minority class region does not encroach upon the majority class region.

A key limitation of many SMOTE variants arises from their reliance on hyperparameters to guide the synthesis process. These hyperparameters influence various aspects, such as the direction of synthesis and the extent of the synthesized region within the feature space. For instance, datasets containing isolated minority instances pose a significant challenge for many oversampling techniques. These techniques often neglect such instances unless specifically designed to synthesize within majority class regions. To address this limitation, several approaches have been proposed that categorize minority instances based on their local characteristics before applying the oversampling process.

One common approach involves categorizing minority instances based on their proximity to majority instances. These categories often include isolated minorities, safe minorities, and borderline minorities. Isolated minority instances are typically surrounded by majority class instances. Safe minority instances are located within dense regions of the minority class. Borderline minority instances reside near the boundary between the minority and majority class regions. These categorizations aim to guide the oversampling process by identifying instances that may require special handling. For example, borderline minorities, due to their proximity to the majority class, may be more susceptible to misclassification and therefore require more careful oversampling strategies.

Techniques such as borderline-SMOTEs (both borderline-SMOTE1 and borderline-SMOTE2), safe-level-SMOTE, MWMOTE, MDO, and FLEX-SMOTE [19], [20], [25], [27], [28] utilize the k-nearest neighbors of a minority instance to determine its category. By examining the proportion of minority and majority class instances among the k-nearest neighbors, these methods attempt to identify the minority instance's proximity to the majority class boundary.

While these neighbor-based approaches provide valuable insights, the accuracy of minority instance categorization can be sensitive to the choice of the parameter k. An inappropriate selection of k can lead to misclassification of minority instances, potentially impacting the effectiveness of the oversampling process.

To address this limitation, the EXOT algorithm utilizes anomalous scores to characterize minority instances and guide the synthesis process, eliminating the need for parameter-based neighbor analysis.

In EXOT, the dangerous minorities or the borderline minorities are identified as those that lie on the boundary of the Positive Anomalous Ball (PAB) of some majority class instance. The PAB of an instance **x** ($PAB_{\mathbf{x}}$) is defined as the largest open ball centered at **x** that contains no other minority instances. By definition, the radius of $PAB_{\mathbf{x}}$ corresponds to the Positive Anomalous Score (PAS) of **x**. These "sensitive positive instances" located on the boundary of a majority class's PAB, are particularly important as their presence significantly influences the positive anomalous scores of the surrounding majority class instances.

Most algorithms prioritize enhancing the accuracy of predicting the minority class, even if it results in a slight decrease in the accuracy of identifying the majority class. When applying binary classification algorithms to multiclass datasets using the OAA approach, the accuracy of predicting the combined minority class can be impacted. This is because synthetic instances generated for one minority class may extend beyond the boundaries of other minority class regions, potentially leading to misclassification.

The EXOT algorithm, by carefully generating synthetic instances within well-defined boundaries determined by anomalous scores, aims to minimize the impact on other minority class regions. However, applying EXOT to multiclass datasets still necessitates the use of the OAA approach, which can increase computational complexity due to the need to compute anomalous scores for each minority instance in each OAA classification. Consequently, further modifications to the EXOT algorithm may be necessary to optimize its performance for multiclass imbalanced learning scenarios.

III. Generalized Extreme Anomalous Oversampling Technique Concept

This section investigates the properties of the anomalous scores employed in the EXOT algorithm, commencing with a generalized definition that encompasses EAS, NAS, and PAS. This unified framework facilitates a more concise and rigorous analysis of their inherent properties, avoiding the redundancy of independent proofs for each individual score.

A. The Extreme Anomalous Score with Respect to a Dataset

Let $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$ denote a dataset comprising n instances, where each instance $\mathbf{x}_i$ is represented by an m-dimensional vector of real numbers, i.e., $\mathbf{x}_i = (x_{i,1}, ..., x_{i,m})$. The generalized definition of EAS, NAS, and PAS is formally presented in Definition III.1. In this context, $B(\mathbf{x}, r)$ represents an open ball centered at $\mathbf{x}$ with radius r.

Definition III.1. (Extreme Anomalous Score with Respect to a Dataset)
For dataset $A \subseteq X$, the extreme anomalous score of instance $\mathbf{x} \in X$ with respect to dataset A denoted by $EAS(\mathbf{x}, A)$ is defined as

$$EAS(\mathbf{x}, A) = \sup\left\{r > 0 \mid B\left(\mathbf{x}, r\right) \cap \left(A \backslash \left\{\mathbf{x}\right\}\right) = \emptyset\right\},$$

where $B\left(\mathbf{x}, r\right)$ is an open ball centered at $\mathbf{x}$ with radius r.

Notably, for a dataset comprising a single instance ($n = 1$), the instance is considered to be inherently anomalous. Consequently, its EAS with respect to any subset of the dataset is defined as infinity, as formally established in Proposition III.1. Conversely, for datasets containing multiple instances ($n > 1$), the EAS with respect to a dataset $A \subseteq X$ of any instance (**x**) within the dataset (X) is finite. This paper formally demonstrates that, in such cases, the EAS of an instance can be directly determined from its Euclidean distance to its nearest neighbor within the specified dataset (A), as established in Theorem III.1.

Proposition III.1. If X is a singleton and $\mathbf{x}_1$ is an instance of X, then $EAS(\mathbf{x}_1, A) = \infty$ for all $A \subseteq X$.

Proof: Given dataset X with exactly one instance $\mathbf{x}_1$, and A be a subset of X. Thus $A = \emptyset$ or $A = \{\mathbf{x}_1\}$. That is $A \setminus \{\mathbf{x}_1\} = \emptyset$. $B(\mathbf{x}, r) \cap (A \setminus \{\mathbf{x}_1\}) = \emptyset$ for all $r > 0$. By Definition III.1, $EAS(\mathbf{x}_1, A) = \infty$. ■

Theorem III.1. Given dataset X with $|X| > 1$. For dataset $A \subseteq X$ and instance $\mathbf{x} \in X$. If $A \setminus \{\mathbf{x}\} \neq \emptyset$, then

$$EAS(\mathbf{x}, A) = \min_{\mathbf{a} \in A \setminus \{\mathbf{x}\}} d(\mathbf{x}, \mathbf{a}).$$

Note that $d(\mathbf{x},\mathbf{a})$ is the Euclidean distance between $\mathbf{x}$ and $\mathbf{a}$, and $|X|$ denotes the cardinality of X.

Proof: Let X be a dataset containing n instances, where $n > 1$. Given subset A of X and instance $\mathbf{x}$ of X which $A \setminus \{\mathbf{x}\} \neq \emptyset$. Let $E = \{r > 0 \mid B(\mathbf{x}, r) \cap (A \setminus \{\mathbf{x}\}) = \emptyset\}$. Then for any $\mathbf{a} \in A \setminus \{\mathbf{x}\}$, $d(\mathbf{x},\mathbf{a}) \geq \varepsilon$ for any $\varepsilon \in E$. Therefore E is bounded above. There exists $\varepsilon^* > 0$, s.t. $\text{EAS}(\mathbf{x}, A) = \sup E = \varepsilon^*$.

Since $A \setminus \{\mathbf{x}\}$ is non-empty and finite, there exists instance $\mathbf{a}^*$ of $A \setminus \{\mathbf{x}\}$ such that $d(\mathbf{x},\mathbf{a}^*) = \min\limits_{\mathbf{a} \in A \setminus \{\mathbf{x}\}} d(\mathbf{x},\mathbf{a}) = \delta$. Thus $\forall \mathbf{a} \in A \setminus \{\mathbf{x}\}, \delta \leq d(\mathbf{x},\mathbf{a})$.

