*mathematics*

*Article*

# Exploring Data Augmentation and Active Learning Benefits in Imbalanced Datasets

**Luis Moles [1,2,*], Alain Andres [1], Goretti Echegaray [2] and Fernando Boto [3]**

[1] TECNALIA, Basque Research and Technology Alliance (BRTA), Parque Científico y Tecnológico de Gipuzkoa, 20009 Donostia-San Sebastián, Spain
[2] Department of Computer Sciences and Artificial Intelligence, University of the Basque Country (UPV/EHU), 20018 Donostia-San Sebastián, Spain
[3] Faculty of Engineering, University of Deusto, 20012 Donostia-San Sebastián, Spain
* Correspondence: luis.moles@tecnalia.com

**Abstract:** Despite the increasing availability of vast amounts of data, the challenge of acquiring labeled data persists. This issue is particularly serious in supervised learning scenarios, where labeled data are essential for model training. In addition, the rapid growth in data required by cutting-edge technologies such as deep learning makes the task of labeling large datasets impractical. Active learning methods offer a powerful solution by iteratively selecting the most informative unlabeled instances, thereby reducing the amount of labeled data required. However, active learning faces some limitations with imbalanced datasets, where majority class over-representation can bias sample selection. To address this, combining active learning with data augmentation techniques emerges as a promising strategy. Nonetheless, the best way to combine these techniques is not yet clear. Our research addresses this question by analyzing the effectiveness of combining both active learning and data augmentation techniques under different scenarios. Moreover, we focus on improving the generalization capabilities for minority classes, which tend to be overshadowed by the improvement seen in majority classes. For this purpose, we generate synthetic data using multiple data augmentation methods and evaluate the results considering two active learning strategies across three imbalanced datasets. Our study shows that data augmentation enhances prediction accuracy for minority classes, with approaches based on CTGANs obtaining improvements of nearly 50% in some cases. Moreover, we show that combining data augmentation techniques with active learning can reduce the amount of real data required.

**Keywords:** active learning; CTGAN; data augmentation; entropy sampling; machine learning

**MSC:** 68T01

check for updates

**Citation:** Moles, L.; Andres, A.; Echegaray, G.; Boto, F. Exploring Data Augmentation and Active Learning Benefits in Imbalanced Datasets. *Mathematics* **2024**, *12*, 1898. https://doi.org/10.3390/math12121898

Academic Editors: Florin Leon

Received: 20 May 2024
Revised: 9 June 2024
Accepted: 13 June 2024
Published: 19 June 2024

## 1. Introduction

The current era of big data has brought an impressive growth in the amount of data available for analysis. Indeed, the proliferation of open-source databases has democratized the access, greatly increasing the use and efficiency of machine learning and Artificial Intelligence algorithms in different domains. Despite this increase, most available data appear in unlabeled form, limiting their usefulness in different scenarios. This situation is particularly challenging in supervised learning paradigms, where labeled data are essential for model training. Moreover, in different domains, such as industrial processes, the availability of real-world data is often limited, and simulations can be prohibitively time-consuming. In addition, the amount of data needed to train efficient models with state-of-the-art technologies such as deep learning is constantly increasing, making their labeling impractical. Consequently, the ability to construct robust and accurate machine learning models with minimal labeled data has become crucial.

To address these challenges, active learning (AL) methods stand out as promising approaches. Active learning is an iterative process that selects the most informative unlabeled instances to reduce the amount of labeled data required for training supervised machine learning models. By identifying these key instances in each iteration, AL strategies maximize the learning process's efficiency, minimizing the amount of data that needs to be labeled [1]. These strategies have proven to be effective in different domains and applications [2–4]. However, they also exhibit their own limitations, such as being sensitive to noisy labeled data or the variety of existing sampling strategies and the different performance between them [5]. Moreover, a critical limitation of AL methods is their performance with imbalanced datasets, where the over-representation of certain classes can bias the selection of new samples towards the majority class, thereby compromising the model's generalization performance, especially in minority classes [6]. This issue is common in a variety of real-world datasets, particularly in industrial or medical scenarios.

To mitigate the challenges posed by imbalanced data, Oversampling and data augmentation techniques often appear as potential solutions [7,8]. Strategies such as random oversampling [9] or the so-called Synthetic Minority Over-sampling Technique (SMOTE) [10] are popular approaches for generating synthetic data. When dealing with image data, Generative Adversarial Networks (GANs) have become a de facto solution due to their outstanding capacity to generate such data [11]. Consequently, recently, different approaches have emerged, attempting to transfer GANs' ability to capture non-linear relationships between features to the generation of tabular data, with CTGAN (Conditional Tabular Generative Adversarial Network) being one of the most successful approaches [12].

Combining active learning and data augmentation techniques presents a promising avenue for enhancing machine learning model performance, particularly in handling imbalanced datasets. By generating high-quality synthetic data, we can potentially reduce the dependence on large amounts of real data in an active learning framework. Furthermore, balancing the dataset with data augmentation can mitigate the bias towards majority classes, allowing active learning to make more effective and equitable queries. This synergy not only optimizes the selection of new samples but also can potentially enhance the performance in predicting minority classes.

Therefore, our research has been conducted to address three key objectives:

- Analyze and evaluate different data augmentation strategies and their impact on minority classes performance.
- Assess the performance that can be obtained via active learning while minimizing the need for real data, independently of any data augmentation technique.
- Quantify the potential improvement of combining data augmentation strategies with active learning in imbalanced datasets.

Despite the existence of studies exploring the combination of data augmentation and active learning, as far as we know, there are scarce results when there is limited data per class and a significant imbalance ratio between classes. To address this gap, we contribute to the field in three significant ways. First, we analyze the effectiveness of generating synthetic tabular data using multiple data augmentation techniques (random oversampling, CTGAN, and G-SMOTE), focusing on the performance of minority classes. Second, we propose an innovative approach by training independent CTGANs conditioned solely on the data of specific classes, which is apparently novel for datasets with small amounts of data and high imbalance ratios. Third, we extend traditional methodologies by proposing a new framework where the active learning strategy is based solely on real data while the model learns from both real and synthetic data generated iteratively. This framework is evaluated to determine whether generating synthetic data exclusively for the minority class or for all classes except the majority class is more effective in improving model performance. Thus, our research contributions can be summarized as follows:

- Contribution 1 (C1)—We analyze the effectiveness of generating synthetic tabular data using multiple data augmentation techniques on imbalanced datasets, focusing on the performance of minority classes.
- Contribution 2 (C2)—We propose a potentially innovative approach to training and generating data using CTGAN: training independent CTGANs conditioned solely on the data of specific classes.
- Contribution 3 (C3)—We extend traditional methodologies that combine data augmentation and active learning by proposing a framework where the active learning strategy selection is based solely on real data, while the model is trained using both real and synthetic data. More concretely, we focus on generating synthetic data informed by the active learning module after each iteration, investigating the effectiveness of generating synthetic data exclusively for the minority class versus generating data for all classes except the majority class.

This work aims to fill the gaps that were not addressed in the literature and provide a thorough analysis of these techniques in various scenarios, emphasizing the improvement of minority class performance. Our goal is to identify the most critical factors when addressing imbalanced datasets, whether through generating synthetic data, selecting effective active learning strategies to minimize the need for additional data, or combining both approaches.

The remainder of this paper is organized as follows: Section 2 provides a review of related work, highlighting existing approaches and their limitations. Section 3 offers a background on the algorithms and methodologies employed in our study. Section 4 delineates the experimental methodology adopted to investigate the research questions. Section 5 presents the experimental results and discusses their implications. Finally, in Section 6, we draw conclusions and outline avenues for future research.

## 2. Related Work

In this section, we will first review diverse contributions reported in the past where data augmentation strategies have been used to solve various problems (Section 2.1). The section continues by analyzing contributions in the field of active learning (Section 2.2), and concludes by reviewing a collection of contributions where previous disciplines are mixed (Section 2.3). As mentioned in Section 1, a wide range of different strategies exists to solve the issues presented in this research, so contributions discussed in the following section will focus on the specific strategies used in our work.

### *2.1. Data Augmentation*

Data augmentation arises with the purpose of training machine learning models in data scarcity scenarios. Data scarcity can appear due to different reasons, such as noisy or bad-quality data, or incredibly time-consuming data collection processes in some scenarios. Sometimes, despite having a large amount of data, it appears in imbalanced proportions, making it necessary to increase data of minority classes to train a robust machine learning model [13]. In order to solve these problems, different works using multiple strategies have been proposed. Simple oversampling techniques, such as random oversampling, aim to tackle this issue by randomly sampling with replacement from the available samples to generate new ones in the minority classes [9].

The Synthetic Minority Over-sampling Technique (SMOTE) was designed to generate new instances from minority classes by interpolating the instances that are close to each other [10]. As such, it has been used in different works, such as in a real-time traffic and weather data based on crash prediction, by generating crash events that tend to be under-represented [14], and to address an imbalanced problem in the context of a five-year survival prognosis prediction, obtaining significant improvements in both sensitivity and specificity [15]. Different modifications of SMOTE have been proposed in order to improve the quality of the generated data. Among them, Geometric SMOTE (G-SMOTE) generates the samples within a geometry instead of within a segment [16]. This algorithm has been used in [17] to improve the prediction of Alzheimer's and Parkinson's diseases for highly class-imbalanced data, showing a 10% increase in the accuracy of the classifier when using G-SMOTE as the oversampling algorithm compared to SMOTE, and a 30% increase compared to not using any oversampling strategies.

On the other hand, as introduced in Section 1, generative models have become very popular due to their capacity to generate realistic synthetic tabular data in multiple contexts. MedGAN [18] combines an auto-encoder with a GAN to generate tabular data, and TableGAN [19] uses Convolutional Neural Networks to capture the correlations between features. Among all the strategies, CTGAN (Conditional Tabular Generative Adversarial Network) achieves state-of-the-art results in the generation of synthetic tabular data [12]. Unlike traditional oversampling techniques, CTGAN operates in a conditional setting, allowing for the generation of realistic samples tailored to specific class distributions and feature dependencies. Thanks to this feature, it has been successfully adopted to improve the detection of IoT Botnet attacks when a high imbalance was present [20]. It has also been used to generate fake data that match the distribution of real data in disk failure prediction imbalance datasets, achieving great results when testing the augmentation in different classic machine learning models (e.g., BP-ANN, SVM, Decision Tree, Random Forest) [21]. In addition, it has shown improvement when used to generate data to estimate costs in Green Building projects [22], minimizing the root mean square error of the predictions. Lastly, it has been adopted in smart-grids to generate additional data to forecast the load with more precision [23].

### 2.2. Active Learning

Active learning is a machine learning strategy that aims to train a model with the least amount of data necessary by iteratively selecting the most informative instances for the model. This strategy is especially useful when the amount of unlabeled data is large but the cost of labeling it is expensive [6]. Active learning has been widely used in various scenarios. In agriculture, a dissimilarity-based active learning method that considers data diversity has been used to select few but representative samples in a crop weed classification system [24], with results showing improving accuracies with reduced data usage. In industrial scenarios, a cost-sensitive active learning method based on bidirectional gated neural networks and a maximum expected cost reduction sampling strategy has been proposed for fault diagnosis with uncertainty in dynamic environments [25], showing better performance in both binary and multiclass fault diagnosis. An active learning framework introducing Convolutional Neural Networks to build an image classifier via a limited amount of labeled training instances has also been proposed [26]. In the study, instead of only focusing on the uncertain samples of low prediction confidence, they also selected the high-confidence samples from the unlabeled set for feature learning, and promising results were achieved on two challenging image classification datasets.

Experiments attempting to solve the problem of active learning in imbalanced datasets have also been done. A solution based on the Extreme Learning Machine classification model has been proposed for this purpose [27]. A reinforcement online active learning ensemble for drifting imbalanced data streams is studied in [28], while [29] shows a method for multiclass imbalanced data streams with concept drift. Finally, a method for tackling imbalanced data that consists of balancing exploitation and exploration in active learning selection strategies has been proposed in [30,31], which presents an active learner that adapts to imbalanced and balanced data without using prior knowledge.

### 2.3. Active Learning + Data Augmentation

Finally, a collection of works trying to solve the problem of active learning with imbalanced datasets by using different data augmentation strategies have also been proposed. For example, Conditional Generative Adversarial Networks (CGANs) have been used to generate realistic chest X-ray images with different disease characteristics by conditioning their generation on a real image sample [32]. Those images were then used as training data in an active learning scenario where a Bayesian neural network identified informative samples to add to the training set. Flip augmentation and Mixup augmentation methods have also been mixed in an active learning scenario with entropy-based query strategy sampling to improve image classification performance [33]. A GAN-based active learning method that seeks to generate high entropy samplings has been proposed in [34], while [35] presented a pool-based active learning algorithm that learns an active learning sampling mechanism in an adversarial manner. They used a variational autoencoder (VAE) and an adversarial network trained to discriminate between unlabeled and labeled data. In this proposal, the VAE tries to trick the adversarial network into predicting that all data points are from the labeled pool, while the adversarial network learns how to discriminate between dissimilarities in the latent space.

A human-in-the-loop approach, combining a CTGAN for data augmentation and an active learning module for addressing data bottlenecks in medical deep learning models, has been proposed in [36]. The effectiveness of artificial data in active learning scenarios has also been studied in [37], by using G-SMOTE as an artificial data generator and introducing it into the traditional active learning framework in order to reduce the amount of labeled data required in active learning. In the conducted experiments, random selection is used as a baseline, while entropy and breaking ties query strategies are compared. Results have shown reduced cost and time requirements for a successful AL implementation in all of the datasets used. The introduction of a hyperparameter optimization component to improve the generation of artificial instances during the AL process has been also tested in [38]. A modified version of G-SMOTE has been used to generate artificial data, and different query strategies have been used (random selection, entropy, and breaking ties). The method has been tested across different datasets, improving the performance of traditional active learning, both in terms of classification performance and data selection efficiency.

### *2.4. Our Contribution*

In this study, we present several contributions that extend beyond the current state of the art in data augmentation and active learning for imbalanced datasets. While previous studies have focused on individual data augmentation techniques or a limited set of methods, our work offers a thorough comparison of three distinct data augmentation strategies: random oversampling, G-SMOTE, and CTGAN. Moreover, we focus on the effect of data augmentation strategies on the performance of minority classes. Secondly, we investigate a method for generating data with CTGAN that consists of training independent CTGANs using only the existing data from that class. Thirdly, we study the integration of data augmentation in an active learning scenario where the synthetic data are generated based on the query strategy selection, and are solely used to train the model along with real data. Lastly, building upon the foundational work of integrating data augmentation within AL processes, we introduce and evaluate two different approaches. We investigate the impact of iteratively augmenting data only from the minority class, as well as from all classes except the majority class.

## **3. Background**

This section is devoted to explaining the different technical concepts surrounding the research conducted in this paper.

### *3.1. Data Augmentation*

Data augmentation aims to provide solutions for scenarios dealing with imbalanced datasets or scarcity of real data. Class imbalance occurs when one class (the minority class) is significantly underrepresented compared to the others (the majority class or classes). This imbalance can lead to biased models that perform poorly in predicting the minority class. To address this issue, various strategies exist, ranging from interpolation techniques to generative network-based models. The following sections explain the three different methods compared in this research.

#### 3.1.1. Random Oversampling

In random oversampling, new instances are synthetically generated for the minority class by randomly duplicating existing instances from that class until a more balanced distribution is achieved between the minority and majority classes. This process involves randomly selecting instances from the minority class and adding them to the training dataset, potentially multiple times, until the desired balance is reached.

While random oversampling can help alleviate class imbalance and improve the performance of models, it also has potential drawbacks. For example, it may lead to overfitting, where the model learns noise present in the duplicated instances. Additionally, if the minority class is very small, random oversampling may not be effective and could even result in the majority class being underrepresented in the training data.

#### 3.1.2. Synthetic Minority Over-Sampling Technique, SMOTE

SMOTE [10] and its variant G-SMOTE [16] are techniques used in machine learning to address class imbalance in datasets. SMOTE aims to tackle class imbalance by generating synthetic samples for the minority class. Unlike simple duplication in random oversampling, SMOTE creates synthetic instances by selecting a minority class instance and finding its k-nearest minority class neighbors. It then creates synthetic instances along the line segments joining these neighbors. This process continues until the desired balance between the minority and majority classes is achieved.

More specifically, the SMOTE algorithm calculates the difference between a sample and its nearest neighbor. It multiplies this difference by a random number between 0 and 1 and adds it to the previous sample, resulting in the generation of a new random point along the line segment between two specific samples.

In order to address some of its limitations, such as the generation of noisy samples or the difficulty in capturing the underlying distribution of the data in some scenarios, the G-SMOTE algorithm is proposed. In this variant, instead of generating synthetic instances along the line segments between nearest neighbors, G-SMOTE generates them in a geometric region of the input space around each selected minority class instance, aiming to generate synthetic instances that are more representative of the underlying data distribution.

#### 3.1.3. Conditional Tabular Generative Adversarial Network, CTGAN

CTGANs [12] are a type of generative model specifically designed to generate synthetic tabular data while preserving their statistical properties. Tabular data refer to structured data organized in rows and columns. Each row represents an instance or observation, while columns represent features or attributes of those instances. CTGANs are based on the concept of GANs [39], which are composed of two neural networks (a generator G and a discriminator D) that are trained in an adversarial manner. The generator aims to produce synthetic samples that are indistinguishable from real data, while the discriminator aims to correctly classify whether a given sample is real or synthetic. In other words, they play a minimax game with the following objective function [39].

$$\min_G \max_D V(D,G) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log D(\boldsymbol{x})] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(\boldsymbol{z})))] \tag{1}$$

where $\mathbf{x}$ represents real data samples, and $\mathbf{z}$ is a noise vector sampled from a prior distribution $p_{\mathbf{z}}$, typically a Gaussian distribution.

The goal of the discriminator is to maximize the objective function by maximizing both $D(x)$ and $1 - D(G(z))$, while the generator tries to minimize both terms.

CTGAN extends this framework to handle tabular data by introducing conditional inputs, allowing the model to generate data conditioned on specific feature values. The

CTGAN objective function modifies the GAN minimax game to include conditional information $\mathbf{c}$:

$$\min_{G}\max_{D}\mathbb{E}_{\mathbf{x},\mathbf{c}\sim p_{\text{data}}(\mathbf{x},\mathbf{c})}[\log D(\mathbf{x},\mathbf{c})]+\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z}),\mathbf{c}\sim p(\mathbf{c})}[\log(1-D(G(\mathbf{z},\mathbf{c}),\mathbf{c}))] \tag{2}$$

The innovations of CTGAN to handle tabular data include the following:

- Conditional Vectors: The conditional vector $\mathbf{c}$ is concatenated with the noise vector $\mathbf{z}$ to form the input to the generator.
- Mode-specific Normalization: CTGAN introduced a mode-specific normalization to handle continuous features in tabular data with multimodal distributions, where probability distributions present different modes. They make use of a Variational Gaussian Mixture model to represent each value in a continuous feature as a one-hot vector indicating the mode, and a scalar value representing the normalized value according to the mode.
- Conditional Generator: CTGANs are able to handle some of the challenges presented by imbalanced categories, which usually make the generative networks collapse.