Since $\mathbf{a}^* \in A \setminus \{\mathbf{x}\}$, thus $d(\mathbf{x},\mathbf{a}^*) \geq \varepsilon$ for all ε. It means that δ in an upper bound of E , hence $\varepsilon^* \leq \delta$.

To prove that $\varepsilon^* = \delta$, it is sufficed to show that $\varepsilon^* \not< \delta$.

Assume that $\varepsilon^* < \delta$, that is $\varepsilon^* < \frac{\varepsilon^*+\delta}{2} < \delta$. Hence $\forall \mathbf{a} \in A \setminus \{\mathbf{x}\}, \frac{\varepsilon^*+\delta}{2} < \delta \leq d(\mathbf{x},\mathbf{a})$. That is $\forall \mathbf{a} \in A \setminus \{\mathbf{x}\}, \mathbf{a} \notin B\left(\mathbf{x}, \frac{\varepsilon^*+\delta}{2}\right)$. Thus $\frac{\varepsilon^*+\delta}{2} \in E$. Since $\frac{\varepsilon^*+\delta}{2} > \varepsilon^*$ and $\frac{\varepsilon^*+\delta}{2} \in E$, thus $\varepsilon^* \neq \sup E$ which is the contradiction. Therfore $EAS(\mathbf{x}, A) = \min\limits_{\mathbf{a} \in A \setminus \{\mathbf{x}\}} d(\mathbf{x},\mathbf{a})$. ■

Based on the findings of Proposition III.1 and Theorem III.1, it can be concluded that the EAS of an instance $\mathbf{x}$ with respect to dataset A is equivalent to the infimum of the set of distances between $\mathbf{x}$ and all other instances within dataset A, as stated in Corollary III.1.

Corollary III.1. Given dataset $A \subseteq X$ and instance $\mathbf{x} \in X$.

$$EAS(\mathbf{x}, A) = \inf\{d(\mathbf{x},\mathbf{a}) \mid \mathbf{a} \in A \setminus \{\mathbf{x}\}\}.$$

Proof: Let H be the set $\{d(\mathbf{x},\mathbf{a}) \mid \mathbf{a} \in A \setminus \{\mathbf{x}\}\}$.

Case 1: Given $A \setminus \{\mathbf{x}\} = \emptyset$. Thus $H = \emptyset$, and $\inf H = \inf \emptyset = \infty$. As shown in Proposition III.1, $EAS(\mathbf{x}, A) = \infty$ when $A \setminus \{\mathbf{x}\} = \emptyset$.

Case 2: Given $A \setminus \{\mathbf{x}\} \neq \emptyset$. Thus H is a non-empty finite set containing its infimum. Therefore $\inf H = \min\limits_{\mathbf{a} \in A \setminus \{\mathbf{x}\}} d(\mathbf{x},\mathbf{a}) = EAS(\mathbf{x}, A)$, by Theorem III.1.

From all cases, it can be concluded that $EAS(\mathbf{x}, A) = \inf\{d(\mathbf{x},\mathbf{a}) \mid \mathbf{a} \in A \setminus \{\mathbf{x}\}\}$. ■

Theorem III.1 and Corollary III.1 serve as foundational principles in the proof of Theorem III.2, which establishes the following property: the Extreme Anomalous Score (EAS) of an instance $\mathbf{x}$ with respect to a dataset A is always less than or equal to the EAS of the same instance $\mathbf{x}$ with respect to any subset S of dataset A.

Theorem III.2. Let S and A be subsets of X which $S \subseteq A$. For every instance $\mathbf{x} \in X$,

$$EAS\left(\mathbf{x}, S\right) \geq EAS\left(\mathbf{x}, A\right).$$

Proof: Given dataset $S \subseteq A \subseteq X$ and instance $\mathbf{x} \in X$.

Case 1: Given $A \setminus \{\mathbf{x}\} = \emptyset$. Since $S \subseteq A$, $S \setminus \{\mathbf{x}\} = \emptyset$. By Corollary III.1, $EAS\left(\mathbf{x}, A\right) = \infty = EAS\left(\mathbf{x}, S\right)$.

Case 2: Given $A \setminus \{\mathbf{x}\} \neq \emptyset$. By Theorem III.1, $\exists r^* > 0$ such that $r^* = EAS(\mathbf{x}, A)$, and $r^* \leq d(\mathbf{x},\mathbf{a})$ for every $\mathbf{a} \in A \setminus \{\mathbf{x}\}$.

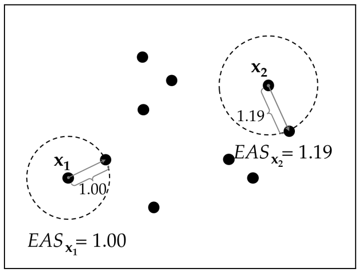


Fig. 2. Values of $EAS_{\mathbf{x_1}}$ and $EAS_{\mathbf{x_2}}$ in a dataset.

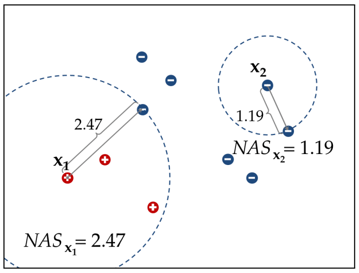


Fig. 3. Values of $NAS_{\mathbf{x_1}}$ and $NAS_{\mathbf{x_2}}$ in a dataset.

Case 2.1: Suppose $S \setminus \{\mathbf{x}\} = \emptyset$. Thus $EAS(\mathbf{x}, S) = \inf \emptyset = \infty > r^*$, by Corollary III.1.

Case 2.2: Suppose $S \setminus \{\mathbf{x}\} \neq \emptyset$. Let $\mathbf{s} \in S \setminus \{\mathbf{x}\}$. Since $S \subseteq A$, therefore $r^* \leq d(\mathbf{x},\mathbf{s})$ for all $\mathbf{s} \in S \setminus \{\mathbf{x}\}$. Hence r^* is a lower bound of set $\{d(\mathbf{x},\mathbf{s}) \mid \mathbf{s} \in S \setminus \{\mathbf{x}\}\}$. From Corollary III.1, $EAS(\mathbf{x}, S) = \inf\{d(\mathbf{x},\mathbf{s}) \mid \mathbf{s} \in S \setminus \{\mathbf{x}\}\}$. Because $EAS(\mathbf{x}, S)$ is the greatest lower bound and r^* is a lower bound of $\{d(\mathbf{x},\mathbf{s}) \mid \mathbf{s} \in S \setminus \{\mathbf{x}\}\}$, thus $EAS(\mathbf{x}, S) \geq r^*$.

In all cases, for any $\mathbf{x} \in X, EAS(\mathbf{x}, S) \geq EAS(\mathbf{x}, A)$. ■

The aforementioned theorems, proposition, and corollary collectively demonstrate the validity of the proposed framework for all anomalous scores employed within the EXOT algorithm, as they are all inherently equivalent to the extreme anomalous score with respect to a specific subset of the entire dataset. Building upon these foundational results, the subsequent sections utilize these theorems and definitions to elucidate the concepts of EAS, NAS, and PAS within the EXOT framework.

B. The Anomalous Scores in EXOT

The EXOT algorithm incorporates three distinct anomalous scores: the Extreme Anomalous Score (EAS), the Negative Anomalous Score (NAS), and the Positive Anomalous Score (PAS). Given an instance $\mathbf{x}$ within the dataset X, where N denotes the set of all negative instances and P denotes the set of all positive instances, the EAS, NAS, and PAS of $\mathbf{x}$ can be formally defined as $EAS(\mathbf{x}, X)$, $EAS(\mathbf{x}, N)$, and $EAS(\mathbf{x}, P)$, respectively, as per Definition III.1 [17].

The Extreme Anomalous Score (EAS) of an instance $\mathbf{x}$, denoted as $EAS_{\mathbf{x}}$, defines the radius of the Extreme Anomalous Ball (EAB) centered at $\mathbf{x}$. By definition, the EAB contains no instances other than $\mathbf{x}$ itself. Fig. 2 illustrates the concept of EAB for two instances, $\mathbf{x}_1$ and $\mathbf{x}_2$. The radii of the

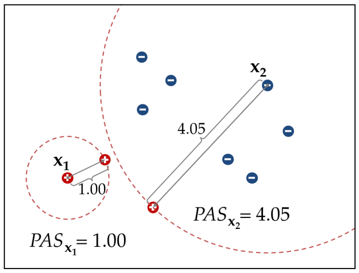


Fig. 4. Values of $PAS_{\mathbf{x}_1}$ and $PAS_{\mathbf{x}_2}$ in a dataset.

EABs, represented by dashed circles, visually demonstrate that $EAS_{\mathbf{x}_2}$ is greater than $EAS_{\mathbf{x}_1}$, reflecting the relative isolation of $\mathbf{x}_2$ within the dataset.

In the EXOT algorithm, the Negative Anomalous Score (NAS) of an instance **x**, denoted by $NAS_{\mathbf{x}}$, defines the radius of the Negative Anomalous Ball (NAB) centered at **x**. The NAB is characterized as the largest open ball centered at **x** that contains no other negative instances.

Fig. 3 illustrates the concept of NAB for two instances: a positive instance ($\mathbf{x}_1$) and a negative instance ($\mathbf{x}_2$). The dashed circles in the figure represent the boundaries of their respective NABs. Notably, while the NAB of $\mathbf{x}_2$ contains only itself, the NAB of $\mathbf{x}_1$ may contain instances other than $\mathbf{x}_1$, including instances from the positive class, as demonstrated in Fig. 3.

From the aforementioned examples, it is evident that the Extreme Anomalous Ball (EAB) and the Negative Anomalous Ball (NAB) of a positive instance both exclude negative instances. To facilitate the synthesis of new positive instances while maintaining the integrity of the negative region, the EXOT algorithm leverages both the EAS and NAS of each positive instance to define the permissible region for synthesis.

As illustrated in Fig. 2 and Fig. 3, the EAS of instance $\mathbf{x}_1$ is less than its NAS, while for instance $\mathbf{x}_2$, the EAS and NAS are equal. This observation is consistent with Theorem III.2, which formally demonstrates that the EAS of any instance is always less than or equal to its NAS.

The EXOT algorithm incorporates the Positive Anomalous Score (PAS) as an additional metric, not explicitly defined in the original EXOT paper [17]. The PAS of an instance **x**, denoted by $PAS_{\mathbf{x}}$, defines the radius of the Positive Anomalous Ball (PAB) centered at **x**, which is characterized as the largest open ball centered at **x** that contains no other positive (minority) instances.

Fig. 4 illustrates the concept of PAB for a positive instance ($\mathbf{x}_1$) and a negative instance ($\mathbf{x}_2$). The dashed circles in the figure represent the boundaries of their respective PABs. Notably, the positive instance located on the boundary of the negative instance's PAB signifies a critical point, representing the nearest positive instance to the negative instance. The EXOT algorithm identifies such instances as "sensitive positive instances".

The PAS of a negative instance plays a crucial role in identifying the boundary of the positive class region. The positive instance situated on the boundary of a negative instance's PAB effectively marks the edge of the positive class region. In real-world datasets, where class regions may exhibit overlap, the identification of these "sensitive positive instances" provides valuable information about the boundaries of the positive class region.

IV. Multiclass Extreme Anomalous Oversampling Technique (MEXT)

The MEXT algorithm, presented in Algorithm 1, extends the principles of the EXOT algorithm to effectively address the challenges of multiclass imbalanced datasets.

Initially, the dataset is partitioned into k subsets, each corresponding to a distinct class. Duplicate instances within each subset are subsequently removed. These subsets are then ordered in descending order based on their cardinality. The oversampling process proceeds iteratively, with each class being oversampled until it reaches the cardinality of the largest class.

For each class c, the algorithm commences by identifying sensitive positive instances and subsequently determines their corresponding synthesis regions based on their respective EAS values.

For non-sensitive positive instances (**p**) within class c, the algorithm compares $EAS(\mathbf{p}, X_{c_*})$ and $EAS(\mathbf{p}, X \setminus (X_c \cup X_{c_*}))$, where X_{c_*} represents the set of all instances belonging to classes with smaller cardinalities, and $X \backslash (X_c \cup X_{c_*})$ represents the set of all instances belonging to classes with larger cardinalities. If $EAS(\mathbf{p}, X_{c_*}) > EAS(\mathbf{p}, X \backslash (X_c \cup X_{c_*}))$, the MEXT algorithm utilizes $NAS_{\mathbf{p}}$ to determine the synthesized region surrounding **p**. This strategy is employed under the assumption that synthesizing instances in the $NAB_{\mathbf{p}}$ region will have minimal impact on a smaller class; otherwise, the synthesized region for **p** is determined by the minimum value between $NAS_{\mathbf{p}}$ and $EAS(\mathbf{p}, P_{sensitive})$, where $P_{sensitive}$ denotes the set of all sensitive positive instances. This constraint aims to prevent the generation of synthetic instances in close proximity to the region of the smaller class. Following the determination of synthesized regions for each positive instance, new instances are synthesized within these regions using the data generation technique employed in the EXOT algorithm.

Fig. 5 illustrates the synthesized regions for two representative instances from distinct classes. In this figure, $\mathbf{x}_1$ denotes a sensitive instance belonging to the smallest class, while $\mathbf{x}_2$ represents a non-sensitive instance from another class. Fig. 5a depicts the synthesized region for $\mathbf{x}_2$ as determined by the EXOT algorithm. This region, bounded by the NAB of $\mathbf{x}_2$, extends beyond the synthesized region of $\mathbf{x}_1$, which is bounded by its EAB. Consequently, the generation of synthetic instances within the synthesized region of $\mathbf{x}_2$ may potentially influence the classification of $\mathbf{x}_1$, potentially impacting the performance of the model with respect to the smallest class.

Fig. 5b illustrates the synthesized regions as determined by the MEXT algorithm. Since instance $\mathbf{x}_1$ from the smallest class resides on the boundary of the $NAB_{\mathbf{x}_2}$, the synthesized region for the non-sensitive instance $\mathbf{x}_2$ is constrained by $EAB(\mathbf{x}_2, P_{sensitive})$, where $P_{sensitive}$ represents the set of all sensitive positive instances. This constraint, visualized as a dotted circle in the figure, effectively limits the generation