### 3.2. Active Learning, AL

AL is a machine learning strategy that tries to handle labeled data scarcity in supervised learning scenarios by selecting the most informative data points to be labeled by a human annotator. The main assumption in these approaches is that, if a learning algorithm can choose the data it wants to learn from, it can achieve the same performance as traditional learning methods using much less labeled data [1].

In other words, the goal of active learning is to find a hypothesis $h \in \mathcal{H}$ ($\mathcal{H}$ being the hypothesis space, i.e., the set of all possible models) that minimizes the expected error on the entire input space $\mathcal{X}$:

$$h^* = \arg\min_{h\in\mathcal{H}}\mathbb{E}_{(x,y)\sim P_{\text{data}}}[L(h(x),y)]$$

where $L$ is the loss function, and $P_{\text{data}}$ is the underlying data distribution.

There are two main aspects that differentiate active learning algorithms. The first one is the way they process unlabeled data, where we can distinguish between three different active learning settings [6].

- Membership Query Synthesis: In this scenario, the learner generates synthetic instances and requests labels for them. This can be useful in finite problem domains, and no unlabeled pool is necessary in this scenario. On the other hand, it can generate samples that are not possible to label by a human annotator.
- Stream-Based Selective Sampling: In this scenario, it is assumed that obtaining an unlabeled instance is inexpensive. Based on this assumption, each unlabeled instance is analyzed (one at a time) and the model decides whether to query the instance's label or reject it based on its information. To determine the amount of information of the instance, different query strategies can be used.
- Pool-based Active Learning: This is the most well-known scenario and the one chosen in this research. Here, it is normally assumed to have a small set of labeled data $\mathcal{L} = \{(x_1,y_1),(x_2,y_2),\ldots,(x_m,y_m)\}$, where each $x_i \in \mathcal{X}$ and $y_i \in \mathcal{Y}$, and a large collection of unlabeled data $\mathcal{U} = \{x_1,x_2,\ldots,x_n\}$, where each $x_i \in \mathcal{X}$. An active learner is trained with the initial labeled set, and, based on a query strategy, it measures the informativeness of the unlabeled points, selecting the most uncertain points and requesting a label from a human annotator. The newly labeled instance $(x^*,y^*)$ is added to the labeled dataset $\mathcal{L}$, and the model is retrained:

$$\mathcal{L} \leftarrow \mathcal{L} \cup \{(x^*,y^*)\}$$

This process is repeated iteratively until a certain criterion is met (performance, label budget, etc.).

The second aspect is the strategy they use for querying a new instance. Based on the criterion used to select a new instance, we can group them into two groups:

- Information-Based Methods: These methods focus on selecting the most informative points, which are typically those close to the decision boundaries of the model. These strategies take the uncertainty of the instances into account to choose the next data to label. Inside this group, several approaches can be followed to measure the uncertainty of the instances. In this research, we have used an uncertainty strategy based on the entropy of the samples. This method computes the entropy of the unlabeled points, and selects the one with the highest entropy for labeling, as it is considered to be the most informative point.

$$\mathbf{x}^* = \arg\max_{\mathbf{x} \in \mathcal{D}_U} \left( -\sum_i P_h(y_i|\mathbf{x}) \log P_h(y_i|\mathbf{x}) \right) \tag{3}$$

- Representation-Based Methods: These methods select new points based on the representativeness of the unlabeled data. These strategies focus on exploring the input space instead of exploiting the points near the decision boundaries. Different strategies such as cluster-based (selecting nearest neighbors to the clusters' centroid) or diversity-based (labeling the most diverse unlabeled point compared to the labeled ones) approaches fall inside this group.

## 4. Methodology

As initially stated in Section 1, the ultimate objective of this research is to analyze what actually matters when addressing a supervised classification problem with imbalanced data. There exist multiple approaches that rely on generating synthetic data to promote a more diverse and representative set of the minority classes, although their performance varies depending on the dataset. Furthermore, active learning can be used to select the strategy that minimizes the amount of real data needed. Towards assessing the impact of these types of approaches in the context of imbalanced datasets, we outline the following research questions (RQs):

- RQ1: How much could data augmentation techniques improve the performance of a machine learning model in imbalanced datasets?
  The hypothesis underlying this question is that data augmentation techniques might effectively address class imbalance by generating more data in classes that are weakly represented, thereby improving the model's generalization and performance in minority classes.
- RQ2: How can data augmentation strategies reduce the number of required samples needed in active learning to achieve a given performance in imbalanced datasets? Is it coherent to combine data augmentation with active learning?
  By using synthetic data to enrich and balance the dataset, active learning gains access to a more diverse and representative dataset for sample selection. This expanded sample space enables active learning to make more informed decisions when acquiring new labeled instances, potentially enhancing its ability to identify and address class imbalance effectively and reducing the overall amount of real data.
- RQ3: Can the iterative generation of synthetic data after each active learning query improve model's decision boundaries?
  By continually refining the dataset with newly acquired real and synthetic instances, the model is exposed to a broader range of patterns, potentially leading to more accurate and robust decision boundaries. Thus, an active learning query strategy could benefit from improved decision boundaries, selecting more informative samples that could lead to a bigger reduction in real data needed.

To answer these questions, we propose a specific scenario for each RQ. Next, we describe each of them in more detail.

### 4.1. Scenario 1: Data Augmentation

This scenario aims to assess the usefulness of data augmentation techniques when having an imbalanced dataset (i.e., RQ1). To this end, we employed multiple data augmentation techniques to generate synthetic data in those classes that were actually imbalanced. Following the augmentation, we trained a Random Forest model with both the real and the augmented data, and we compared results against the model trained with a non-augmented dataset, that is, solely with real data (Figure 1). Regarding the training procedure of the Random Forest model, it was iterative. This means that we randomly selected new data to be used for the training in each iteration. This allowed a future comparison when using active learning.

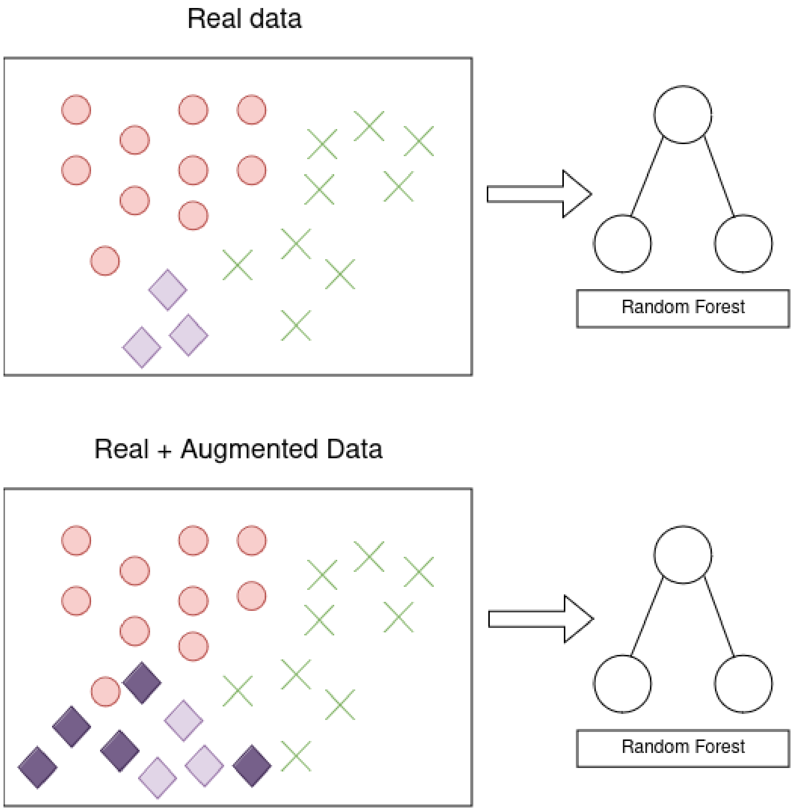

**Figure 1.** Scenario 1 illustration. (**Top**) Random Forest solely trained with real data. (**Bottom**) Real + augmented data are used to train the Random Forest. Augmented data considers any data generated by either random oversampling, G-SMOTE, or CTGAN.

### 4.2. Scenario 2: Active Learning

In this scenario, the impact of active learning was evaluated (i.e., RQ2). The pipeline for assessing its impact is outlined in Figure 2. First, active learning was performed on the original dataset (without augmented data). Next, the same process was conducted using the different augmented datasets, allowing the active learning strategy to query not only real data, but also the datasets composed of both real and synthetic data. The goal was to measure the effect of synthetic data on active learning strategies. An entropy query strategy was used to decide which data to add to the training set in each iteration.

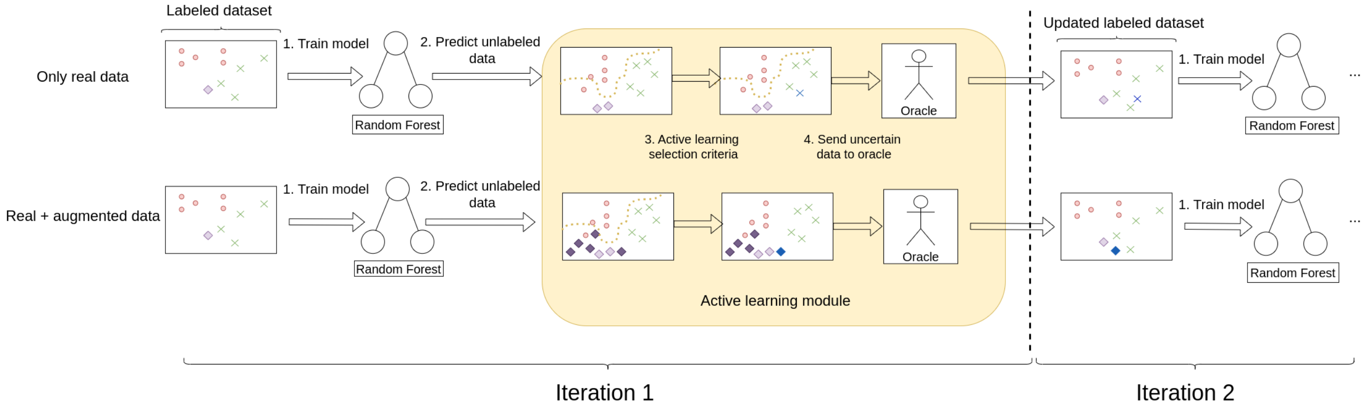

**Figure 2.** Scenario 2 illustration. Active learning is analyzed considering only the real data (**top**) and various datasets built with real + augmented data (**bottom**). The latter option considers datasets generated with either random oversampling, G-SMOTE, or CTGAN.

### *4.3. Scenario 3: Iterative Synthetic Data Generation*

The final proposed scenario focused on the iterative generation of synthetic data and its potential benefits in active learning scenarios (i.e., RQ3). The pipeline is illustrated in Figure 3. The training begins with a typical active learning setup using the original (non-augmented) dataset. After each iteration of active learning, where the model queries a new unlabeled instance and adds it to the labeled dataset, a synthetically generated data point is also added to the labeled set. Both real and synthetic data are then used to retrain our Random Forest model.

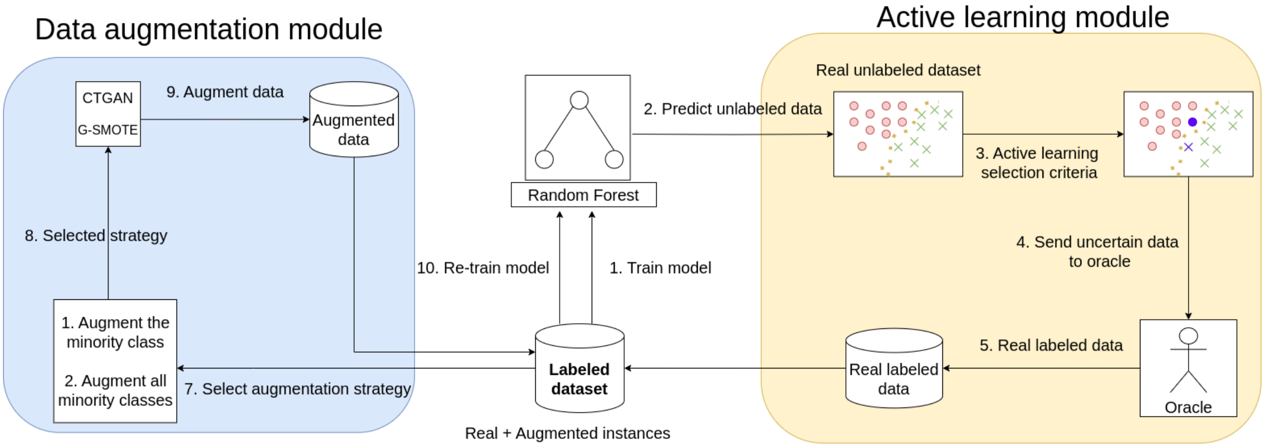

**Figure 3.** Scenario 3 pipeline. Active learning only selects the next data based on real data. However, its decision boundaries are affected by both the real and synthetic data, which are constructed iteratively.

In this setup, we maintain two datasets: one containing the remaining unlabeled data from which active learning selects the most informative next real data, and another pool comprising real labeled data and synthetically generated data. The latter is used to improve decision boundaries, which, in turn, guide the selection of the next real data.

Synthetic data were generated using either CTGAN or G-SMOTE, with two different strategies for generating extra points:

- `Minority Class Sampling`: This strategy involves generating one synthetic instance belonging to the minority class of the labeled set. As training progresses, the minority class in the labeled set may change; thus, this approach focuses on augmenting the current minority class in each iteration.
- `All-Minority-Class Sampling`: This strategy generates one instance for each of the minority classes, adding these to the dataset in every iteration.

The pseudocode of the proposed pipeline is presented below (Algorithm 1):

**Algorithm 1** Iterative data augmentation with active learning

```
Require: Labeled dataset $\mathcal{L}$, Unlabeled dataset $\mathcal{U}$, Budget $B$, Augmentation method $\mathcal{A}$
Ensure: Trained model $M$
Initialize $\mathcal{L} \leftarrow$ Initial labeled set
Initialize $\mathcal{U} \leftarrow$ Initial unlabeled set
while stopping criterion not met do
    Train model $M$ on $\mathcal{L}$
    Compute uncertainty scores for all $x \in \mathcal{U}$
    Select top $B$ most uncertain samples $\mathcal{Q} \subseteq \mathcal{U}$
    Obtain labels for $\mathcal{Q}$ and update $\mathcal{L} \leftarrow \mathcal{L} \cup \mathcal{Q}$
    Apply augmentation method $\mathcal{A}$ on $\mathcal{L}$ to generate synthetic samples $\mathcal{S}$
    Update $\mathcal{L} \leftarrow \mathcal{L} \cup \mathcal{S}$
    Update $\mathcal{U} \leftarrow \mathcal{U} \setminus \mathcal{Q}$
end while
return Trained model $M$
```

## 5. Experimentation

### *5.1. Datasets*

The experiments described in the different scenarios were conducted on three distinct classification datasets: `Wine`, `Baseball`, and `Steel Plates`. These datasets were selected due to their high degree of class imbalance, with imbalance ratios of 68.1%, 20.98%, and 12.24%, respectively, which is essential for assessing the effectiveness of data augmentation and active learning techniques in addressing imbalanced data challenges.

As shown in Table 1, the `Wine` dataset contains six classes, the `Baseball` dataset contains three classes and the `Steel Plates` contains seven classes. In all datasets, one class constitutes a large proportion of the overall data, overshadowing the remaining classes in the three datasets. Consequently, the model's learning could be biased towards the majority class, leading to poorer predictions for the minority classes.

**Table 1.** Summary of the `Wine`, `Baseball`, and `Steel Plates` datasets. "Inst" refers to the number of instances, "Features" refers to the number of features for each dataset, "Classes: Instances (Proportion %)" indicates the different classes, the instances for each of them, and the proportion of the real data that is represented, and "Imbalance Ratio" is calculated as the ratio of majority to minority class instances. Majority Instance and Minority Instance.

| Dataset | Inst | Features | Classes: Instances (Proportion %) | Imbalance Ratio |
|---|---|---|---|---|
| Wine | 1599 | 11 | 0:10 (0.62%)<br>1: 53 (3.31%)<br>2: 681 (42.59%)<br>3: 638 (39.9%)<br>4: 199 (12.45%)<br>5: 18 (1.13%) | 68.1 |
| Baseball | 1320 | 15 | 0:1196 (90.6%)<br>1: 57 (4.31%)<br>2: 67 (5.07%) | 20.98 |
| Steel Plates | 1941 | 24 | 0:158 (8.14%)<br>1: 190 (9.78%)<br>2: 391 (20.14%)<br>3: 72 (3.7%)<br>4: 55 (2.8%)<br>5: 402 (20.71%)<br>6: 673 (34.67%) | 12.24 |

The datasets were each divided into two subsets: 80% for training and 20% for testing. From each training subset, a certain amount of data were initially selected as labeled (fifteen from `Wine`, six from `Baseball`, fourteen from `Steel Plates`), while the remaining instances were treated as unlabeled data for the active learning process. This setup was consistently applied across all experiments.

### 5.2. Evaluation Metrics

Considering the imbalanced ratio of the datasets, and our goal of improving the performance of minority classes, metrics that were not sensitive to class imbalance were selected to test our experiments. In this context, we considered the following:

- F-Score: It is the harmonic mean of precision and recall, and reaches its best value at 1 (perfect precision and recall) and worst at 0:

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

  where precision is the number of true positive predictions divided by the total number of positive predictions (true positives plus false positives), and recall is the number of true positive predictions divided by the total number of actual positive instances (true positives plus false negatives).
- Recall: Recall is defined as the ratio of correctly predicted positive observations to the total actual positives. Specifically, we used it to measure the performance of each strategy towards each class individually, in order to test the correct predicted instances in minority classes.

  The recall for each class $i$ in a multiclass classification problem can be calculated as follows:

$$\text{Recall}_i = \frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i} \tag{5}$$

  where $\text{TP}_i$ (true positives for class $i$) is the instances correctly predicted as class $i$ and $\text{FN}_i$ (false negatives for class $i$) is the instances of class $i$ that were incorrectly predicted as some other class.
- Balanced Accuracy: It is defined as the average of the recall obtained on each class, and it provides a balanced view of the model's performance. As a consequence, it is well suited for imbalanced datasets as it accounts for the performance across all classes equally, mitigating the bias towards the majority class. This metric is calculated as follows:

$$\text{Balanced Accuracy} = \frac{1}{K} \sum_{i=1}^{K} \frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i} \tag{6}$$

  where $K$ is the number of classes, $\text{TP}_i$ is the number of true positives for class $i$, and $\text{FN}_i$ is the number of false negatives for class $i$.