Algorithm 1: The MEXT algorithm

```
Input  : Dataset $X$, Class label $\mathbf{y}$
Output: $X_{resampled}$, $\mathbf{y}_{resampled}$.
$X_{resampled} = \emptyset$;
$\mathbf{y}_{resampled} = [\,]$;
$C = \{c \mid c \in \mathbf{y}\}$;
for $c \in C$ do
    $X_c = \{\mathbf{x}_i \in X \mid y_i = c\}$;
end
$k = |C|$;
$\mathbf{c}_{sorted} = [c_1, c_2, ..., c_k]$ s.t. $|X_{c_1}| >> |X_{c_2}| >> ... >> |X_{c_k}|$;
$Th = |X_{c_1}|$;
$C_* = C$;
for $c \in \mathbf{c}_{sorted}$ do
    $C_* \leftarrow C_* \setminus \{c\}$;
    $X_{c_*} = \bigcup_{c_i \in C_*} X_{c_i}$;
    $X_{syn} = \emptyset$;
    if $|X_c| < Th$ then
        $n_{samples} = Th - |X_c|$;
        $P_{sensitive} =$
          $\{\mathbf{x}_i \in X_c \mid \exists \mathbf{n} \notin X_c, d(\mathbf{n}, \mathbf{x}_i) = EAS(\mathbf{n}, X_c)\}$;
        for $\mathbf{p}_i \in X_c$ do
            if $\mathbf{p}_i \in P_{sensitive}$ then
                $Rad_{\mathbf{p}_i} = EAS(\mathbf{p}_i, X)$;
            else if
              $EAS(\mathbf{p}_i, X_{c_*}) > EAS(\mathbf{p}_i, X \setminus (X_c \cup X_{c_*}))$
              then
                $Rad_{\mathbf{p}_i} = EAS(\mathbf{p}_i, X \setminus X_c)$;
            else
                $Rad_{\mathbf{p}_i} =$
                  $\min\{EAS(\mathbf{p}_i, X \setminus X_c), EAS(\mathbf{p}_i, P_{sensitive})\}$;
            end
        end
        while $n_{samples} > |X_{syn}|$ do
            for $\mathbf{p}_i \in X_c$ do
                $\gamma =$ random number between 0 and 1;
                $\mathbf{p}_j =$ random instance from $X_c \setminus \{\mathbf{p}_i\}$;
                $\mathbf{p}_{syn} = \mathbf{p}_i + \gamma \cdot Rad_{\mathbf{p}_i} \cdot \frac{\mathbf{p}_j - \mathbf{p}_i}{d(\mathbf{p}_i, \mathbf{p}_j)}$;
                $X_{syn} = X_{syn} \cup \{\mathbf{p}_{syn}\}$;
            end
        end
    end
    $n_c = |X_c \cup X_{syn}|$;
    $\mathbf{y}_{new} = [c, c, ..., c]_{1 \times n_c}$;
    $\mathbf{y}_{resampled} \leftarrow [\mathbf{y}_{resampled} \mid \mathbf{y}_{new}]$;
    $X_{resampled} \leftarrow X_{resampled} \cup X_c \cup X_{syn}$;
end
return $X_{resampled}$, $\mathbf{y}_{resampled}$
```

of synthetic instances in the vicinity of the smallest class, mitigating the potential for adverse effects on the classification of minority class instances.

It is important to note that when applied to a binary class imbalanced dataset, the MEXT algorithm operates in a manner analogous to the EXOT algorithm.

Let $d(\mathbf{p}_i, \mathbf{p}_{syn})$ represent the Euclidean distance between the original instance $\mathbf{p}_i$ and the synthesized instance $\mathbf{p}_{syn}$. For each synthesized instance $\mathbf{p}_{syn}$, if $\mathbf{p}_i$ is a sensitive positive instance, then $d(\mathbf{p}_i, \mathbf{p}_{syn})$ is less than or equal to $\text{EAS}_{\mathbf{p}_i}$; otherwise, $d(\mathbf{p}_i, \mathbf{p}_{syn})$ is less than or equal to $\text{NAS}_{\mathbf{p}_i}$.

The MEXT algorithm iteratively synthesizes new instances until all classes within the dataset achieve equal cardinality. Upon completion, the algorithm returns the balanced dataset, denoted as $X_{resampled}$, along with the corresponding class labels, $\mathbf{y}_{resampled}$.

V. Experiment

This section presents a comparative evaluation of the MEXT algorithm against a suite of state-of-the-art oversampling techniques, including SMOTE, ADASYN, Safe-Level SMOTE (SLS), MDO, and DSRBF. SMOTE, ADASYN, and SLS are widely recognized algorithms within the SMOTE family, readily available in various software modules. MDO and DSRBF represent contemporary multiclass oversampling techniques, both accessible within the smote-variants package.

A. Datasets

The experimental evaluation was conducted on a collection of 36 imbalanced datasets sourced from the UCI Machine Learning Repository [29]. Table I provides a summary of these datasets, ordered by their Multiclass Imbalance Ratio (MIR). The MIR is computed as follows:

$$MIR = \sum_{i=1}^{k-1} \sum_{j>i} \left(\frac{n_{c_i}}{n_{c_j}} - 1 \right), \tag{2}$$

where n_{c_i} and n_{c_j} represent the number of instances in classes c_i and class c_j, respectively, with $n_{c_i} \geq n_{c_j}$ for all $i, j \in \{1, 2, ..., k\}$ and $j > i$. This metric quantifies the degree of class imbalance within a dataset. A value of $MIR = 0$ indicates perfect class balance, while any non-zero value signifies the presence of class imbalance.

B. Oversampling Methods

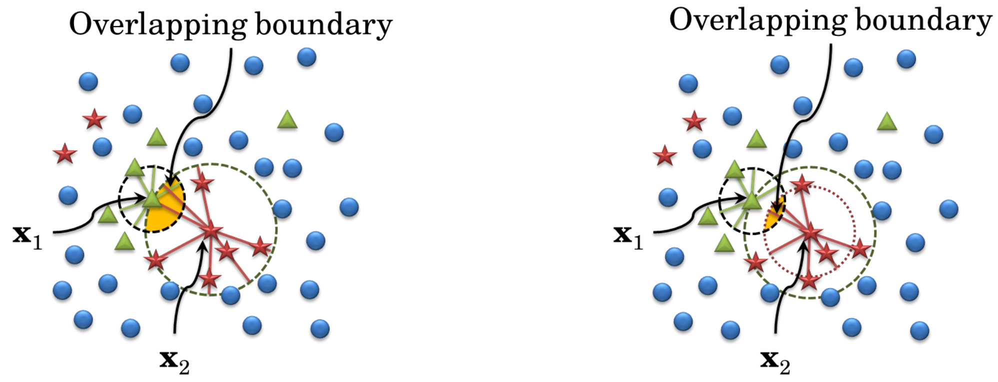
For comparative analysis, a grid search was conducted to determine optimal hyperparameter values for each of the following oversampling methods: SMOTE, ADASYN, SLS, MDO, and DSRBF. The grid search considered values of 5, 10, 15, and 20 for the relevant hyperparameters [30]. Each dataset was oversampled to achieve class balance, ensuring an equal number of instances in each class.

C. Base Classifiers

The primary focus of this study lies in evaluating the effectiveness of various oversampling techniques for addressing class imbalance in multiclass scenarios. Consequently, the assessment of these techniques was conducted by evaluating the performance of classifiers trained on the oversampled datasets. To optimize classifier performance, an automated machine learning (AutoML) framework was employed to identify the optimal classifier and its corresponding hyperparameter configuration for each dataset, ensuring unbiased evaluation of the oversampling methods.

D. Experimental Procedure

All experiments were conducted within a Jupyter Notebook environment hosted on Google Colaboratory [31], utilizing the Ubuntu 18.04 operating system on an Intel Xeon processor with 13022KB RAM. The smote-variants package served as the primary implementation source for all comparative oversampling techniques [8].



(a) The regions from the EXOT algorithm

(b) The regions from the MEXT algorithm

Fig. 5. Example of the synthesized regions for $\mathbf{x}_1$ and $\mathbf{x}_2$ using the EXOT and the MEXT algorithm.

TABLE I. DATA DESCRIPTIONS

Notations	Datasets	Instances	Att.	Classes	Description of classes	Class distribution	Avg	MIR
D1	Sonars	208	60	2	‘M’, ‘R’	111, 97	104	0.14
D2	Banknote	1372	4	2	‘0’, ‘1’	762, 610	686	0.25
D3	Vehicle	846	18	4	‘bus’, ‘saab’, ‘opel’, ‘van’	218, 217, 212, 199	211.50	0.31
D4	Audit	772	26	2	‘not risk’, ‘risk’	467, 305	386	0.53
D5	Magic	19020	10	2	‘gamma’, ‘hadron’	12332, 6688	9510	0.84
D6	Breast Cancer	683	9	2	‘malignant’, ‘benign’	444, 239	341.5	0.86
D7	Pima	768	8	2	‘inliers’, ‘outliers’	500, 268	384	0.87
D8	Haberman	306	3	2	‘died within 5 years’, ‘survived 5 years or longer’	225, 81	153	1.78
D9	Parkinsons	195	22	2	‘healthy’, ‘Parkinson’s’	147, 48	97.5	2.06
D10	Blood	748	4	2	’not donate’, ‘donate’	570, 178	374	2.20
D11	Vertebral	310	6	3	‘SL’, ‘NO’, ‘DH’	150, 100, 60	103.33	2.67
D12	Gastroenterology	152	466	3	‘adenoma’, ‘hyperplasic’, ‘serrated’	80, 42, 30	50.67	2.97
D13	Breast Tissue 4c	106	9	4	‘fad&mas&gla’, ‘adi’, ‘car’, ‘con’	49, 22, 21, 14	26.5	6.18
D14	Climate	540	18	2	‘failure’, ‘success’	494, 46	270	9.74
D15	Satimage	6435	36	6	1.0, 7.0, 3.0, 5.0, 2.0, 4.0	1533, 1508, 1358, 707, 703, 626	1072.5	11.02
D16	Ozone8hr	1847	72	2	‘0’, ‘1’	1719, 128	923.5	12.43
D17	Glass	214	9	5	‘2’, ‘1’, ‘7’, ‘5&6’, ‘3’	76, 70, 29, 22, 17	42.80	15.66
D18	Cannabis	1885	12	7	‘CL6’, ‘CL0’, ‘CL2’, ‘CL3’, ‘CL1’, ‘CL5’, ‘CL4’	463, 413, 266, 211, 207, 185, 140	269.29	16.26
D19	Ecoli	327	7	5	‘cp’, ‘im’, ‘pp’, ‘imU’, ‘om’	143, 77, 52, 35, 20	65.40	19.21
D20	Nicotine	1885	12	7	‘CL6’, ‘CL0’, ‘CL2’, ‘CL1’, ‘CL3’, ‘CL5’, ‘CL4’	610, 428, 204, 193, 185, 157, 108	269.29	26.48
D21	Ozone1hr	1848	72	2	‘0’, ‘1’	1791, 57	924	30.42
D22	Benzodiazepine	1885	12	7	‘CL0’, ‘CL3’, ‘CL2’, ‘CL4’, ‘CL1’, ‘CL6’, ‘CL5’	1000, 236, 234, 120, 116, 95, 84	269.29	53.87
D23	Amphetamines	1885	12	7	‘CL0’, ‘CL2’, ‘CL1’, ‘CL3’, ‘CL6’, ‘CL4’, ‘CL5’	976, 243, 230, 198, 102, 75, 61	269.29	65.00
D24	Legal highs	1885	12	7	‘CL0’, ‘CL3’, ‘CL2’, ‘CL4’, ‘CL6’, ‘CL5’, ‘CL1’	1094, 323, 198, 110, 67, 64, 29	269.29	121.97
D25	Alcohol	1885	12	7	‘CL5’, ‘CL6’, ‘CL4’, ‘CL3’, ‘CL2’, ‘CL1’, ‘CL0’	759, 505, 287, 198, 68, 34, 34	269.29	126.34
D26	Ecstasy	1885	12	7	‘CL0’, ‘CL3’, ‘CL2’, ‘CL4’, ‘CL1’, ‘CL5’, ‘CL6’	1021, 277, 234, 156, 113, 63, 21	269.29	130.34
D27	Methadone	1885	12	7	‘CL0’, ‘CL3’, ‘CL2’, ‘CL6’, ‘CL4’, ‘CL5’, ‘CL1’	1429, 149, 97, 73, 50, 48, 39	269.29	147.56
D28	Cocaine	1885	12	7	‘CL0’, ‘CL2’, ‘CL3’, ‘CL1’, ‘CL4’, ‘CL5’, ‘CL6’	1038, 270, 258, 160, 99, 41, 19	269.29	157.86
D29	LSD	1885	12	7	‘CL0’, ‘CL1’, ‘CL3’, ‘CL2’, ‘CL4’, ‘CL5’, ‘CL6’	1069, 259, 214, 177, 97, 56, 13	269.29	192.20
D30	Yeast	1479	8	9	‘CYT’, ‘NUC’, ‘MIT’, ‘ME3’, ‘ME2’, ‘ME1’, ‘EXC’, ‘VAC’, ‘POX’	463, 429, 244, 163, 51, 44, 35, 30, 20	164.33	192.27
D31	Caffeine	1885	12	7	‘CL6’, ‘CL5’, ‘CL4’, ‘CL3’, ‘CL0’, ‘CL2’, ‘CL1’	1385, 273, 106, 60, 27, 24, 10	269.29	361.30
D32	Heroin	1885	12	7	‘CL0’, ‘CL2’, ‘CL1’, ‘CL3’, ‘CL4’, ‘CL5’, ‘CL6’	1605, 94, 68, 65, 24, 16, 13	269.29	384.58
D33	DrugMushrooms	1885	12	7	‘CL0’, ‘CL3’, ‘CL2’, ‘CL1’, ‘CL4’, ‘CL5’, ‘CL6’	982, 275, 260, 209, 115, 40, 4	269.29	525.95
D34	DrugVSA	1885	12	7	‘CL0’, ‘CL1’, ‘CL2’, ‘CL3’, ‘CL5’, ‘CL4’, ‘CL6’	1455, 200, 135, 61, 14, 13, 7	269.29	571.84
D35	DrugKetamine	1885	12	7	‘CL0’, ‘CL2’, ‘CL3’, ‘CL1’, ‘CL4’, ‘CL5’, ‘CL6’	1490, 142, 129, 45, 42, 33, 4	269.29	610.53
D36	Avila	20867	10	12	‘A’, ‘F’, ‘E’, ‘I’, ‘X’, ‘H’, ‘G’, ‘D’, ‘Y’, ‘C’, ‘W’, ‘B’	8572, 3923, 2190, 1663, 1044, 1039, 893, 705, 533, 206, 89, 10	1738.92	2487.61

E. Evaluation Metrics

To evaluate the performance of the classifiers, a standard train-test split was employed. Each dataset was divided into a training set (80% of the data) used for model training and a testing set (20% of the data) used for independent evaluation.

To ensure robust model selection, 5-fold cross-validation was performed on the training set during the model training and hyperparameter tuning process. The performance of the final, optimally configured classifiers was then assessed on the held-out testing set using four commonly employed metrics for multiclass imbalanced learning: macro-precision, macro-recall, F_1-measure, and G-mean [32].

Let TP_{c_i} denote the number of true positives for class c_i, FP_{c_i} denote the number of false positives for class c_i, and FN_{c_i} denote the number of false negatives for class c_i. The evaluation metrics are computed as follows:

$$\text{Precision}_{macro} = \frac{1}{|C|}\sum_{i=1}^{|C|}\frac{TP_{c_i}}{TP_{c_i}+FP_{c_i}} \tag{3}$$

$$\text{Recall}_{macro} = \frac{1}{|C|}\sum_{i=1}^{|C|}\frac{TP_{c_i}}{TP_{c_i}+FN_{c_i}} \tag{4}$$

$$F_{1macro} = \frac{2\cdot\text{Precision}_{macro}\cdot\text{Recall}_{macro}}{\text{Precision}_{macro}+\text{Recall}_{macro}} \tag{5}$$

$$G\text{-mean} = \sqrt[|c|]{(\prod_{i=1}^{|C|}\frac{TP_{c_i}}{TP_{c_i}+FN_{c_i}})} \tag{6}$$

F. Statistical Testing

To evaluate whether the performances of the optimal classifier from autoML over various datasets after applying MEXT (parameter-free method) differ from the ones applied with benchmark methods or not, statistical testing was then used. Wilcoxon signed-rank test which is a non-parametric statistical hypothesis test [33] was used for comparing a pair of oversampling methods: MEXT versus each other methods. The null and alternative hypotheses for two-tailed Wilcoxon signed-rank test were set as follows:

$$H_0 : M_1 - M_2 = 0,$$

$$H_1 : M_1 - M_2 \neq 0,$$

where M_1 denotes the median of the results from MEXT while M_2 denotes the one of compared method.