### 5.3. Data Augmentation Strategies

As previously introduced in Section 3.1, our experimental setup considered three data augmentation techniques: random oversampling, G-SMOTE, and CTGAN.

To generate synthetic data and balance the number of samples in each class, we employed a straightforward approach. This involved generating enough synthetic samples in each minority class to match the number of samples in the majority class. For instance, in the `Wine` dataset, the majority class has 681 samples, while the minority class has 10 samples. Therefore, we generated $681 - 10 = 671$ synthetic samples for the minority class, ensuring that all classes were balanced before the training began. This balancing method was applied in both Scenario 1 and Scenario 2, where synthetic data generation was completed before the training phase began (It should be noted that, in Scenario 3, the data generation process was iterative and integrated with the training phase).

Regarding CTGAN, we adopted the implementation of the Synthetic Data Vault (SDV) project [40]. We considered training the CTGAN using two different approaches:

- `Naive`: Training a single CTGAN on all available data to learn the overall dataset distribution and generate data for all classes.
- `Per class`: Training separate CTGANs for each class, so that each model learns the distribution of a specific class independently.

For CTGAN's hyperparameters, we conducted a grid search over the set of the most sensitive parameters, as shown in Table 2. The configurations that yielded the best results are highlighted in bold. These optimal configurations were used in our multiple experiments. Plots of some of the trained CTGANs regarding the evolution of loss functions of the generator and the discriminator over the training epochs, and some statistical information about the generated data over the Wine dataset can be seen in Appendix A.

**Table 2.** Grid search of CTGAN hyperparameters. The Gen/Discr. Dim refer to the number of fully connected layers and the amount of neurons in each layer. The learning rate value for both the generator and discriminator was fixed at 2 × 10 $^{-4}$. Selected hyperparameters are highlighted in **bold**.

| Hyperparam | Wine Values | Baseball Values | Steel Plates Values |
|---|---|---|---|
| Epochs | 1000, **12,000**, 20,000, 50,000 | **2500**, 5000, 20,000, 50,000 | **20,000**, 30,000 |
| Batch Size | 100, 300, 500, **1000** | 100, **300**, 500, 1000 | 300, 500, **1000** |
| Gen. Dim | (256, 256), **(256, 256, 256)** | (256, 256) **(256, 256, 256)** | (256, 256), (256, 256, 256)<br>(256, 256, 256, 128)<br>(256, 256, 256, 256)<br>**(256, 256, 256, 256, 256)** |
| Discr. Dim | **(256, 256)**, (256, 256, 256) | **(256, 256)**, (256, 256, 256) | (256, 256), (256, 256, 256)<br>**(256, 256, 256, 128)**<br>(256, 256, 256, 256)<br>(256, 256, 256, 256, 256) |
| Discr. steps | **1**, 2 | **1**, 2 | **1**,2 |

### 5.4. Active Learning Setup

As mentioned before in Section 3.2, different active learning settings and query strategies exist in the literature. Among all possibilities, in this paper, we adopted the configuration shown in Table 3. We chose a pool-based strategy due to the popularity of this approach. Since many different query strategies exist and the selection is not straightforward, we decided to use entropy sampling, as it is a well-known strategy used in typical active learning scenarios [36,41].

**Table 3.** Configuration of active learning setup.

| Configuration | Values |
|---|---|
| Instances in Initial labeled data | 15 for Wine, 6 for Baseball,<br>14 for Steel Plates |
| Active learning scenario | Pool-based sampling |
| Active learner | Random Forest |
| Batch size (queried instances in each AL iteration) | 1 |
| Query strategy | Entropy sampling |
| Stopping criteria | All instances labeled |

## 6. Results and Discussion

This section presents the results addressing each of the research questions stated above. For clarity, the results are organized into different subsections, each corresponding to a research question and displaying the findings for each of the three datasets.

### 6.1. Scenario 1: Data Augmentation

Scenario 1 compared the performance of different data augmentation strategies across three different datasets. The performance of the models was analyzed focusing on the effectiveness of these strategies in improving predictions for minority classes.

Figure 4 shows the evolution in the performance of the different strategies in terms of F-score every time a new sample is labeled. The shaded area in our graphs represents the standard deviation, calculated based on experiments conducted with three different random seeds. This provides a visual indication of the variability of our results. For the Wine dataset, the baseline (no augmentation) approach performs the best in terms of F1-score, with random oversampling showing better performance among CTGAN-based approaches and G-SMOTE. Similarly, in the Baseball dataset, the baseline approach yields the highest performance, followed by random oversampling. The high baseline performance (0.95) means that the model is capable of predicting most instances accurately without the need for augmentation. In the case of the Steel Plates dataset, random oversampling presents slightly better results than the baseline approach, and training one CTGAN per class also matches the baseline's performance. Overall, all approaches perform similarly in terms of F1-score for the Steel Plates dataset.

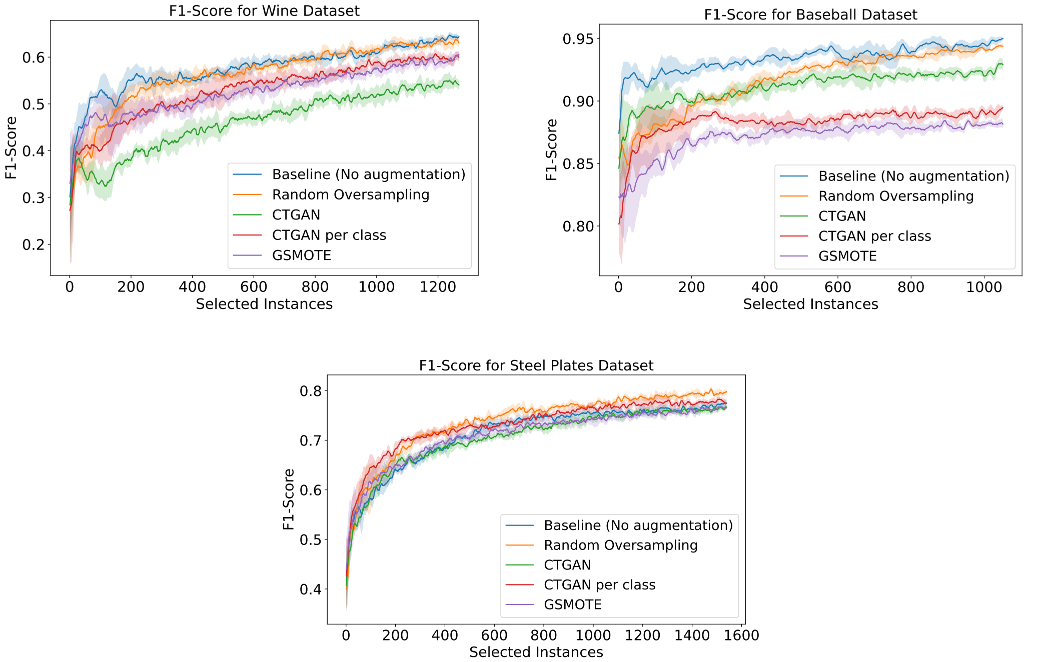

**Figure 4.** F1-score for Wine (**left**), Baseball (**right**), and Steel Plates (**bottom**) datasets with various data augmentation strategies.

Although the F1-score is a useful metric for evaluating the performance of models in imbalanced datasets, it sometimes can be influenced by the performance in majority classes, especially in highly imbalanced datasets. Therefore, with our purpose being the improvement of minority classes, balanced accuracy was also analyzed (see Figure 5), as it ensures that the performance on each class is given equal importance.

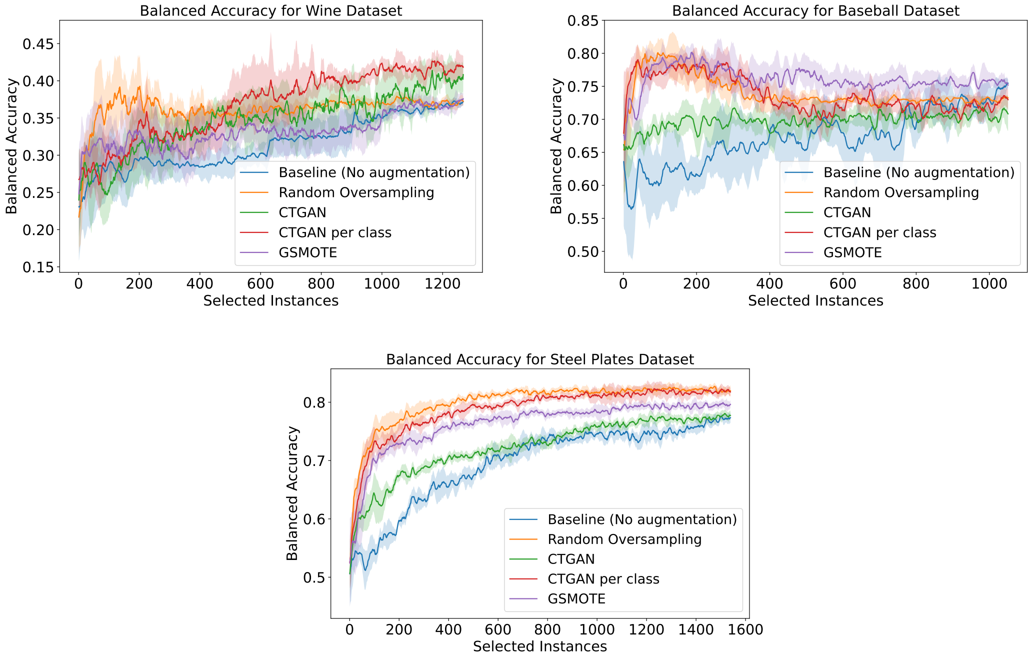

**Figure 5.** Balanced accuracy for Wine (**left**), Baseball (**right**), and Steel Plates (**bottom**) datasets with various augmentation strategies.

In the Wine dataset, CTGAN-based approaches present the best performance, with the CTGAN per class approach being the most effective, achieving a balanced accuracy of 0.43 (see Table 4). All augmentation techniques outperform the baseline non-augmented data on the Wine dataset. In the Baseball dataset, random oversampling and GSMOTE achieve the best results, both with values of 0.8. The CTGAN per class approach also shows better performance compared to the non-augmented data. For the Steel Plates dataset, the CTGAN per class approach and random oversampling achieve the best results (balanced accuracy of 0.82), with G-SMOTE also outperforming the baseline.

**Table 4.** Max F1-score and balanced accuracy for each data augmentation technique. Best results are highlighted in **bold**.

| Dataset | | Baseline | Random Oversampling | CTGAN | CTGAN Per Class | GSMOTE |
|---|---|---|---|---|---|---|
| Wine | F1-Score | **0.65** | 0.64 | 0.55 | 0.61 | 0.61 |
| | Balanced | 0.37 | 0.39 | 0.41 | **0.43** | 0.38 |
| Baseball | F1-Score | **0.95** | 0.94 | 0.93 | 0.89 | 0.88 |
| | Balanced | 0.76 | **0.8** | 0.71 | 0.79 | **0.8** |
| Steel Plates | F1-Score | 0.78 | **0.8** | 0.76 | 0.78 | 0.77 |
| | Balanced | 0.78 | **0.82** | 0.78 | **0.82** | 0.8 |

To specifically evaluate the improvement in minority classes, we analyzed the recall per class for each augmentation technique, as shown in Table 5. This analysis was conducted to understand the impact of different augmentation strategies on individual class performance, providing insights into their effectiveness in addressing class imbalance. Detailed performance graphs of recall per class for each dataset are provided in Appendix B.

**Table 5.** Percentage of recall per class for different augmentation techniques in maximum balanced accuracy iteration. Best results for each class are highlighted in **bold**.

| Dataset | Class | Baseline | Random Oversampling | CTGAN | CTGAN Per Class | GSMOTE |
|---|---|---|---|---|---|---|
| Wine | 0 | 0.00% | 21.21% | **48.48%** | 3.03% | 0.00% |
| | 1 | 0.00% | 16.80% | 17.91% | **33.33%** | 11.11% |
| | 2 | **76.09%** | 53.63% | 48.17% | 63.57% | 64.13% |
| | 3 | **70.86%** | 44.58% | 43.99% | 50.85% | 57.34% |
| | 4 | 43.48% | 66.06% | 56.52% | **71.97%** | 59.58% |
| | 5 | **33.33%** | **33.33%** | **33.33%** | **33.33%** | **33.33%** |
| Baseball | 0 | **98.70%** | 86.15% | 94.92% | 82.73% | 84.11% |
| | 1 | 82.41% | 82.07% | 61.62% | **85.61%** | 77.53% |
| | 2 | 45.30% | 72.03% | 58.97% | 68.76% | **78.79%** |
| Steel Plates | **0** | 62.41% | 82.01% | 66.95% | **82.95%** | 73.20% |
| | 1 | 74.16% | 84.05% | 79.27% | **87.00%** | 81.90% |
| | 2 | 94.95% | 94.91% | 95.53% | **96.31%** | 94.91% |
| | 3 | **92.86%** | **92.86%** | **92.86%** | **92.86%** | **92.86%** |
| | 4 | 76.31% | 79.06% | 71.90% | **81.27%** | 78.79% |
| | 5 | 66.67% | **74.86%** | 70.67% | 71.90% | 71.31% |
| | 6 | **75.91%** | 71.43% | 69.27% | 63.91% | 66.62% |

Analyzing the values from Table 5, we observe how in the Wine dataset, classes with a larger number of instances (Classes 2 and 3) are best predicted by the baseline approach. However, for classes with fewer instances (Classes 0, 1, 4, and 5), most augmentation strategies outperform the baseline recall, with the exception of Class 5 where all strategies perform similarly. Specifically, the CTGAN per class strategy presents the best results in predicting Classes 1 and 4, incrementing the recall by 33.33% and 28.49%, respectively, while CTGAN performs best for Class 0 with a recall of 48.48%. In the Baseball dataset, we can see a similar behavior. The majority class (Class 0) is best predicted by the baseline approach. However, for the minority classes (Classes 1 and 2), CTGAN per class and GSMOTE improve the results compared to the baseline. This suggests that augmentation strategies are particularly effective for classes with fewer instances. The Steel Plates dataset further reinforces this trend. CTGAN per class outperforms all other strategies for Classes 0 to 4, while random oversampling performs best in predicting Class 5. Once again, the majority class is best predicted by the baseline approach.

After analyzing the results on specific classes, we can confirm that the global F1-score, as shown in earlier graphs, was influenced by the performance of majority classes, leading to an apparent decrease with augmentation strategies. However, a deeper analysis of balanced accuracy and individual recall per class revealed that most techniques improved performance for minority classes, in exchange for a slight decrease in majority class recall. We hypothesize that maintaining the imbalance ratio of the dataset, rather than balancing it to the majority class, could better balance these trade-offs. This highlights the importance of focusing on minority class performance in imbalanced datasets and suggests that augmentation strategies can provide significant benefits in these scenarios. Overall, although the performance of different augmentation strategies varies over the specific case, the CTGAN per class approach consistently shows the best results.

Table 6 illustrates the percentage of real data usage required to achieve the maximum performance of the baseline approach for different augmentation strategies. Specifically, the baseline values represent the percentage of real data needed by the baseline to achieve the best performance in both the F1-score and balanced accuracy. The subsequent columns indicate the percentage of real data needed by the augmentation techniques to match the baseline performance. If an augmentation technique does not reach the baseline performance, it is denoted by a dash (“-”).

**Table 6.** Percentage of real data needed by each data augmentation technique to reach the max score shown by the baseline method. "-" means that the method does not reach the performance of the baseline approach. Best results are highlighted in **bold**.

| Dataset | | Baseline | Random Oversampling | CTGAN | CTGAN Per Class | GSMOTE |
|---|---|---|---|---|---|---|
| Wine | F1-Score | **97.73%** | - | - | - | - |
| | Balanced | 97.65% | **8.99%** | 54.57% | 40.5% | 88.58% |
| Baseball | F1-Score | **100%** | - | - | - | - |
| | Balanced | 100% | 3.41% | - | **1.98%** | 5.87% |
| Steel Plates | F1-Score | 99.55% | **66.1%** | - | 75.64% | - |
| | Balanced | 97.6% | **15.27%** | 79.25% | 24.16% | 39.76% |

Based on the results, and in consequence with the values presented in Table 4, no augmentation strategy is able to outperform the baseline F1-score in the Wine and Baseball datasets. However, in the Steel Plates dataset, random oversampling is capable of reducing the amount of real data needed to reach the baseline performance by 33.45%. On the other hand, nearly all strategies manage to achieve the baseline performance in balanced accuracy while using less real data (with the exception of CTGAN in the Baseball dataset). Among these strategies, random oversampling stands out for its significant reduction in data requirement in both the Wine and Steel Plates datasets, while CTGAN achieves an impressive reduction of 98% in the Baseball dataset.

### 6.2. Scenario 2: Active Learning

Results of this scenario analyzed the effect of data augmentation approaches in an active learning framework, and tried to determine if data augmentation helps reduce the number of real samples needed to achieve a certain performance in active learning scenarios.

Graphs of the F1-scores in Scenario 2 (Figure 6) present similar patterns to the ones in Scenario 1. For the three datasets, both the baseline and random oversampling techniques achieve the highest F1-score (see Table 7 for F1-score values). On the Wine dataset, the CTGAN per class approach and GSMOTE achieve similar results as both the baseline and random oversampling strategies, while, on the Steel Plates dataset, the CTGAN approach performs as well as both strategies.

**Table 7.** Max F1-score and max balanced accuracy for each augmentation technique in an active learning scenario. Best results are highlighted in **bold**.

| Dataset | | Baseline (No Augm.) | Random Oversampling | CTGAN | CTGAN Per Class | GSMOTE |
|---|---|---|---|---|---|---|
| Wine | F1-Score | **0.65** | **0.65** | 0.56 | 0.62 | 0.61 |
| | Balanced | 0.38 | 0.38 | 0.42 | **0.44** | 0.39 |
| Baseball | F1-Score | **0.95** | **0.95** | 0.93 | 0.91 | 0.89 |
| | Balanced | 0.76 | 0.76 | 0.74 | 0.76 | **0.78** |
| Steel Plates | F1-Score | 0.78 | **0.8** | 0.77 | 0.79 | 0.78 |
| | Balanced | 0.78 | 0.83 | 0.79 | **0.84** | 0.81 |

Focusing on the balanced accuracy graphs in Figure 7, and the values in Table 7, we observe how the CTGAN per class approach performs the best on the Wine dataset, reaching a value of 0.44, whereas baseline and random oversampling both scored 0.38. GSMOTE achieved the best value (0.78) on the Baseball dataset, slightly outperforming the other methods. Finally, on the Steel Plates dataset, the CTGAN per class approach again showed the best performance of 0.84, followed by the random oversampling strategy. Overall, results in Table 7 show that, while a simple augmentation strategy such as random oversampling presents good results in terms of the F1-score, more sophisticated methods like CTGAN per class and GSMOTE tend to perform better in terms of balanced accuracy, indicating their effectiveness in managing class imbalances.