This Wilcoxon ranks the differences in the performance of a classifier for each dataset which were rebalanced by two oversampling methods. This ranking sorts the difference values in ascending by ignoring the signs and the zero differences. Then it compares the sum of ranks for the positive and negative differences called R^+ and R^-, respectively. The statistical value T is obtained from $\min\{R^+, R^-\}$. If T is from R^+, then the compared method is better; otherwise, EXOT is better. With a level of significance $\alpha = 0.05$, the null hypothesis is rejected in favor of the alternative hypothesis if T is smaller than the critical value which depends on the number of non-zero differences and α value.

VI. Results

This section presents the experimental results obtained using AutoML and six oversampling techniques: SMOTE, ADASYN, SLS, MDO, DSRBF, and MEXT, on a collection of UCI datasets. The performance of these techniques was evaluated using four metrics (macro-precision, macro-recall, F1-measure, and G-mean) for multiclass imbalanced learning, employing the optimal classifiers identified by AutoML for each oversampled dataset. The performance of AutoML without oversampling (labeled as **None**) serves as a baseline for comparison.

A heatmap (Fig. 6) visually represents the performance of each technique across datasets, with color intensity indicating performance levels. Darker hues signify superior performance, while lighter hues represent lower performance. The results demonstrate that most methods exhibit improved performance on datasets with lower Multiclass Imbalance Ratio (MIR). However, some datasets exhibited inferior performance with oversampling compared to the baseline, indicating that the evaluated oversamplers may not be universally effective.

To facilitate comparison, the average performance of each technique across all datasets was calculated for each metric. Fig. 7 illustrates these average performances along with their standard deviations. The results demonstrate that all oversampling techniques, on average, enhance classification performance compared to the baseline. However, no significant performance differences were observed among the six techniques.

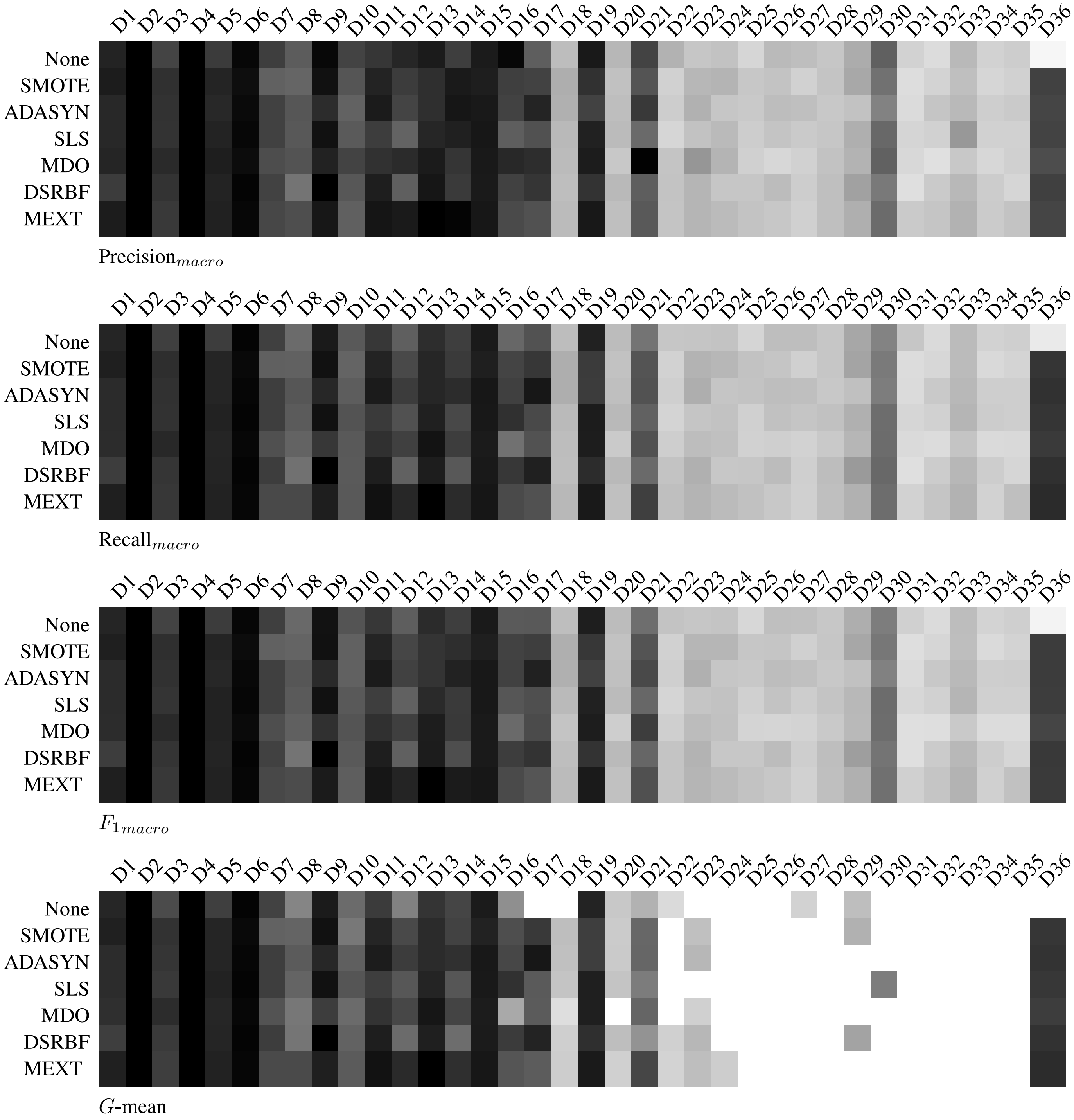
To further analyze the relative performance, the techniques were ranked within each dataset, with lower ranks indicating better performance. Fig. 8 presents the average ranks of each technique across all datasets. This analysis revealed that three multiclass oversampling approaches (MDO, DSRBF, and MEXT) exhibited superior macro-precision compared to the binary-class oversampling approaches (SMOTE, ADASYN, and SLS). However, MDO and DSRBF did not consistently outperform binary approaches in terms of macro-recall, F1-measure, and G-mean. In contrast, MEXT demonstrated superior performance across all four metrics, exhibiting both the highest average performance and the lowest average rank.

To statistically validate the superior performance of MEXT, a Wilcoxon signed-rank test was conducted. Table II presents the results of the Wilcoxon signed-rank test, conducted under the null hypothesis that there is no statistically significant difference in performance between MEXT and each of the five comparative oversampling methods (SMOTE, ADASYN, SLS, MDO, and DSRBF), all employing their respective optimal hyperparameter configurations determined via grid search.

The results, presented in Table II, indicate that MEXT significantly outperforms SMOTE in terms of macro-precision, macro-recall, and F1-measure; ADASYN in terms of macro-recall and F1-measure; SLS in terms of macro-precision and F1-measure; and MDO in terms of macro-recall, F1-measure, and G-mean. These statistically significant differences provide strong evidence of MEXT's superior performance compared to the other evaluated oversampling techniques.