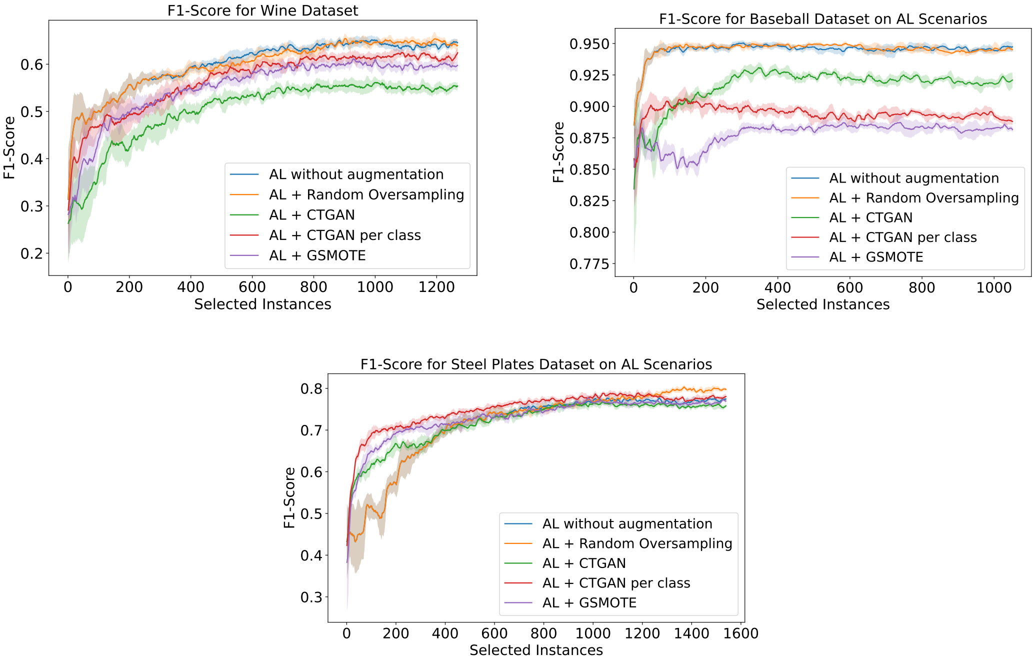

**Figure 6.** F1-score for Wine (**left**), Baseball (**right**), and Steel Plates (**bottom**) datasets with various augmentation strategies on active learning entropy sampling scenario.

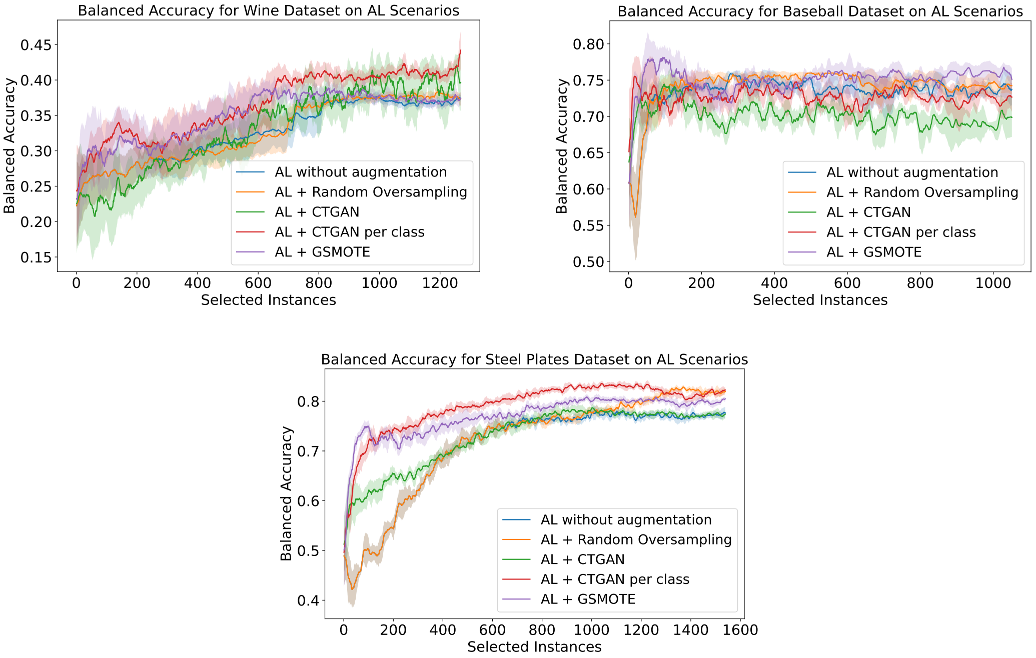

**Figure 7.** Balanced accuracy for Wine (**left**), Baseball (**right**) and Steel Plates (**bottom**) datasets with various augmentation strategies on active learning entropy sampling scenario.

Because the improvement of minority class predictions is our main interest, recall values for each class at the point of maximum balanced accuracy are presented in Table 8. While the detailed performance graphs of recall per class are not included here, they are provided in Appendix C.

**Table 8.** Percentage of recall per class for different augmentation techniques in maximum balanced accuracy iteration on active learning scenario. Best results are highlighted in **bold**.

| Dataset | Class | Baseline | Random Oversampling | CTGAN | CTGAN Per Class | GSMOTE |
|---|---|---|---|---|---|---|
| Wine | 0 | 0.00% | 0.00% | **45.45%** | 8.33% | 0.00% |
| | 1 | 0.00% | 0.55% | 21.21% | **29.80%** | 20.11% |
| | 2 | **76.89%** | 75.58% | 50.98% | 65.73% | 65.82% |
| | 3 | **70.74%** | 69.84% | 44.58% | 56.29% | 53.36% |
| | 4 | 45.00% | 50.98% | 57.12% | **71.81%** | 60.68% |
| | 5 | **33.33%** | **33.33%** | **33.33%** | **33.33%** | **33.33%** |
| Baseball | 0 | 98.43% | **98.54%** | 89.70% | 86.26% | 82.77% |
| | 1 | 83.33% | 83.33% | 67.42% | **87.88%** | 83.84% |
| | 2 | 45.92% | 46.15% | 66.20% | 52.45% | **67.83%** |
| Steel Plates | 0 | 63.26% | 80.11% | 70.64% | **82.58%** | 72.73% |
| | 1 | 79.98% | 83.81% | 78.79% | **94.74%** | 83.49% |
| | 2 | 94.83% | 94.99% | 94.99% | **95.84%** | 94.87% |
| | 3 | 92.86% | 92.86% | 92.86% | 92.86% | 92.86% |
| | 4 | 75.21% | 81.82% | 75.48% | 80.99% | **81.54%** |
| | 5 | 66.67% | **73.40%** | 68.87% | 73.33% | 72.65% |
| | 6 | **74.39%** | 72.99% | 69.23% | 64.58% | 68.08% |

Results for the Wine dataset show that GANs and GSMOTE augmentation techniques decrease the recall of the classes with more instances (Classes 2 and 3), with the baseline and random oversampling being the ones achieving higher recall rates of 76.89% and 70.74% in those classes, respectively. However, they present significant improvements in performance for the minority classes. CTGAN and CTGAN per class show notable improvements in recall for Classes 0, 1, and 4. For instance, CTGAN achieves a recall of 45.45% for Class 0, while all other strategies present recalls close to 0%. Similarly, the CTGAN per class approach performs best for Class 1, obtaining a recall of 29.80%, compared to the baseline's 0% and random oversampling's 0.55%, and also outperforms other methods for Class 4 with a recall of 71.81%. With respect to Class 5, all methods achieve the same recall of 33.33%.

In the Baseball dataset, the random oversampling and baseline approaches, again, achieve the highest recall for the majority class (Class 0), with values of 98.54% and 98.43%, respectively. For Class 1, CTGAN per class stands out with a recall of 87.88%, surpassing baseline and random oversampling, both at 83.33%. GSMOTE performs best for minority Class 2, achieving a recall of 67.83%, slightly higher than CTGAN's 66.20%. Finally, on the Steel Plates dataset, CTGAN per class shows the highest recall for Classes 0, 1, and 2, with recall rates of 82.58%, 94.74%, and 95.84%, respectively. For Class 3, all methods achieve the same recall rate of 92.86%. GSMOTE achieves the highest recall for Class 4 with 81.54%. Finally, for classes with majority of instances (5 and 6), random oversampling (73.4% in Class 5) and baseline (74.39%) once again present the best results.

Recall results over individual classes demonstrate that the effectiveness of augmentation techniques varies across different classes and datasets. CTGAN and CTGAN per class often show significant improvements in recall for minority class instances, particularly in the Wine and Steel Plates datasets, while methods like baseline and random oversampling normally achieve the highest recall rates in predicting majority classes, but without being able to increase the performance of minority classes.

Table 9 shows the percentage of real data usage required to achieve the maximum performance of the baseline approach for different augmentation strategies on active learning scenarios, with the aim of determining whether data augmentation can help active learning (AL) reduce the amount of real data needed.

**Table 9.** Percentage of real data needed by each data augmentation technique to reach the max score shown by the baseline method on an active learning scenario. "-" means that the method does not reach the performance of the baseline approach. Best results are highlighted in **bold**.

| Dataset | | Baseline | Random Oversampling | CTGAN | CTGAN Per Class | GSMOTE |
|---|---|---|---|---|---|---|
| Wine | F1-Score | 78.19% | **71.15%** | - | - | - |
| | Balanced | 78.19% | 71.07% | 60.52% | 51.29% | **46.29%** |
| Baseball | F1-Score | **29.4%** | - | - | - | - |
| | Balanced | **26.99%** | 46.78% | - | - | 44.5% |
| Steel Plates | F1-Score | 82.54% | 70.3% | - | **59.4%** | 63.92% |
| | Balanced | 66.62% | 66.11% | 57.8% | **27.4%** | 48% |

Results show that data augmentation does not help active learning use less real data on the Baseball dataset. The baseline without augmentation achieves the desired performance with less data in terms of both F1-score and balanced accuracy. This suggests that, for this particular dataset, active learning alone is sufficient. In the Wine dataset, random oversampling is able to use less real data to achieve the baseline performance in terms of F1-score. Similarly, for the Steel Plates dataset, all augmentation techniques, except for CTGAN, outperform the baseline in terms of data efficiency, with CTGAN per class being the most effective. This indicates how data augmentation can effectively enhance the efficiency of active learning on these datasets. When considering balanced accuracy, all data augmentation strategies outperform the baseline in terms of the amount of real data used for the Wine and Steel Plates datasets. In the Wine dataset, GSMOTE is the most effective, while in the Steel Plates dataset, CTGAN per class stands out as the best technique. After analyzing the results, we can conclude that, while the benefits of mixing data augmentation with active learning may depend on the data structure, as seen with the Baseball dataset, it is able to enhance the efficiency of active learning by reducing the amount of real data needed, such as with the Wine and Steel Plates datasets.

### *6.3. Scenario 3: Iterative Synthetic Data Generation*

This scenario studied the effect of iterative data augmentation in active learning scenarios. After each active learning iteration, new generated synthetic data were introduced in the labeled dataset. Two different strategies were followed to generate the data: generating synthetic data for the minority class or generating one instance for all minority classes. Figure 8 presents the F1-score performance. Wine dataset results are displayed in the top row, Baseball dataset results in the middle row, and the Steel Plates dataset results in the bottom row. For each dataset, the left column shows the F1-scores for the minority class approach, while the right column presents the F1-scores for the all-minority-class approach.

The graphs show a really similar performance of all data augmentation techniques in terms of F1-score in the Wine and Steel Plates datasets. However, in the Baseball dataset, the CTGAN approach does not achieve the same performance level as the other techniques. Furthermore, when comparing the two augmentation strategies (augmenting only the minority class versus augmenting all minority classes) the performance is very similar in both of them across all datasets. This suggests that both approaches are equally effective in enhancing the F1-score, regardless of the specific dataset.

This pattern is confirmed in balanced accuracy graphs (Figure 9), except for a slight improvement of CTGAN per class in the Wine dataset. Given the similar results observed between the two augmentation strategies (see Table 10), and for clarity purposes in the body of the manuscript, we will present the results of specific recall per classes, focusing only on the minority class approach. Results from the all-minority-class approach will be included in Appendix D for reference. Other analysis regarding real data usage will be detailed for both approaches.

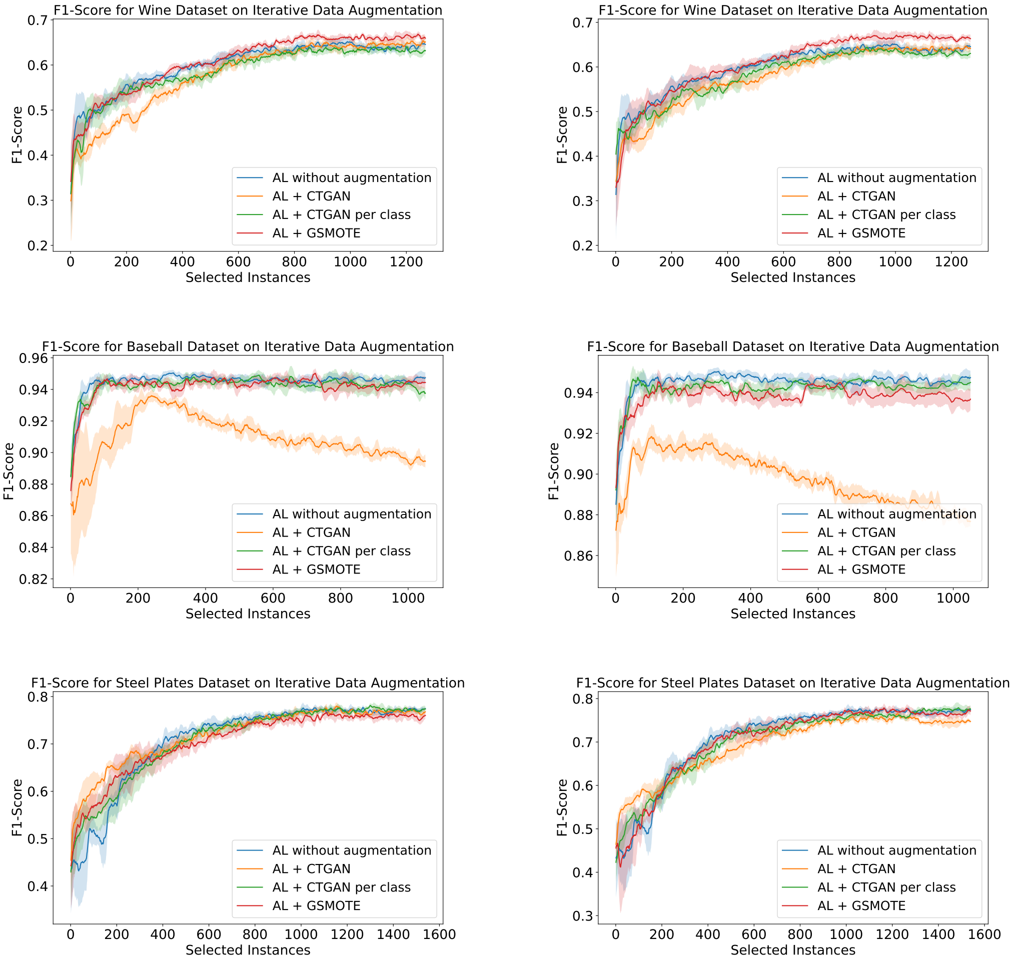

**Figure 8.** F1-score for Wine (**top**), Baseball (**middle**), and Steel Plates (**bottom**) datasets with iterative data augmentation and active learning. Data augmentation: minority class (**left**), and all minority classes (**right**).

**Table 10.** Max F1-score and max balanced accuracy performance of different augmentation techniques performed in an iterative manner. Best results are highlighted in **bold**.

| **Dataset** | | | **Baseline** | **CTGAN** | **CTGAN Per Class** | **GSMOTE** |
|---|---|---|---|---|---|---|
| Wine | Minority class | F1 | 0.65 | 0.65 | 0.64 | **0.67** |
| | | Balanced | 0.38 | **0.42** | **0.42** | 0.39 |
| | All Min class | F1 | **0.65** | **0.65** | 0.64 | **0.65** |
| | | Balanced | 0.38 | 0.43 | **0.45** | 0.4 |
| Baseball | Minority class | F1 | **0.95** | 0.94 | **0.95** | **0.95** |
| | | Balanced | 0.76 | **0.78** | 0.74 | 0.75 |
| | All Min class | F1 | **0.95** | 0.92 | **0.95** | **0.95** |
| | | Balanced | 0.76 | **0.79** | 0.74 | 0.73 |
| Steel Plates | Minority class | F1 | **0.78** | **0.78** | **0.78** | 0.77 |
| | | Balanced | 0.78 | **0.79** | 0.78 | 0.77 |
| | All Min class | F1 | **0.78** | 0.77 | **0.78** | **0.78** |
| | | Balanced | **0.78** | 0.77 | 0.77 | 0.77 |

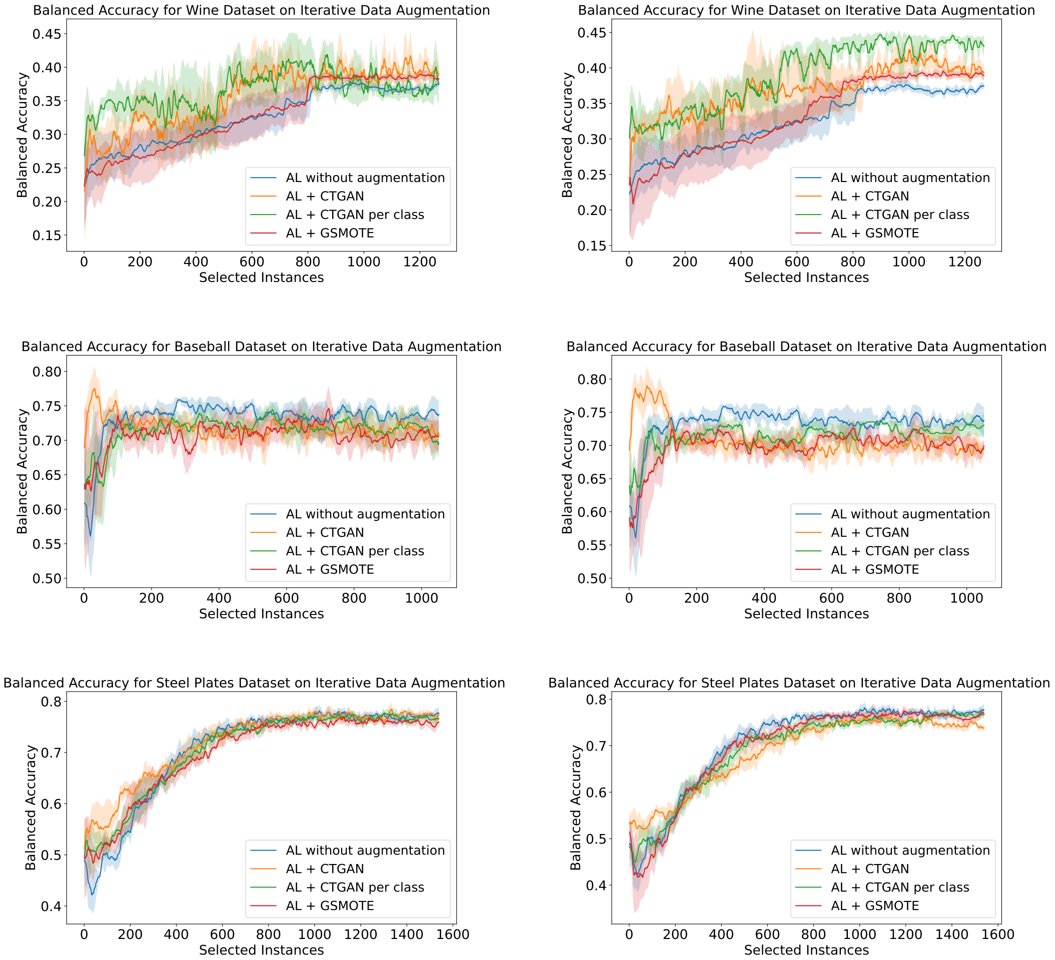

**Figure 9.** Balanced Accuracy for Wine (**top**), Baseball (**middle**), and Steel Plates (**bottom**) datasets with iterative data augmentation and active learning. Data augmentation: minority class (**left**), and all minority classes (**right**).

Table 10 shows the maximum values of the F1-score and balanced accuracy on iterative data augmentation across the different datasets. For the Wine dataset, GSMOTE achieves the best F1-score (0.67) when using the minority class approach. In terms of balanced accuracy, CTGAN and CTGAN per class both outperform the baseline, achieving a score of 0.42. The all-minority-class approach shows similar behavior, with CTGAN per class presenting the best results in balanced accuracy. In the Baseball dataset, the baseline method achieves the same F1-score than CTGAN per class and GSMOTE in both the minority class and all-minority-class approaches. In balanced accuracy, CTGAN outperforms the other methods in both the minority class and all-minority-class approaches. For the Steel Plates dataset, all methods, again, perform similarly in both strategies, with a best F1-score of 0.78 and a best balanced accuracy of 0.79. Specific recall results for each classes are presented in Table 11.

Recall results of iterative data augmentation in specific classes present some interesting insights. In the Wine dataset, CTGAN per class shows the best result in predicting minority Class 0, while the CTGAN approach improves the baseline prediction for minority Classes 1 and 4 by 7.71% and 11.88%, respectively. All strategies have the same results in predicting Class 5, while, just as in previous Scenarios, the baseline achieves the best result in predicting majority Class 2. However, unlike in previous scenarios, GSMOTE is able to slightly outperform the prediction of the other majority class (Class 3). This pattern is also presented in the Baseball dataset, where, again, it is able to perform marginally better than the baseline in predicting majority Class 0. On the Steel Plates dataset, majority Class 6 is best predicted by the baseline approach, while CTGAN- and CTGAN-based approaches have the best results in predicting minority Classes 4 and 5, respectively. The results show that the effectiveness of iterative data augmentation varies over different classes and datasets. This suggests that the best specific approach depends on the data structure and distribution. Trying to find some patterns, CTGAN-based approaches seem to present a more balanced performance across different classes, and GSMOTE seems to reduce the loss of majority classes often generated by augmentation techniques. Apart from recall results, Table 12 presents the percentage of real data used by the strategies to achieve baseline performance.

**Table 11.** Percentage of recall per class for iterative data augmentation techniques in maximum balanced accuracy iteration on active learning scenario. (minority class approach). Best results are highlighted in **bold**.

| Dataset | Class | Baseline | CTGAN | CTGAN Per Class | GSMOTE |
|---|---|---|---|---|---|
| Wine | 0 | 0.00% | 19.70% | **39.39%** | 0.00% |
| | 1 | 0.00% | **7.71%** | 0.00% | 0.00% |
| | 2 | **76.89%** | 70.41% | 73.93% | 76.38% |
| | 3 | 70.74% | 61.27% | 67.19% | **72.96%** |
| | 4 | 45.00% | **57.88%** | 37.58% | 51.36% |
| | 5 | **33.33%** | **33.33%** | **33.33%** | **33.33%** |
| Baseball | 0 | 98.43% | 86.43% | 98.86% | **98.94%** |
| | 1 | 83.33% | **90.91%** | 82.83% | 82.07% |
| | 2 | **45.92%** | 55.24% | 41.72% | 42.89% |
| Steel Plates | 0 | 63.26% | **71.12%** | 64.11% | 61.74% |
| | 1 | **79.98%** | 74.40% | 75.04% | 76.48% |
| | 2 | **94.83%** | 94.79% | 94.44% | **94.83%** |
| | 3 | **92.86%** | **92.86%** | **92.86%** | **92.86%** |
| | 4 | 75.21% | **76.86%** | 72.45% | 74.10% |
| | 5 | 66.67% | 65.25% | **67.86%** | 65.73% |
| | 6 | **74.39%** | 74.32% | 77.17% | 73.02% |

**Table 12.** Percentage of real data needed by each iterative data augmentation technique to reach the max score shown by the baseline method on an active learning scenario. "-" means that the method does not reach the performance of the baseline approach. Best results are highlighted in **bold**.

| Dataset | | | Baseline | CTGAN | CTGAN Per Class | GSMOTE |
|---|---|---|---|---|---|---|
| Wine | Minority class | F1 | 78.18% | 96% | - | **58.17%** |
| | | Balanced | 78.18% | 42.5% | **40.7%** | 63.7% |
| | All Min class | F1 | 78.18% | - | - | **54.4%** |
| | | Balanced | 78.18% | 34% | **33.6%** | 60.7% |
| Baseball | Minority class | F1 | **29.36%** | - | - | - |
| | | Balanced | 27% | **2.6%** | - | - |
| | All Min class | F1 | **29.36%** | - | - | - |
| | | Balanced | 27% | **1.42%** | - | - |
| Steel Plates | Minority class | F1 | 82.5% | **74.9%** | 84.2% | - |
| | | Balanced | **66.6%** | 86.6% | - | - |
| | All Min class | F1 | 82.5% | - | 86.9% | **75.8%** |
| | | Balanced | **66.6%** | - | - | - |

Data efficiency results for the Wine dataset show that iterative data augmentation with GSMOTE reduces the real data needed to achieve the baselines F1-score, both in the minority and all-minority-class cases, while CTGAN per class performs the same way in balanced accuracy. For the Baseball dataset, while any approach is able to perform better than the baseline in terms of F-score, the CTGAN approach reaches the baseline performance in balanced accuracy using 24.6% less real data on the minority class approach, and 25.5% on the all-minority-class approach. Finally, CTGAN reduces the data needed in the minority class approach to reach the baseline F1-score, and GSMOTE does the same in the all-minority-class approach. These results indicate that data augmentation techniques applied in an iterative manner, particularly CTGAN and GSMOTE, can significantly reduce the amount of real data needed to achieve baseline performance in certain scenarios.

## 7. Conclusions and Future Work

Our investigation aimed to study the benefits of active learning and data augmentation in imbalanced datasets, focusing on potential improvements in predictions for minority classes and the reduction of real data needed. To achieve this, we utilized three different data augmentation techniques and evaluated their effectiveness across three distinct scenarios: without active learning, with active learning, and iteratively augmenting data within active learning. Three datasets, each presenting a different ratio of imbalance, were used to ensure a comprehensive comparison. Our findings indicate that data augmentation can effectively improve performance in minority classes, as demonstrated in the Scenario 1 experiments. However, this improvement often comes at the cost of a reduction in predictions for the majority classes. Among all the strategies tested, considering various metrics such as balanced accuracy and recall per class, CTGAN per class showed consistent results across most datasets, indicating its robustness in handling imbalanced data. Specifically, it has shown improvements in the recall of the classes 0–4 on the Steel Plates dataset, with increases of 20% in some cases. In Scenario 2, our results demonstrate that while data augmentation does not always effectively combine with active learning to enhance efficiency in terms of real data usage, such as in the Baseball dataset, its integration may produce different benefits. Specifically, in the Wine and Steel Plates datasets, the combination of both data augmentation and active learning improved upon the baseline performance without augmentation, the CTGAN per class approach being the one obtaining greater reductions of even 39% in data efficiency. This suggests that, under certain conditions, integrating data augmentation with active learning can yield significant benefits in reducing the amount of real data required while maintaining or improving predictive performance. Finally, iterative data augmentation tested in Scenario 3 also showed improvements in predicting minority classes and real data usage. This could imply that the iterative integration of synthetic data into the training process has a positive effect in the decisions made by the model, thus improving the quality of the active learning query strategy. Specifically, both CTGAN-based approaches obtained better results in general, achieving improvements of even 40% in some classes.

Finally, building upon the current study, we recognize several avenues for future research that could further enhance our understanding and effectiveness of data augmentation and active learning in imbalanced datasets. During this research, three different data augmentation strategies have been used to analyze the different scenarios proposed. To enrich the comparison and provide a more comprehensive analysis, future work will involve the incorporation of additional data augmentation techniques. Exploring methods such as Variational Autoencoders (VAEs) or other advanced GAN variants (e.g., StyleGAN, or CycleGAN) could offer new insights into their impact on model performance and generalization. With regards to the machine learning model, Random Forest has been used in this experimentation, and, while it has proven to be effective, evaluating the performance of other machine learning algorithms could be beneficial for our study. Thus, future research will include experiments with algorithms such as Gradient Boosting Machines (e.g., XGBoost, or LightGBM), Support Vector Machines (SVMs) and neural network architectures. This will help to identify the most suitable algorithms for handling imbalanced datasets under different conditions. Experimenting with a wider range of active learning frameworks, including query-by-committee, and expected model change strategies will be another clear research line. Finally, to further generalize our findings, it could be interesting to extend the comparison to a broader range of data types, including non-tabular data such as medical images, or even regression imbalanced problems.

**Author Contributions:** Conceptualization, L.M., A.A. and G.E.; Methodology, L.M., A.A., G.E. and F.B.; Software, L.M.; Validation, L.M.; Formal analysis, L.M. and A.A.; Investigation, L.M.; Writing—original draft, L.M.; Writing—review & editing, L.M., A.A., G.E. and F.B.; Supervision, A.A., G.E. and F.B. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was financed by the Basque Government through their Elkartek program (SONETO project, ref. KK-2023/00038).

**Data Availability Statement:** The code will be make available upon acceptance in the following repository: https://github.com/luismoles/imbalanced_data_augm_al (accessed on 12 June 2024).

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A. CTGAN Training

In this appendix, we provide various plots illustrating the evolution of CTGAN losses over the training epochs across the three datasets. Each figure contains multiple images showing the performance with various hyperparameter settings, being the top left image in each figure the one representing the chosen one for our experiments. Additionally, we present statistical information about the generated data produced by the selected CTGAN model for the Wine dataset.

Wine

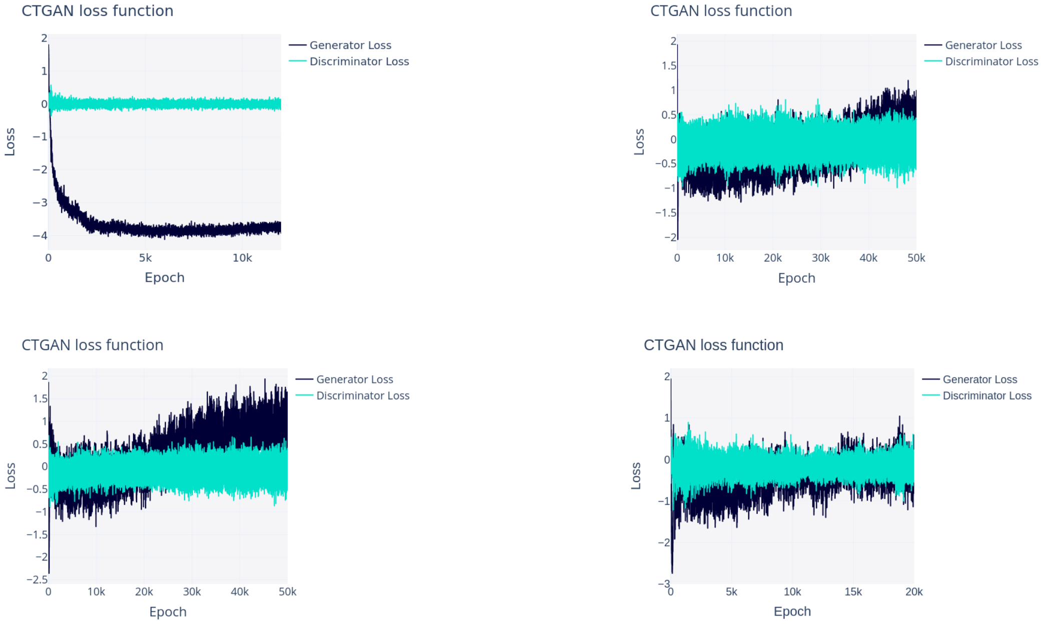

**Figure A1.** Loss functions of trained CTGAN on Wine dataset.

**Table A1.** Statistical comparison of real and generated data by CTGAN on Wine dataset (Part 1).

| Metric | Dataset | Fixed_Acidity | Volatile_Acidity | Citric_Acid |
|---|---|---|---|---|
| Count | Synthetic Data | 1991 | 1991 | 1991 |
| | Real Data | 1279 | 1279 | 1279 |
| Mean | Synthetic Data | 7.93 | 0.60 | 0.23 |
| | Real Data | 8.31 | 0.53 | 0.27 |
| Std | Synthetic Data | 1.46 | 0.27 | 0.20 |
| | Real Data | 1.70 | 0.18 | 0.19 |
| Min | Synthetic Data | 4.60 | 0.15 | 0.00 |
| | Real Data | 4.60 | 0.12 | 0.00 |
| 25% | Synthetic Data | 6.90 | 0.38 | 0.05 |
| | Real Data | 7.10 | 0.40 | 0.10 |
| 50% | Synthetic Data | 7.70 | 0.57 | 0.20 |
| | Real Data | 7.90 | 0.52 | 0.26 |
| 75% | Synthetic Data | 8.70 | 0.75 | 0.40 |
| | Real Data | 9.30 | 0.64 | 0.43 |
| Max | Synthetic Data | 14.10 | 1.35 | 0.79 |
| | Real Data | 15.90 | 1.58 | 1.00 |

**Table A2.** Statistical comparison of real and generated data by CTGAN on Wine dataset (Part 2).

| Metric | Dataset | Residual_Sugar | Chlorides | Free_Sulfur_Dioxide |
|---|---|---|---|---|
| Count | Synthetic Data | 1991 | 1991 | 1991 |
| | Real Data | 1279 | 1279 | 1279 |
| Mean | Synthetic Data | 2.55 | 0.09 | 10.85 |
| | Real Data | 2.54 | 0.09 | 16.13 |
| Std | Synthetic Data | 1.38 | 0.05 | 8.54 |
| | Real Data | 1.38 | 0.05 | 10.47 |
| Min | Synthetic Data | 1.02 | 0.03 | 1.00 |
| | Real Data | 0.90 | 0.01 | 1.00 |
| 25% | Synthetic Data | 1.86 | 0.07 | 4.60 |
| | Real Data | 1.90 | 0.07 | 7.00 |
| 50% | Synthetic Data | 2.17 | 0.08 | 7.50 |
| | Real Data | 2.20 | 0.08 | 14.00 |
| 75% | Synthetic Data | 2.62 | 0.09 | 14.90 |
| | Real Data | 2.60 | 0.09 | 22.00 |
| Max | Synthetic Data | 15.40 | 0.56 | 48.20 |
| | Real Data | 15.40 | 0.61 | 68.00 |

**Table A3.** Statistical comparison of real and generated data by CTGAN on Wine dataset (Part 3).

| Metric | Dataset | Total_Sulfur_Dioxide | Density | pH | Sulphates | Alcohol | Class |
|---|---|---|---|---|---|---|---|
| Count | Synthetic Data | 1991 | 1991 | 1991 | 1991 | 1991 | 1991 |
| | Real Data | 1279 | 1279 | 1279 | 1279 | 1279 | 1279 |
| Mean | Synthetic Data | 29.38 | 0.996 | 3.37 | 0.69 | 11.22 | 5.41 |
| | Real Data | 47.28 | 0.997 | 3.31 | 0.66 | 10.42 | 5.64 |
| Std | Synthetic Data | 22.91 | 0.002 | 0.15 | 0.20 | 1.42 | 2.09 |
| | Real Data | 33.33 | 0.002 | 0.15 | 0.18 | 1.07 | 0.81 |
| Min | Synthetic Data | 6.00 | 0.991 | 2.85 | 0.37 | 8.67 | 3.00 |
| | Real Data | 6.00 | 0.990 | 2.74 | 0.37 | 8.40 | 3.00 |
| 25% | Synthetic Data | 12.60 | 0.994 | 3.26 | 0.55 | 9.98 | 3.00 |
| | Real Data | 22.00 | 0.996 | 3.21 | 0.55 | 9.50 | 5.00 |
| 50% | Synthetic Data | 20.90 | 0.996 | 3.36 | 0.64 | 11.05 | 4.00 |
| | Real Data | 39.00 | 0.997 | 3.31 | 0.62 | 10.20 | 6.00 |
| 75% | Synthetic Data | 41.55 | 0.997 | 3.48 | 0.81 | 12.27 | 8.00 |
| | Real Data | 64.00 | 0.998 | 3.40 | 0.74 | 11.10 | 6.00 |
| Max | Synthetic Data | 131.60 | 1.001 | 3.85 | 2.00 | 14.90 | 8.00 |
| | Real Data | 289.00 | 1.004 | 3.90 | 2.00 | 14.90 | 8.00 |

Baseball

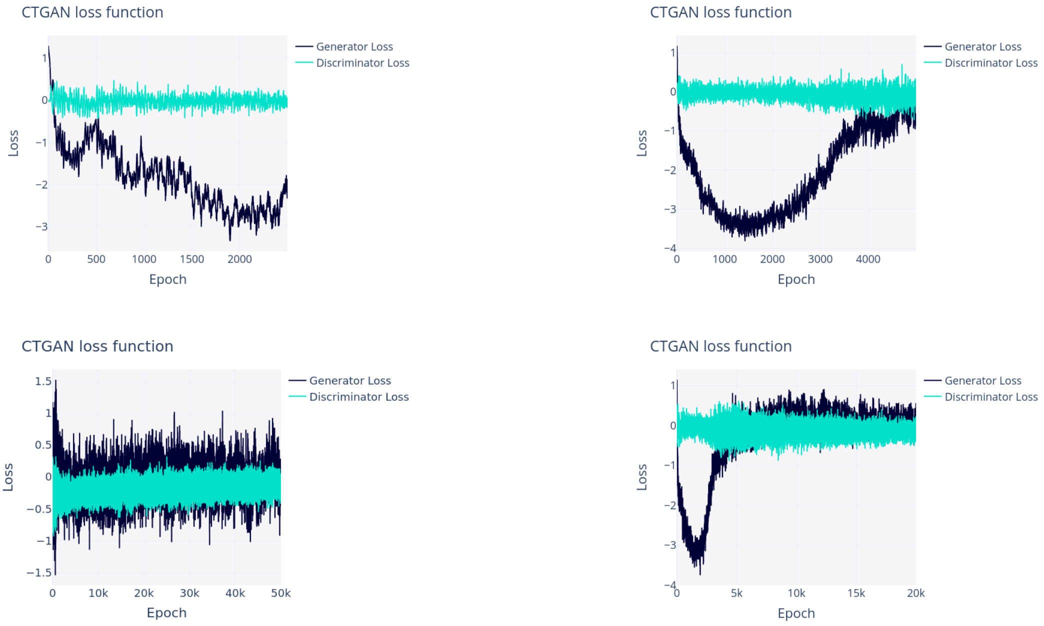

**Figure A2.** Loss functions of trained CTGAN on Baseball dataset.

Steel Plates:

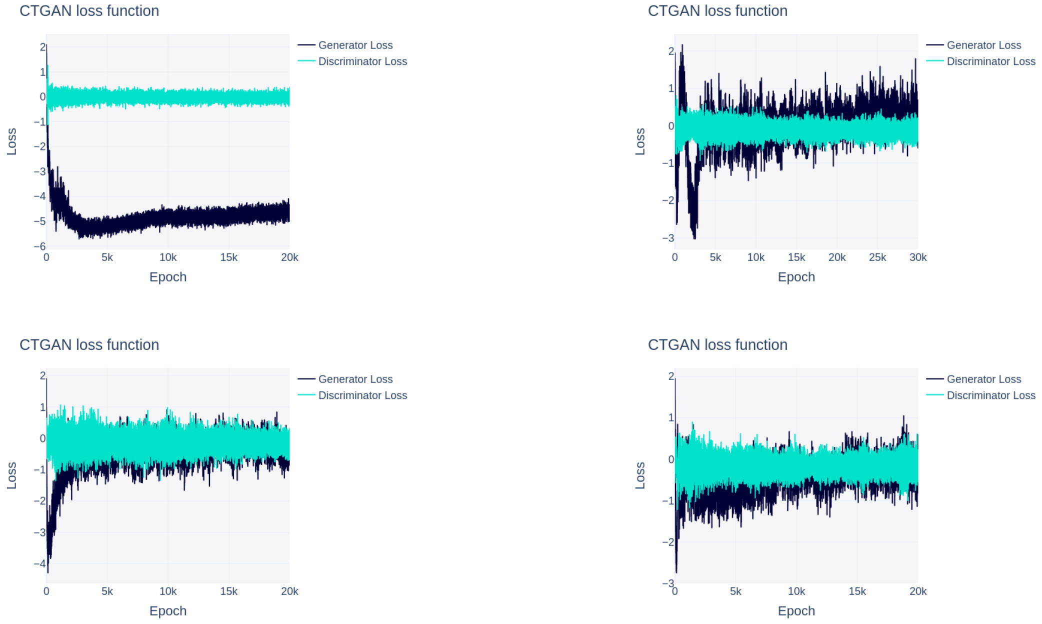

**Figure A3.** Loss functions of trained CTGAN on Steel Plates dataset.

## Appendix B. Scenario 1

The subsequent appendix contains plots that display the recall per class achieved in the Scenario 1 experiments, for each of the tested datasets. These plots provide a detailed view of the model's performance for each class, highlighting the effectiveness of our methods in handling imbalanced data, specially in minority classes.

Wine Dataset

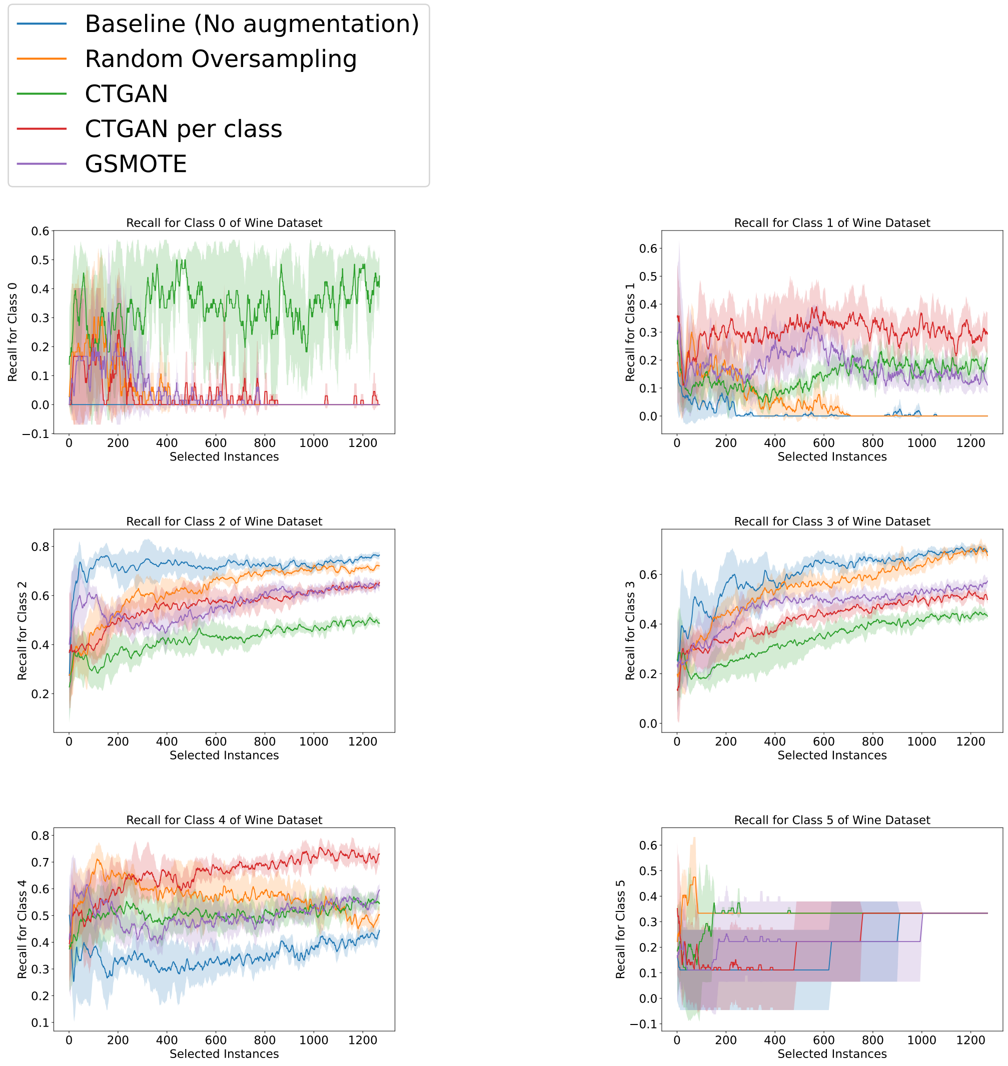

**Figure A4.** Recall for each individual class of Wine dataset with augmentation strategies.

Baseball Dataset

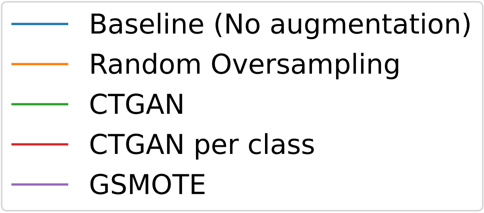

**Figure A5.** *Cont.*

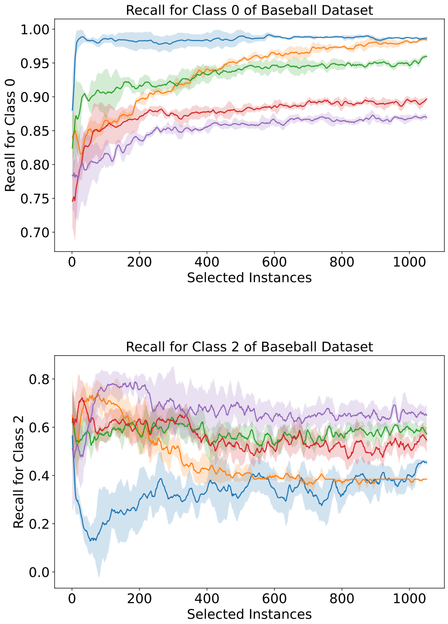

**Figure A5.** Recall for each individual class of Baseball dataset with augmentation strategies.

Steel Plates Dataset

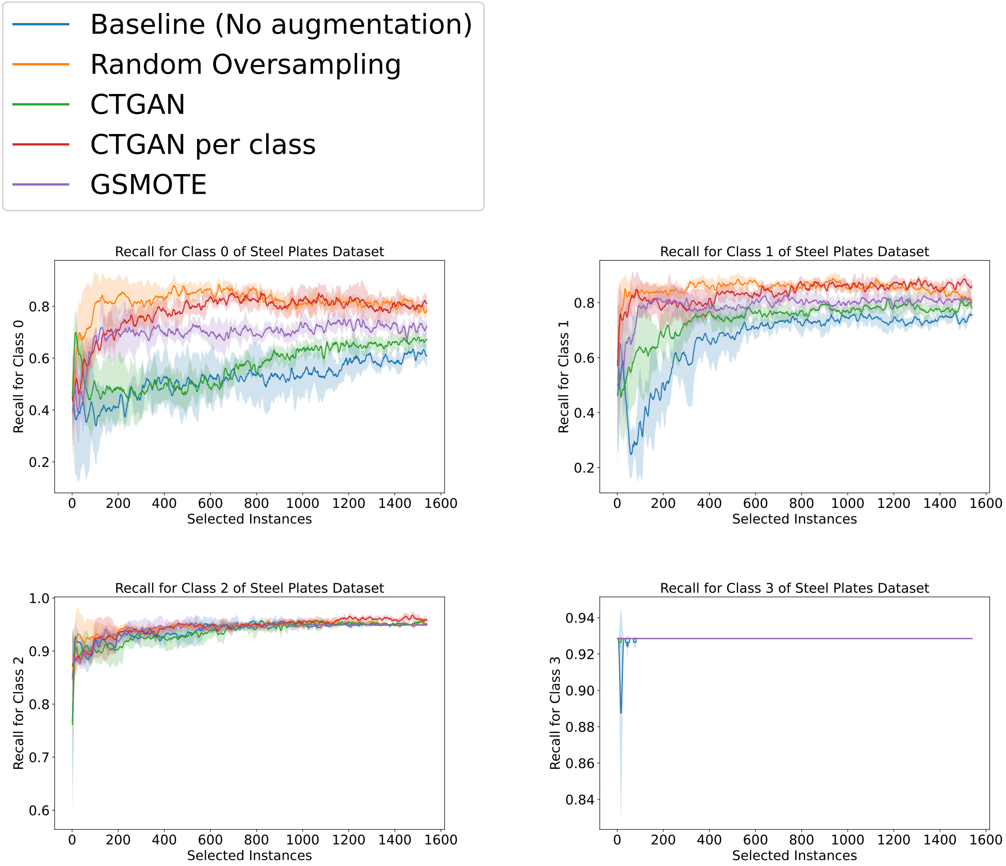

**Figure A6.** *Cont.*

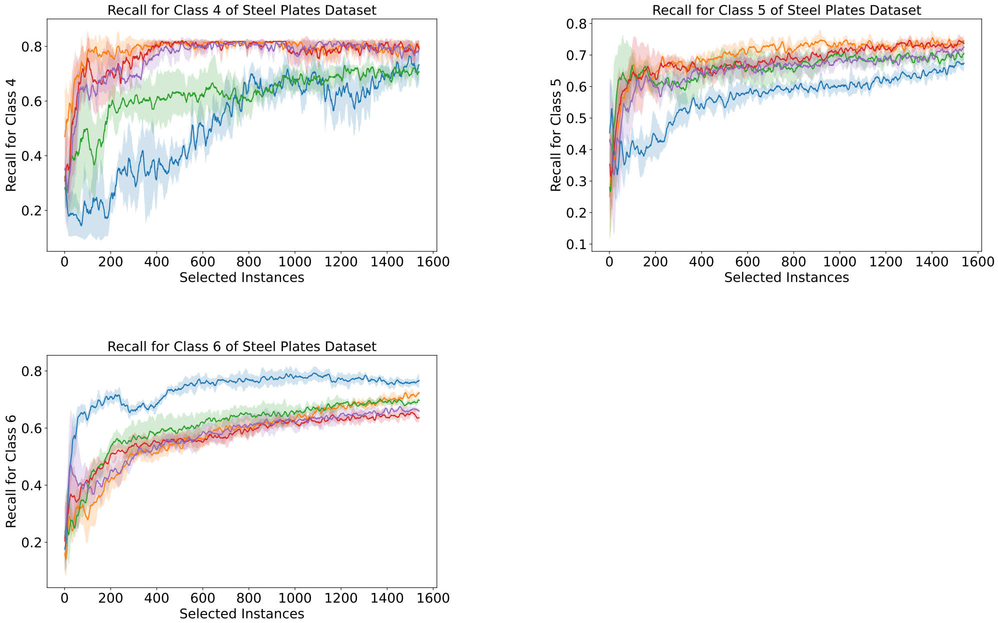

**Figure A6.** Recall for each individual class of Steel Plates dataset with augmentation strategies.

## Appendix C. Scenario 2

The subsequent appendix contains plots that display the recall per class achieved in the Scenario 2 experiments, for each of the tested datasets. In Scenario 2, different data augmentation strategies were compared on an Active Learning scenario.

Wine Dataset

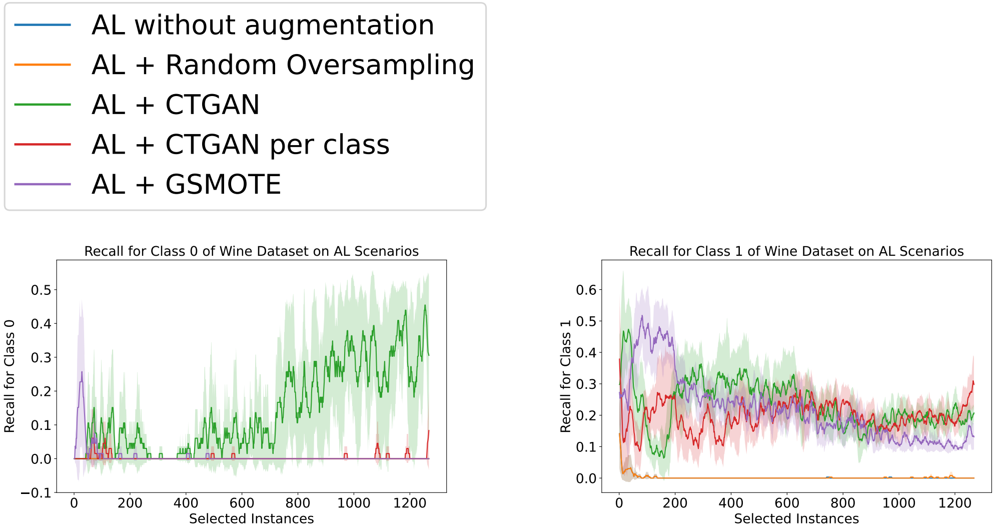

**Figure A7.** *Cont.*

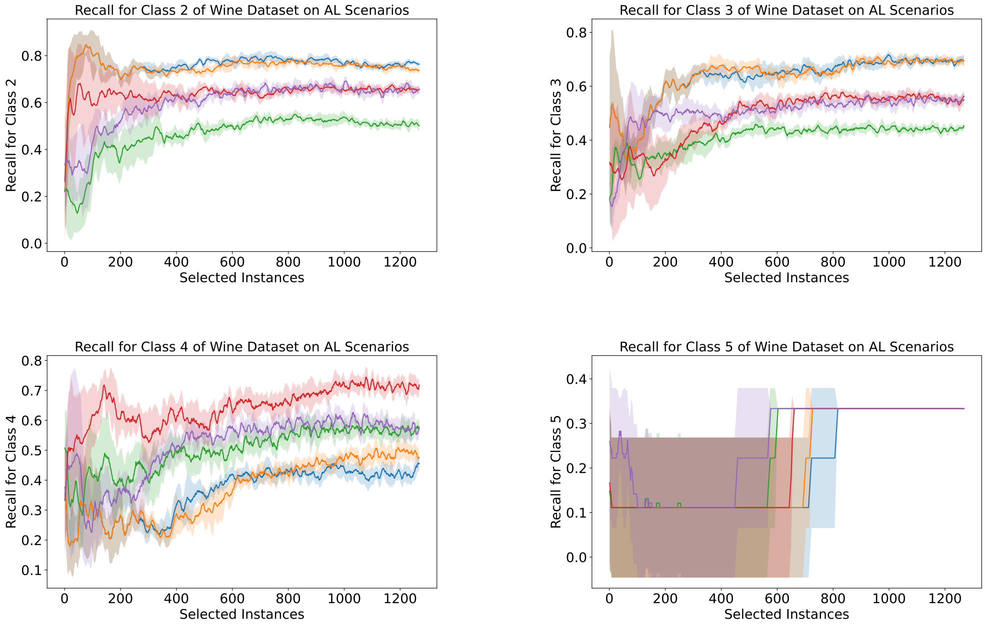

**Figure A7.** Recall for each individual class of Wine dataset with augmentation strategies in active learning scenario.

Baseball Dataset

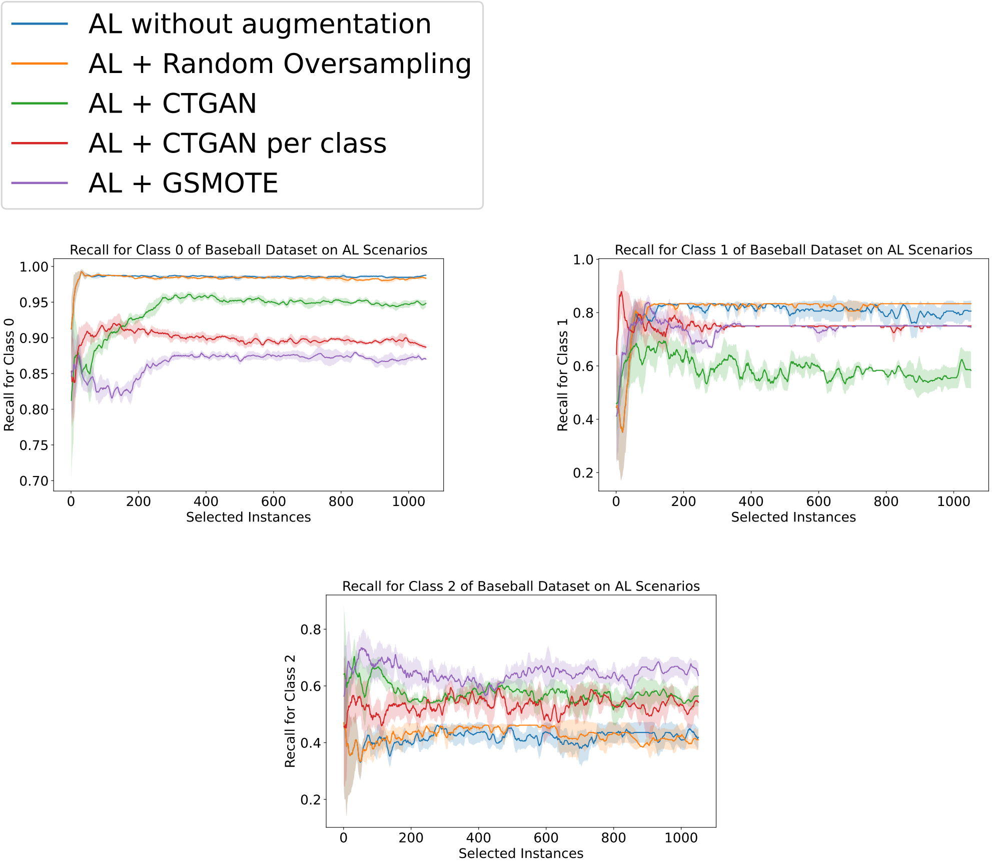

**Figure A8.** Recall for each individual class of Baseball dataset with augmentation strategies in active learning scenario.