VII. Discussion

The empirical findings affirm that oversampling methodologies, in general, can yield improvements in classification performance when applied to multiclass imbalanced datasets, as evidenced by the enhancement observed relative to the baseline condition. However, the heatmap visualization (Fig. 6) reveals a discernible heterogeneity in the efficacy of these techniques across the diverse datasets examined, with instances of decreased performance observed post-oversampling. This



Precision_{macro}

Recall_{macro}

$F_{1\,macro}$

G-mean

Fig. 6. Heatmaps of the performance of the oversampling methods on 36 datasets.

TABLE II. STATISTICAL RESULTS FROM THE WILCOXON SIGNED-RANK TEST COMPARING MEXT AGAINST 5 OVERSAMPLING METHODS

Methods	**Precision**$_{macro}$			**Recall**$_{macro}$			$F_{1\,macro}$			G**-mean**		
MEXT vs	T	R^+	p-value	T	R^+	p-value	T	R^+	p-value	T	R^+	p-value
SMOTE	157	404	**0.0273**	123	438	**0.0049**	132	429	**0.0080**	77	199	0.0636
ADASYN	183	412	0.0503	173	422	**0.0333**	180	415	**0.0446**	88	188	0.1283
SLS	154	441	**0.0142**	183	412	0.0503	126	469	**0.0034**	101	199	0.1615
MDO	213	382	0.1485	26	569	**0.0000**	75	520	**0.0001**	13	263	**0.0001**
DSRBF	218	377	0.1741	206	389	0.1177	216	379	0.1635	117	183	0.3458

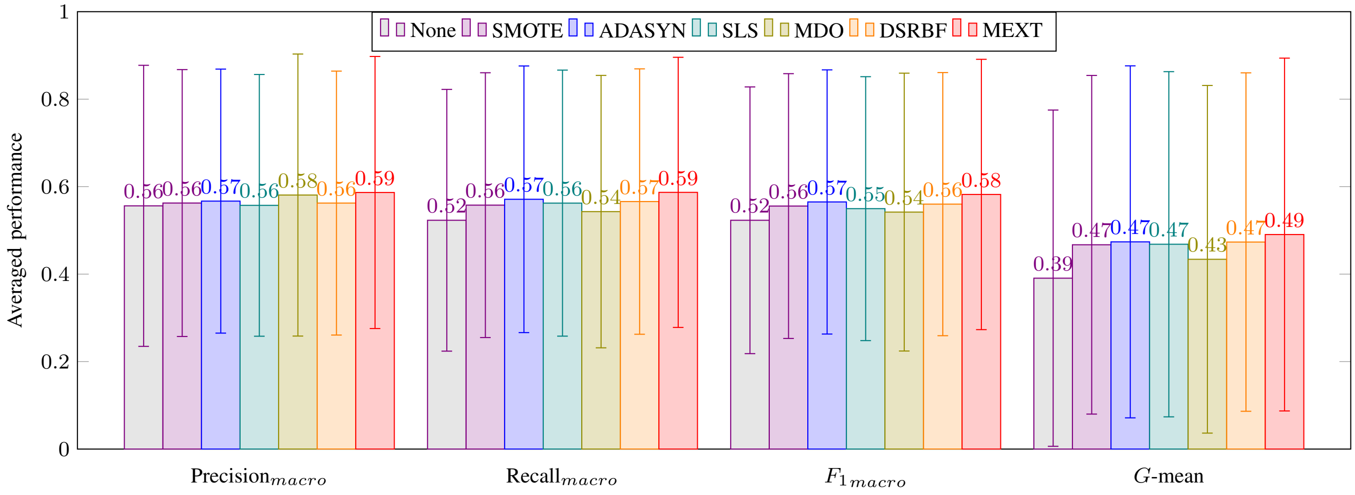


Fig. 7. Bar charts of the performances of the oversampling methods averaged across all datasets.

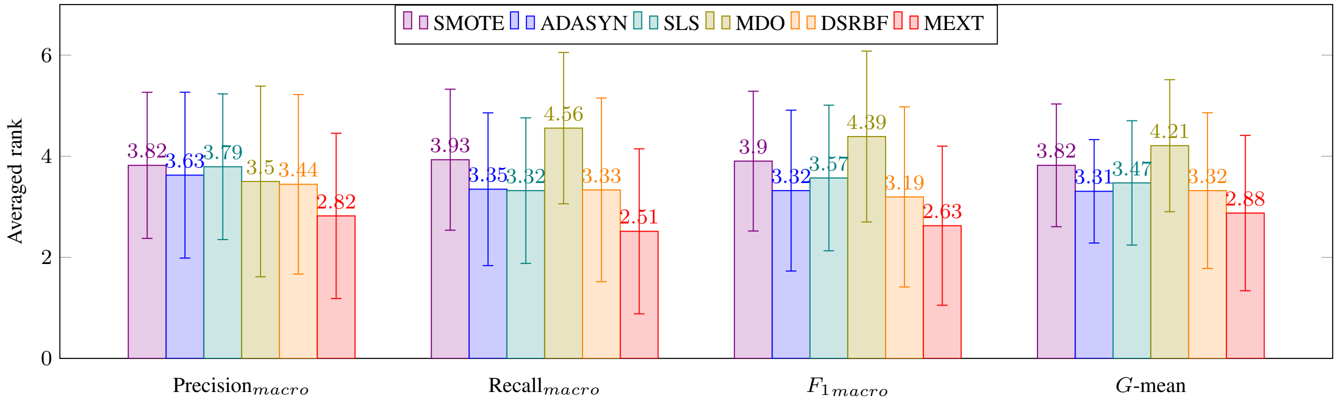


Fig. 8. Bar charts of ranking on four measurements of the oversampling methods averaged across all datasets.

variability underscores the critical importance of considering inherent dataset characteristics, such as the Multiclass Imbalance Ratio (MIR), as pivotal factors in the selection of an appropriate oversampling strategy.

The observation that all oversampling techniques, on average, resulted in enhanced performance compared to the baseline condition suggests that addressing class imbalance is a significant determinant of improved classification outcomes. Nonetheless, the absence of statistically significant differences in average performance among the six techniques (Fig. 7) implies that the choice of oversampling technique alone does not fully account for performance variation. This observation necessitates a more comprehensive investigation into other potentially influential factors, including but not limited to classifier selection and hyperparameter optimization, which may exert a substantial influence on classification performance.

The ranking analysis (Fig. 8) demonstrably illustrates that MEXT consistently outperformed the comparative techniques across all evaluation metrics, achieving both the highest average performance and the lowest average rank. This superior performance is further substantiated by the statistical significance demonstrated through the Wilcoxon signed-rank test (Table II). These results provide compelling evidence of MEXT's effectiveness in mitigating the challenges posed by multiclass imbalance, particularly when juxtaposed with both binary and alternative multiclass oversampling techniques.

The statistically significant improvements of MEXT over SMOTE, ADASYN, SLS, and MDO across multiple metrics provide robust evidence for its efficacy. The consistent performance of MEXT across all metrics suggests that it offers a robust and effective solution for multiclass imbalanced learning challenges.

VIII. Conclusion

This study introduces MEXT, a novel parameter-free oversampling technique designed to address the challenges of multiclass imbalanced datasets. Building upon the EXOT algorithm, MEXT enhances accessibility by utilizing a generalized extreme anomalous score, thereby eliminating the need for class-specific conversions. Furthermore, the inclusion of a synthesis region shrinking mechanism ensures the generation of high-quality synthetic data. The experimental results provide compelling evidence that MEXT consistently outperforms state-of-the-art oversampling techniques, particularly in terms of the F_1-measure across diverse datasets. This superior performance, achieved without hyperparameter optimization, highlights MEXT's potential as a valuable tool for researchers and practitioners tackling multiclass imbalanced learning problems.