Steel Plates Dataset

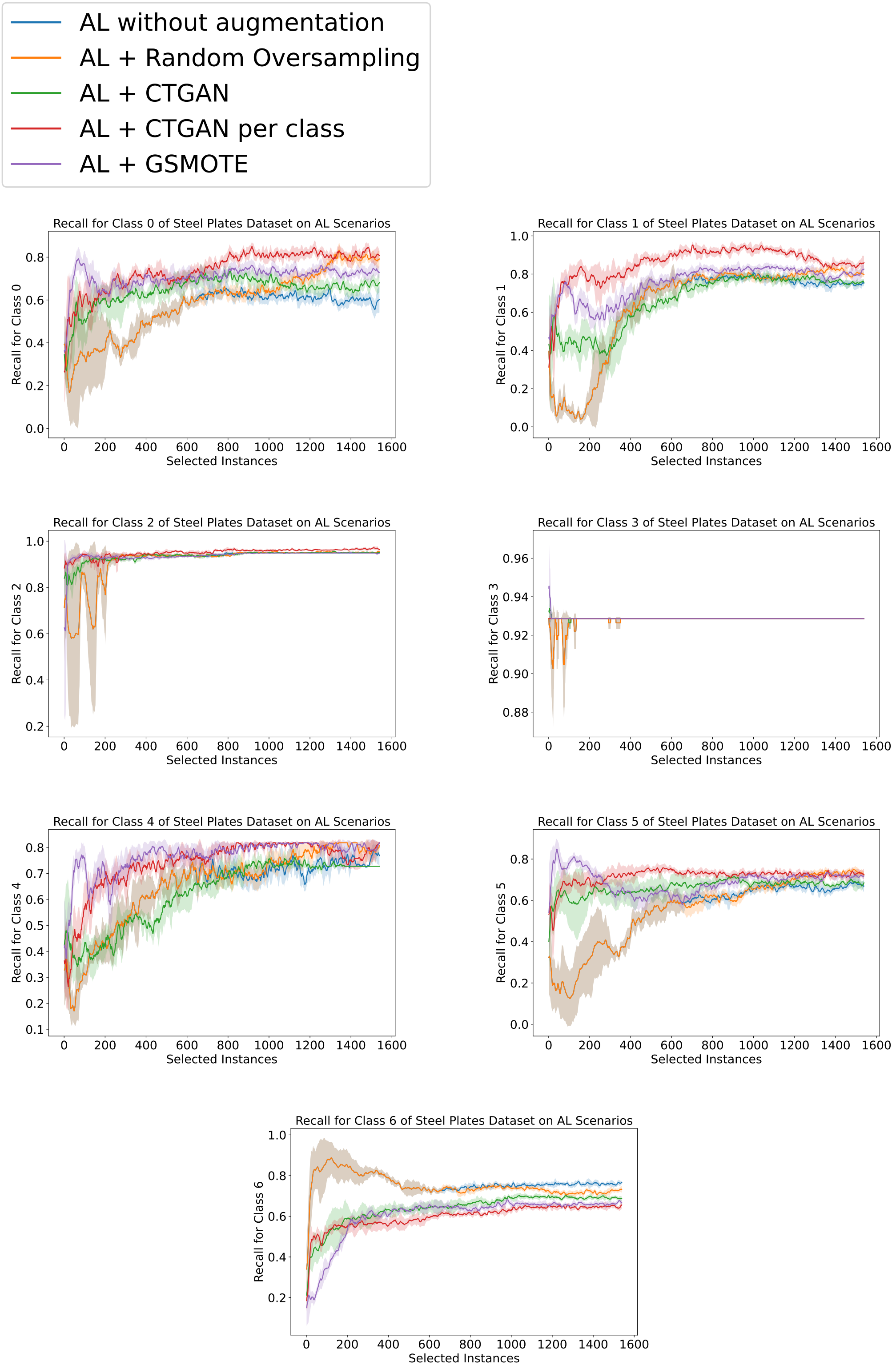

**Figure A9.** Recall for each individual class of Steel Plates dataset with augmentation strategies in active learning scenario.

## Appendix D. Scenario 3

This appendix contains plots that display the recall per class achieved in the Scenario 3 experiments, where the effect of iterative data augmentation in active learning scenarios was studied. Subsections of the appendix represent the two strategies followed to generate synthetic data in Scenario 3. Generating synthetic data for the minority class or generating one instance for all minority classes.

### *Appendix D.1. Minority Class*

Wine Dataset

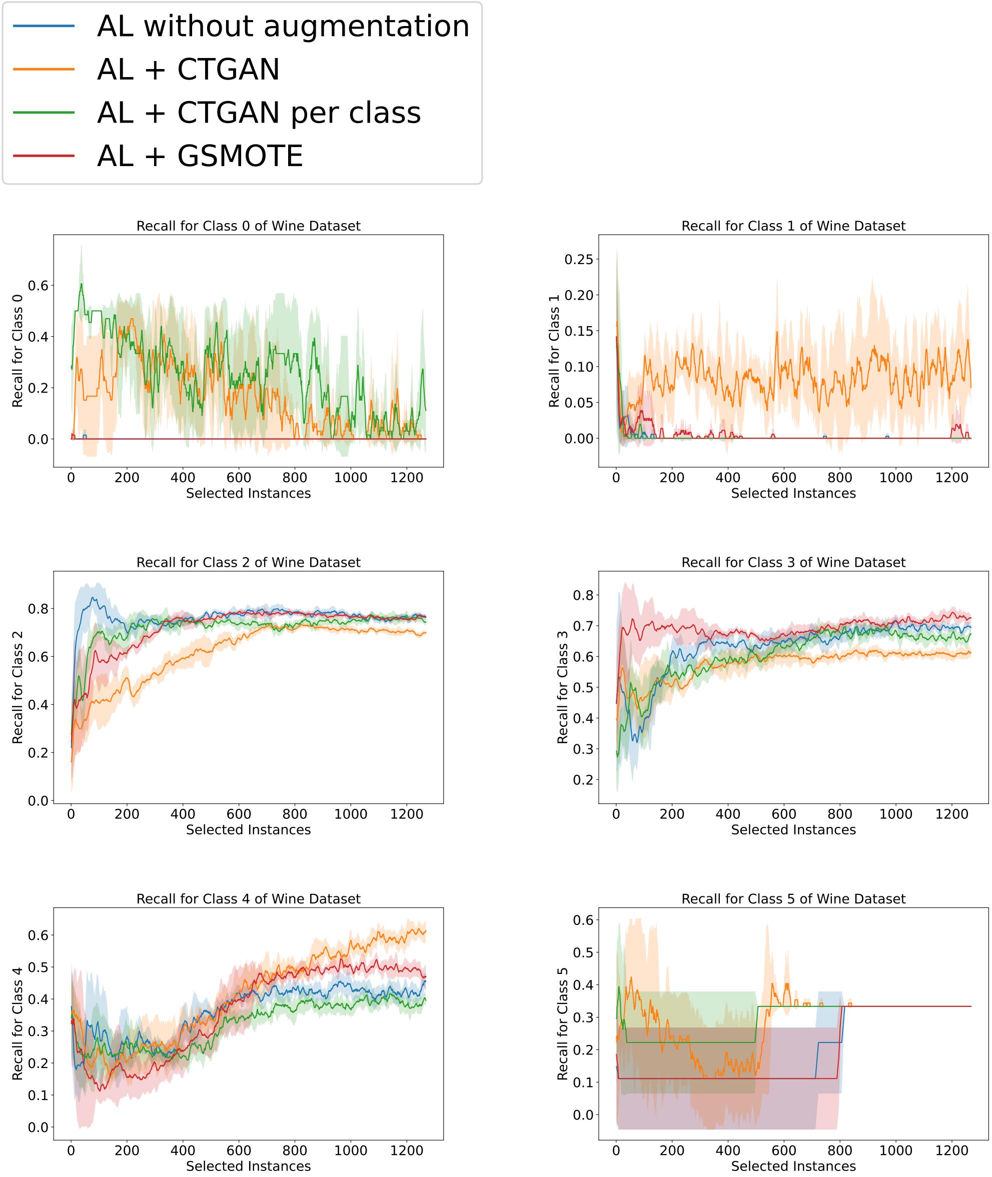

**Figure A10.** Recall for each individual class of Wine dataset with iterative data augmentation and minority class approach.

Baseball Dataset

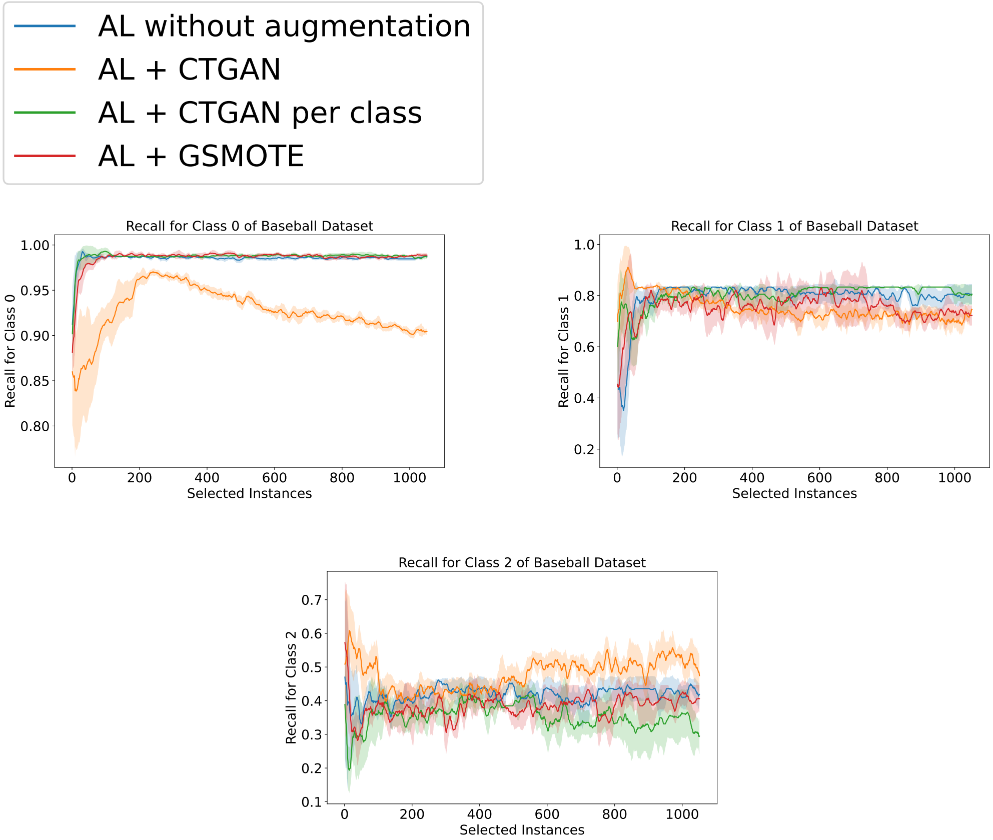

**Figure A11.** Recall for each individual class of Baseball dataset with iterative data augmentation and minority class approach.

Steel Plates Dataset

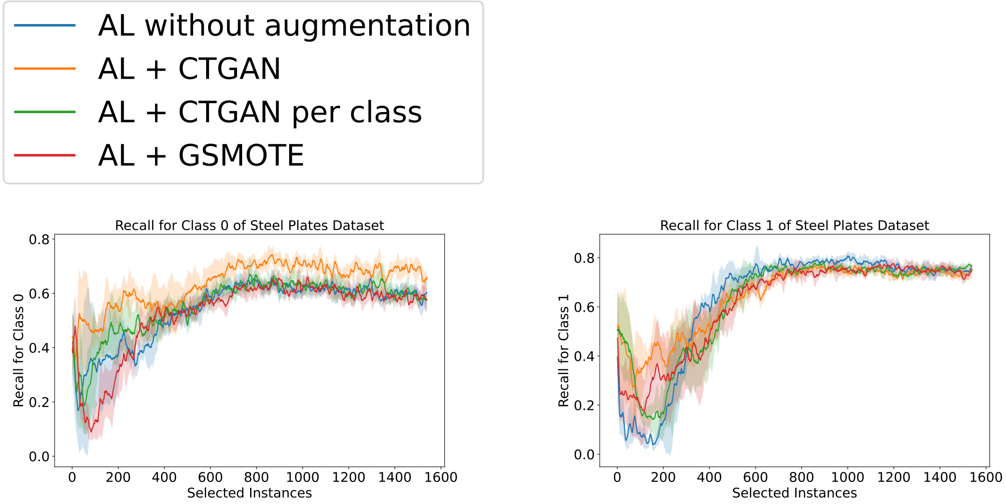

**Figure A12.** *Cont.*

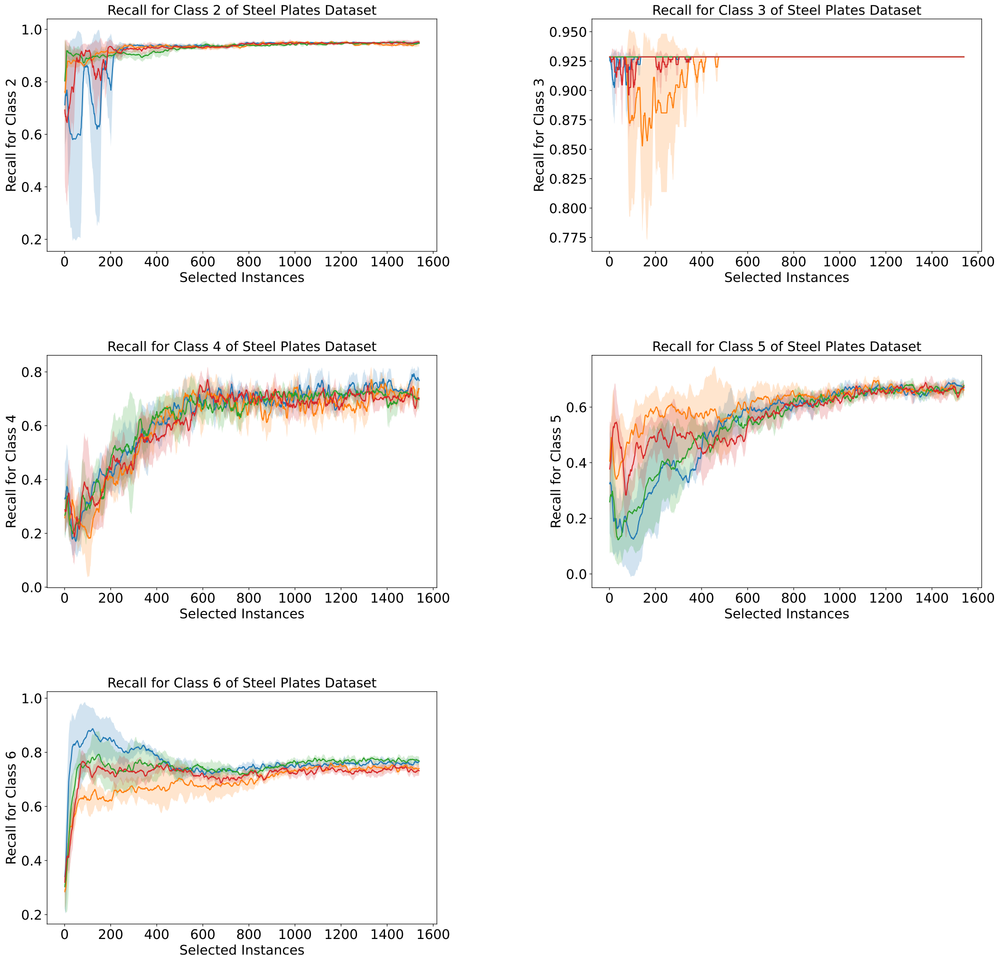

**Figure A12.** Recall for each individual class of Steel Plates dataset with iterative data augmentation and minority class approach.

### *Appendix D.2. All Minority Classes*

Wine Dataset

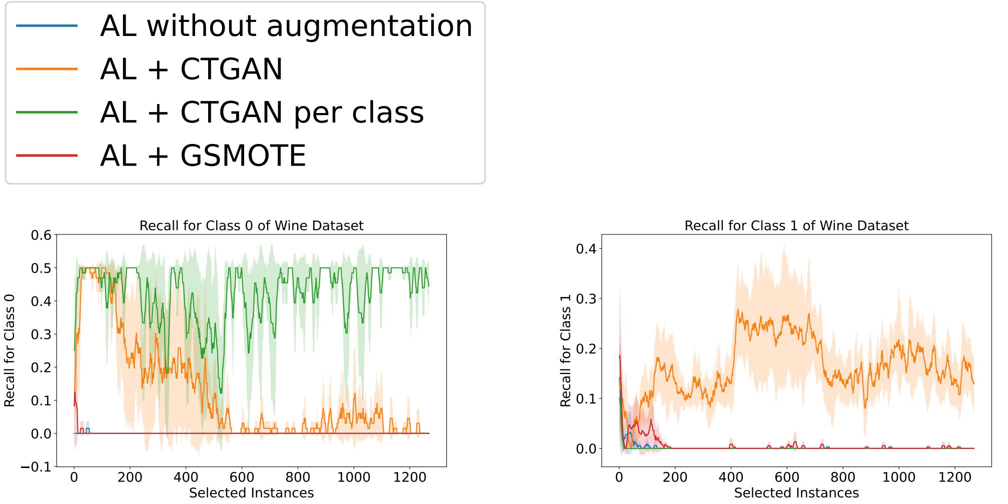

**Figure A13.** *Cont.*

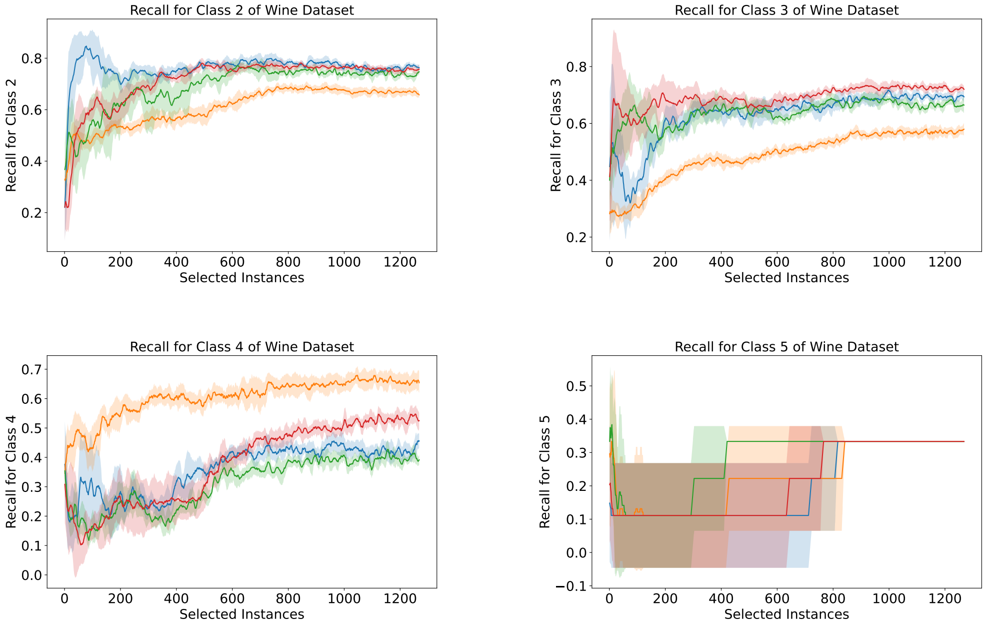

**Figure A13.** Recall for each individual class of Wine dataset with iterative data augmentation and all-minority-class approach.