Like the SMOTE-variance algorithm, MEXT cannot directly handle categorical variables. The required transformation of these variables to a numerical format is problem-specific and must be addressed by the user. Looking ahead, future research should focus on expanding the applicability of MEXT to a wider array of domains and datasets, including those with high dimensionality and complex data distributions. Specifically, investigating the algorithm's performance on real-world datasets with varying degrees of imbalance and noise would provide valuable insights into its robustness. Moreover, while MEXT is designed to be parameter-free, a thorough analysis of the impact of potential hyperparameter configurations, such as the parameters related to the synthesis region shrinkage, would further refine its performance and provide a deeper understanding of its behavior. Exploring adaptive mechanisms for these parameters could also lead to further performance gains. Additionally, integrating MEXT with other advanced techniques like deep learning models for imbalanced data could unlock new avenues for research and practical applications.

Acknowledgment

This research was supported in part by the Development and Promotion of Science and Technology Talents project (DPST) and the Applied Mathematics and Computational Science Program within the Department of Mathematics and Computer Science, Faculty of Science, at Chulalongkorn University, Thailand. The authors express their sincere gratitude to the anonymous reviewers for their valuable feedback. The authors acknowledge the utilization of a Large Language Model (LLM) to enhance the clarity and coherence of this manuscript. However, the scientific content and conclusions presented herein remain the sole responsibility of the authors.

References

[1] Q. Yang and X. Wu, "10 challenging problems in data mining research," *International Journal of Information Technology & Decision Making*, vol. 5, no. 4, pp. 597–604, 2006.

[2] L. Mena and J. A. Gonzalez, "Symbolic one-class learning from imbalanced datasets: application in medical diagnosis," *International Journal on Artificial Intelligence Tools*, vol. 18, no. 2, pp. 273–309, 2009.

[3] M. Kubat, R. C. Holte, and S. Matwin, "Machine learning for the detection of oil spills in satellite radar images," *Machine learning*, vol. 30, no. 2–3, pp. 195–215, 1998.

[4] W.-Z. Lu and D. Wang, "Ground-level ozone prediction by support vector machine approach with a cost-sensitive classification scheme," *Science of the total environment*, vol. 395, no. 2, pp. 109–116, 2008.

[5] D. P. Williams, V. Myers, and M. S. Silvious, "Mine classification with imbalanced data," *IEEE Geoscience and Remote Sensing Letters*, vol. 6, no. 3, pp. 528–532, 2009.

[6] Dimensions, "Dimensions," Accessed: May. 25, 2023. [Online]. Available: https://app.dimensions.ai/discover/publication

[7] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[8] G. Kovács, "smote-variants: a python implementation of 85 minority oversampling techniques," *Neurocomputing*, vol. 366, pp. 352–354, 2019.

[9] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on.* IEEE, 2008, pp. 1322–1328.

[10] S. Wang, Z. Li, W. Chao, and Q. Cao, "Applying adaptive oversampling technique based on data density and cost-sensitive svm to imbalanced learning," in *The 2012 International Joint Conference on Neural Networks (IJCNN)*, 2012, pp. 1–8.

[11] B. Tang and H. He, "KernelADASYN: Kernel based adaptive synthetic data generation for imbalanced learning," in *2015 IEEE Congress on Evolutionary Computation (CEC)*, 2015, pp. 664–671.

[12] I. Nekooeimehr and S. K. Lai-Yuen, "Adaptive semi-unsupervised weighted oversampling (a-suwo) for imbalanced datasets," *Expert Systems with Applications*, vol. 46, pp. 405–416, 2016.

[13] W. Siriseriwan and K. Sinapiromsaran, "Adaptive neighbor synthetic minority oversampling technique under 1nn outcast handling," *Songklanakarin Journal of Science and Technology*, vol. 39, pp. 565–576, 09 2017.

[14] J. Li, S. Fong, R. K. Wong, and V. W. Chu, "Adaptive multi-objective swarm fusion for imbalanced data classification," *Information Fusion*, vol. 39, pp. 1–24, 2018.

[15] Z. Huang, C. Yang, X. Chen, K. Huang, and Y. Xie, "Adaptive oversampling method for classification with application to imbalanced datasets in aluminum electrolysis," *Neural computing and applications*, vol. 32, pp. 7183–7199, 2020.

[16] Y. Yan, R. Liu, Z. Ding, X. Du, J. Chen, and Y. Zhang, "A parameter-free cleaning method for smote in imbalanced classification," *IEEE Access*, vol. 7, pp. 23 537–23 548, 2019.

[17] C. Chiamanusorn and K. Sinapiromsaran, "Extreme anomalous oversampling technique for class imbalance," in *Proceedings of the 2017 International Conference on Information Technology*, ser. ICIT 2017. New York, NY, USA: ACM, 2017, pp. 341–345.

[18] P. Lisuwan, P. Boonserm, and K. Sinapiromsaran, "Extreme anomalous score clustering algorithm," in *Proceedings of the 2017 International Conference on Information Technology*, ser. ICIT 2017. New York, NY, USA: ACM, 2017, pp. 66–70.

[19] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning," *Advances in intelligent computing*, pp. 878–887, 2005.

[20] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "Safe-level-SMOTE: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem," *Advances in knowledge discovery and data mining*, pp. 475–482, 2009.

[21] D. Cieslak, N. Chawla, and A. Striegel, "Combating imbalance in network intrusion datasets," in *2006 IEEE International Conference on Granular Computing*, 01 2006, pp. 732–737.

[22] S. Chen, G. Guo, and L. Chen, "A new over-sampling method based on cluster ensembles," in *2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops*, 2010, pp. 599–604.

[23] L. Chen, Z. Cai, L. Chen, and Q. Gu, "A novel differential evolution-clustering hybrid resampling algorithm on imbalanced datasets," in *2010 Third International Conference on Knowledge Discovery and Data Mining*, 2010, pp. 81–85.

[24] G. Douzas, F. Bação, and F. Last, "Improving imbalanced learning through a heuristic oversampling method based on k-means and smote," *Information Sciences*, vol. 465, 06 2018.

[25] S. Barua, M. M. Islam, X. Yao, and K. Murase, "MWMOTE–majority weighted minority oversampling technique for imbalanced data set learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 2, pp. 405–425, Feb 2014.

[26] X. Li and Q. Liu, "DDSC-SMOTE: an imbalanced data oversampling algorithm based on data distribution and spectral clustering," *The Journal of Supercomputing*, vol. 80, pp. 17 760–17 789, 2024.

[27] L. Abdi and S. Hashemi, "To combat multi-class imbalanced problems by means of over-sampling techniques," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 1, pp. 238–251, 2016.

[28] C. Bunkhumpornpat, E. Boonchieng, V. Chouvatut, and D. Lipsky, “FLEX-SMOTE: Synthetic over-sampling technique that flexibly adjusts to different minority class distributions,” *Patterns*, vol. 5, no. 11, p. 101073, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666389924002320

[29] M. Lichman, “UCI machine learning repository,” 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[30] J. Bergstra and Y. Bengio, “Random search for hyper-parameter optimization,” *The Journal of Machine Learning Research*, vol. 13, pp. 281–305, 03 2012.

[31] Google, “Colaboratory-Google,” Accessed: Nov. 11, 2022. [Online]. Available: https://research.google.com/colaboratory/faq.html

[32] H. He and E. A. Garcia, “Learning from imbalanced data,” *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[33] J. Demšar, “Statistical comparisons of classifiers over multiple data sets,” *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.