Baseball Dataset

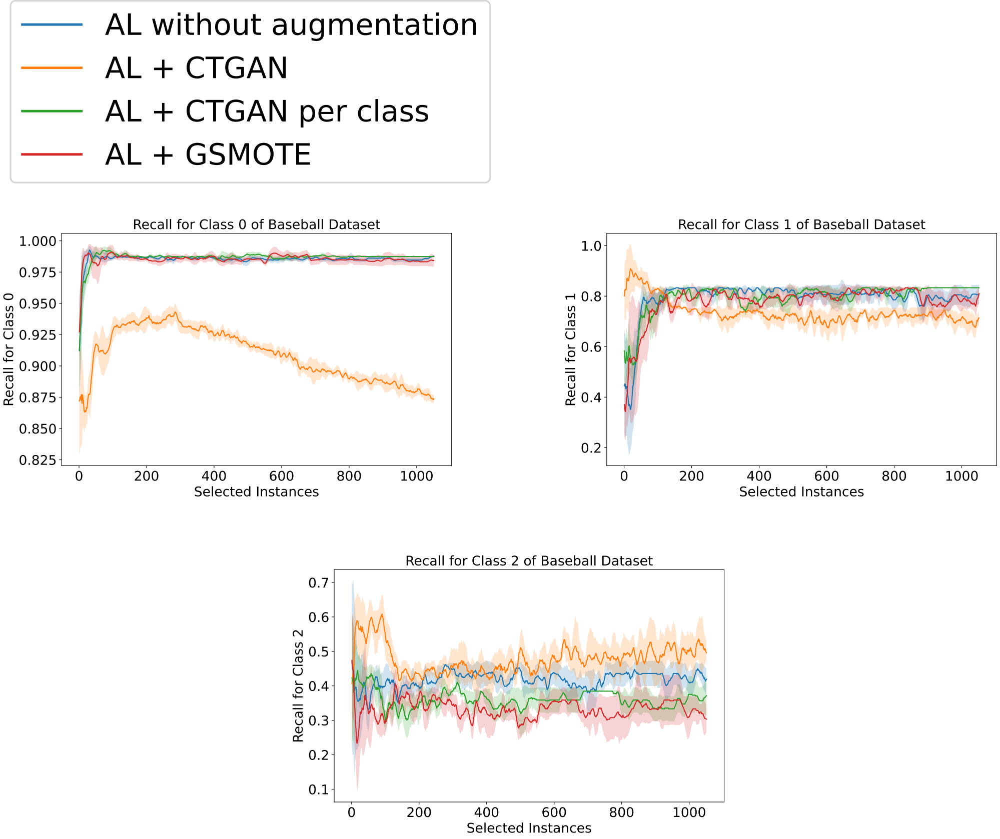

**Figure A14.** Recall for each individual class of Baseball dataset with iterative data augmentation and all-minority-class approach.

Steel Plates Dataset

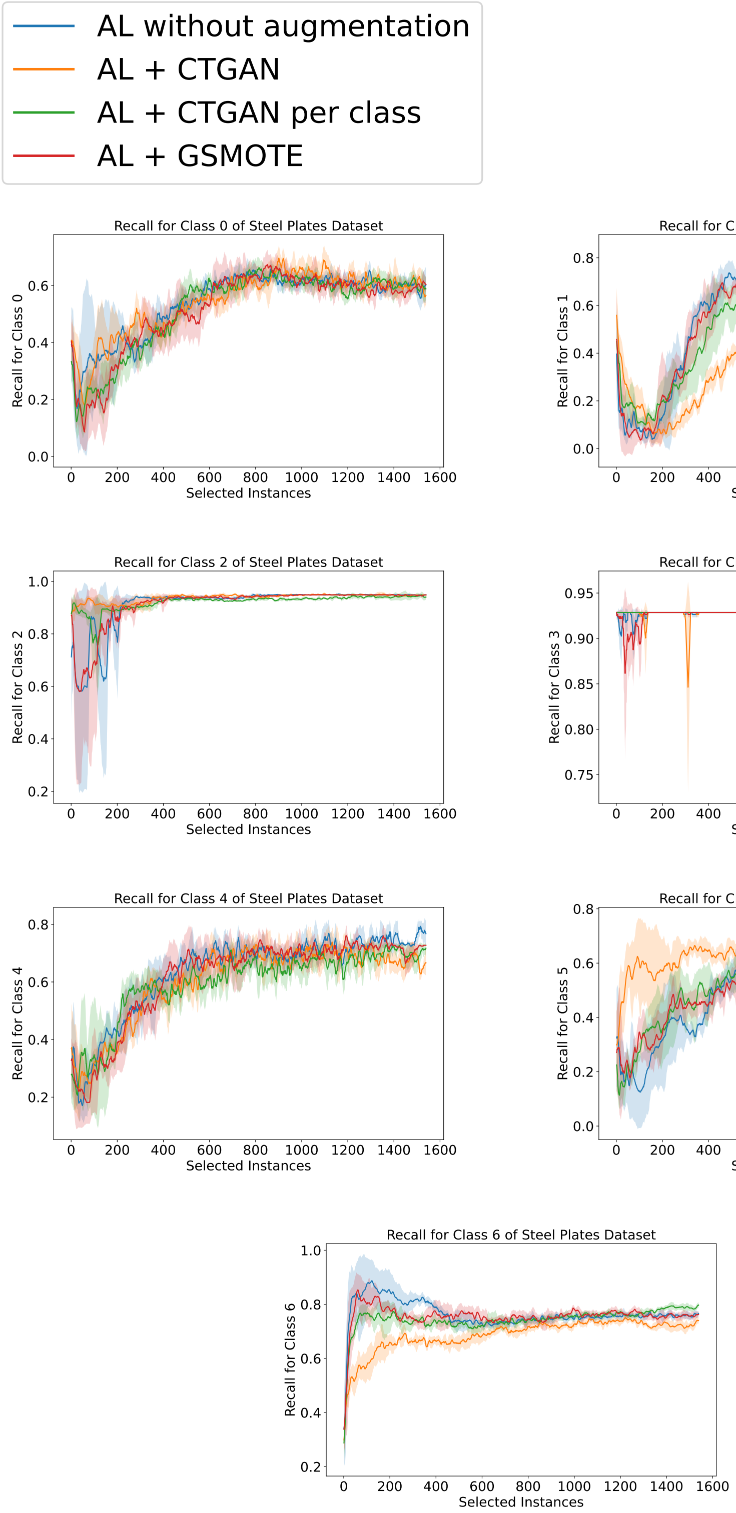

**Figure A15.** Recall for each individual class of Steel Plates dataset with iterative data augmentation and all-minority-class approach.

Recall per class for all-minority-class approach

**Table A4.** Percentage of recall per class for iterative data augmentation techniques in maximum balanced accuracy iteration on active learning scenario. (All minority class approach).

| Dataset | Class | Baseline | CTGAN | CTGAN Per Class | GSMOTE |
|---|---|---|---|---|---|
| Wine | **0** | 0.00% | 10.61% | **50.00%** | 0.00% |
| | **1** | 0.00% | **21.49%** | 0.00% | 0.00% |
| | **2** | **76.89%** | 67.58% | 76.20% | 76.16% |
| | **3** | 70.74% | 57.27% | 67.95% | **73.11%** |
| | **4** | 45.00% | **65.08%** | 40.76% | 54.70% |
| | **5** | **33.33%** | 33.33% | 33.33% | 33.33% |
| Baseball | **0** | 98.43% | 91.71% | **98.87%** | 98.83% |
| | **1** | 83.33% | **86.62%** | 83.33% | 83.33% |
| | **2** | 45.92% | **58.74%** | 38.46% | 35.66% |
| Steel Plates | **0** | 63.26% | **66.95%** | 62.22% | 60.89% |
| | **1** | **79.98%** | 69.22% | 75.36% | 74.08% |
| | **2** | 94.83% | **94.99%** | 94.06% | 94.72% |
| | **3** | **92.86%** | 92.86% | 92.86% | 92.86% |
| | **4** | 75.21% | 73.00% | 69.97% | **76.03%** |
| | **5** | 66.67% | 67.12% | **67.23%** | 67.19% |
| | **6** | 74.39% | 73.65% | **77.76%** | 76.68% |

## References

1. Settles, B. *Active Learning Literature Survey*; University of Wisconsin-Madison Department of Computer Sciences: Madison, WI, USA, 2009.
2. Malbasa, V.; Zheng, C.; Chen, P.C.; Popovic, T.; Kezunovic, M. Voltage stability prediction using active machine learning. *IEEE Trans. Smart Grid* **2017**, *8*, 3117–3124. [CrossRef]
3. Murphy, R.F. An active role for machine learning in drug development. *Nat. Chem. Biol.* **2011**, *7*, 327–330. [CrossRef]
4. Zhong, M.; Tran, K.; Min, Y.; Wang, C.; Wang, Z.; Dinh, C.T.; De Luna, P.; Yu, Z.; Rasouli, A.S.; Brodersen, P.; et al. Accelerated discovery of CO2 electrocatalysts using active machine learning. *Nature* **2020**, *581*, 178–183. [CrossRef]
5. Zhang, J.; Shao, S.; Verma, S.; Nowak, R. Algorithm selection for deep active learning with imbalanced datasets. *Adv. Neural Inf. Process. Syst.* **2024**, *36*, 9614–9647
6. Tharwat, A.; Schenck, W. A survey on active learning: State-of-the-art, practical challenges and research directions. *Mathematics* **2023**, *11*, 820. [CrossRef]
7. Liu, Y.; Liu, Y.; Bruce, X.; Zhong, S.; Hu, Z. Noise-robust oversampling for imbalanced data classification. *Pattern Recognit.* **2023**, *133*, 109008. [CrossRef]
8. Feng, F.; Li, K.C.; Yang, E.; Zhou, Q.; Han, L.; Hussain, A.; Cai, M. A novel oversampling and feature selection hybrid algorithm for imbalanced data classification. *Multimed. Tools Appl.* **2023**, *82*, 3231–3267. [CrossRef]
9. Wongvorachan, T.; He, S.; Bulut, O. A comparison of undersampling, oversampling, and SMOTE methods for dealing with imbalanced classification in educational data mining. *Information* **2023**, *14*, 54. [CrossRef]
10. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
11. Wang, L.; Chen, W.; Yang, W.; Bi, F.; Yu, F.R. A state-of-the-art review on image synthesis with generative adversarial networks. *IEEE Access* **2020**, *8*, 63514–63537. [CrossRef]
12. Xu, L.; Skoularidou, M.; Cuesta-Infante, A.; Veeramachaneni, K. Modeling tabular data using conditional gan. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 7335–7345.
13. Rezvani, S.; Wang, X. A broad review on class imbalance learning techniques. *Appl. Soft Comput.* **2023**, *143*, 110415. [CrossRef]
14. Elamrani Abou Elassad, Z.; Mousannif, H.; Al Moatassime, H. Class-imbalanced crash prediction based on real-time traffic and weather data: A driving simulator study. *Traffic Inj. Prev.* **2020**, *21*, 201–208. [CrossRef]
15. Makond, B.; Wang, K.J.; Wang, K.M. Benchmarking prognosis methods for survivability–A case study for patients with contingent primary cancers. *Comput. Biol. Med.* **2021**, *138*, 104888. [CrossRef]
16. Douzas, G.; Bacao, F. Geometric SMOTE a geometrically enhanced drop-in replacement for SMOTE. *Inf. Sci.* **2019**, *501*, 118–135. [CrossRef]
17. Venkataramana, L.Y.; Jacob, S.G.; Prasad, V.; Athilakshmi, R.; Priyanka, V.; Yeshwanthraa, K.; Vigneswaran, S. Geometric SMOTE-Based Approach to Improve the Prediction of Alzheimer's and Parkinson's Diseases for Highly Class-Imbalanced Data. In *AI, IoT, and Blockchain Breakthroughs in E-Governance*; IGI Global: Hershey, PA, USA, 2023; pp. 114–137.
18. Choi, E.; Biswal, S.; Malin, B.; Duke, J.; Stewart, W.F.; Sun, J. Generating multi-label discrete patient records using generative adversarial networks. In Proceedings of the Machine Learning for Healthcare Conference, PMLR, Boston, MA, USA, 18–19 August 2017; pp. 286–305.

19. Park, N.; Mohammadi, M.; Gorde, K.; Jajodia, S.; Park, H.; Kim, Y. Data synthesis based on generative adversarial networks. *arXiv* **2018**, arXiv:1806.03384.
20. Habibi, O.; Chemmakha, M.; Lazaar, M. Imbalanced tabular data modelization using CTGAN and machine learning to improve IoT Botnet attacks detection. *Eng. Appl. Artif. Intell.* **2023**, *118*, 105669. [CrossRef]
21. Jia, J.; Wu, P.; Zhang, K.; Zhong, J. Imbalanced disk failure data processing method based on CTGAN. In Proceedings of the International Conference on Intelligent Computing, Xi'an, China, 7–11 August 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 638–649.
22. Hong, E.; Yi, J.S.; Lee, D. CTGAN-Based Model to Mitigate Data Scarcity for Cost Estimation in Green Building Projects. *J. Manag. Eng.* **2024**, *40*, 04024024. [CrossRef]
23. Moon, J.; Jung, S.; Park, S.; Hwang, E. Conditional tabular GAN-based two-stage data generation scheme for short-term load forecasting. *IEEE Access* **2020**, *8*, 205327–205339. [CrossRef]
24. Yang, Y.; Li, Y.; Yang, J.; Wen, J. Dissimilarity-based active learning for embedded weed identification. *Turk. J. Agric. For.* **2022**, *46*, 390–401. [CrossRef]
25. Peng, P.; Zhang, W.; Zhang, Y.; Xu, Y.; Wang, H.; Zhang, H. Cost sensitive active learning using bidirectional gated recurrent neural networks for imbalanced fault diagnosis. *Neurocomputing* **2020**, *407*, 232–245. [CrossRef]
26. Wang, K.; Zhang, D.; Li, Y.; Zhang, R.; Lin, L. Cost-effective active learning for deep image classification. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *27*, 2591–2600. [CrossRef]
27. Yu, H.; Yang, X.; Zheng, S.; Sun, C. Active learning from imbalanced data: A solution of online weighted extreme learning machine. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *30*, 1088–1103. [CrossRef]
28. Zhang, H.; Liu, W.; Liu, Q. Reinforcement online active learning ensemble for drifting imbalanced data streams. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 3971–3983. [CrossRef]
29. Liu, W.; Zhang, H.; Ding, Z.; Liu, Q.; Zhu, C. A comprehensive active learning method for multiclass imbalanced data streams with concept drift. *Knowl.-Based Syst.* **2021**, *215*, 106778. [CrossRef]
30. Tharwat, A.; Schenck, W. Balancing Exploration and Exploitation: A novel active learner for imbalanced data. *Knowl.-Based Syst.* **2020**, *210*, 106500. [CrossRef]
31. Tharwat, A.; Schenck, W. A novel low-query-budget active learner with pseudo-labels for imbalanced data. *Mathematics* **2022**, *10*, 1068. [CrossRef]
32. Mahapatra, D.; Bozorgtabar, B.; Thiran, J.P.; Reyes, M. Efficient active learning for image classification and segmentation using a sample selection and conditional generative adversarial network. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Granada, Spain, 16–20 September 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 580–588.
33. Ma, Y.; Lu, S.; Xu, E.; Yu, T.; Zhou, L. Combining active learning and data augmentation for image classification. In Proceedings of the 3rd International Conference on Big Data Technologies, Qingdao, China, 18–20 September 2020; pp. 58–62.
34. Mayer, C.; Timofte, R. Adversarial sampling for active learning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 3071–3079.
35. Sinha, S.; Ebrahimi, S.; Darrell, T. Variational adversarial active learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5972–5981.
36. Mosqueira-Rey, E.; Hernández-Pereira, E.; Bobes-Bascarán, J.; Alonso-Ríos, D.; Pérez-Sánchez, A.; Fernández-Leal, Á.; Moret-Bonillo, V.; Vidal-Ínsua, Y.; Vázquez-Rivera, F. Addressing the data bottleneck in medical deep learning models using a human-in-the-loop machine learning approach. *Neural Comput. Appl.* **2024**, *36*, 2597–2616. [CrossRef]
37. Fonseca, J.; Douzas, G.; Bacao, F. Increasing the effectiveness of active learning: Introducing artificial data generation in active learning for land use/land cover classification. *Remote Sens.* **2021**, *13*, 2619. [CrossRef]
38. Fonseca, J.; Bacao, F. Improving Active Learning Performance through the Use of Data Augmentation. *Int. J. Intell. Syst.* **2023**, *2023*, 1–17. [CrossRef]
39. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 2672–2680.
40. Patki, N.; Wedge, R.; Veeramachaneni, K. The synthetic data vault. In Proceedings of the 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), Montreal, QC, Canada, 17–19 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 399–410.
41. Mosqueira-Rey, E.; Hernández-Pereira, E.; Alonso-Ríos, D.; Bobes-Bascarán, J.; Fernández-Leal, Á. Human-in-the-loop machine learning: A state of the art. *Artif. Intell. Rev.* **2023**, *56*, 3005–3054. [CrossRef]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.