IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 24 April 2023, accepted 3 May 2023, date of publication 8 May 2023, date of current version 23 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3273925

RESEARCH ARTICLE

Ensemble Partition Sampling (EPS) for Improved Multi-Class Classification

BRAHIM JABIR[1], ISABEL DE LA TORRE DÍEZ[2], ERNESTO FRANCISCO BAUTISTA THOMPSON[3,4,5], DEBORA LIBERTAD RAMÍREZ VARGAS[3,4,6], AND ÁNGEL GABRIEL KUC CASTILLA[4,7,8]

[1]LIMATI Laboratory, Sultan Moulay Slimane University, Beni Mellal 23000, Morocco
[2]Department of Signal Theory and Communications, University of Valladolid, 47011 Valladolid, Spain
[3]Higher Polytechnic School, Universidad Europea del Atlántico, 39011 Santander, Spain
[4]Department of Engineering and Projects, Universidad Internacional Iberoamericana, Campeche 24560, Mexico
[5]Facultade de Engenharias, Universidade Internacional do Cuanza, Cuito EN250, Bié, Angola
[6]Engineering and projects, Universidad Internacional Iberoamericana, Arecibo, PR 00613, USA
[7]School of Engineering, Fundación Universitaria Internacional de Colombia, Bogotá 11001, Colombia
[8]Higher Polytechnic School, Universidad de La Romana, La Romana 22000, República Dominicana

Corresponding author: Brahim Jabir (Ibra.jabir@gmail.com)

This work was supported by the European University of the Atlantic, Spain.

ABSTRACT Classification is a commonly used technique in data mining and is applied in various fields such as sentiment analysis, fraud detection, and fault diagnosis. Multiclass classification, which involves more than two classes, is more complex than binary classification. There are mainly two ways to approach multiclass classification, one is to expand the binary classifier into a multiclass classifier through various strategies and the other is to divide the multiclass classification problem into multiple binary problems (binarization). Two popular approaches for binarization are One vs One (OvO) and One vs All (OvA). It is simpler to aggregate the outputs of all binary classifiers as the number of classifiers decreases. However, it causes an imbalance of positive and negative sample numbers, which affects the classification effect of each binary classifier. In this article, we contribute to the field of ensemble learning and multi-class classification by proposing a new method called Ensemble Partition Sampling (EPS). This article presents a new approach to multiclass classification using an ''Ensemble Partition Sampling'' method within the ''one-vs-all'' (OvA) framework. The primary goal of this method is to tackle the problem of data imbalance by incorporating ensemble learning and preprocessing techniques into each binary dataset. The study found that Ensemble Partition Sampling (EPS) is the most effective method for imbalanced and multiclass imbalanced classification, outperforming other methods including OvA, SMOTE, k-means-SMOTE, Bagging-RB, DES-MI, OvO-EASY, and OvO-SMB. The study used CART, Random Forest, and SVM as classifiers, and the results consistently showed that EPS outperformed all other algorithms. Based on the findings, it can be concluded that the EPS approach is a highly effective method for improving classification performance in imbalanced and multiclass imbalanced datasets.

INDEX TERMS Ensemble partition sampling (EPS), one vs one (OvO), one vs all (OvA), multi-class classification, imbalanced learning, multiclass imbalanced classification.

I. INTRODUCTION

In recent years, the field of deep learning has seen a significant increase in research and development in various areas such as computer vision, natural language processing, and speech recognition. One area that has gained a lot of attention is ensemble learning, which is a method of combining multiple models to improve the overall performance of the system. Ensemble learning is particularly useful in deep learning, where the model's complexity is high and the risk of overfitting is significant.

There are several ensemble methods that have been proposed in the literature, such as bagging, boosting, and stacking. However, other methods have been proposed, these

The associate editor coordinating the review of this manuscript and approving it for publication was Xiong Luo.

include: Random subspace method: This method involves training multiple models on different subsets of the feature space [1]. This can increase the diversity of the ensemble and improve performance. Adaptive boosting: This method is a variation of the boosting method, but it adapts the weights of the training samples at each iteration to focus more on the misclassified samples [2]. Hybrid methods: This method combines two or more ensemble methods to improve performance [3]. For example, one could combine bagging and boosting to create a hybrid ensemble. Multi-objective optimization: This method uses multi-objective optimization techniques to find a set of models that are diverse and accurate at the same time [4]. Bayesian Model Averaging (BMA): This method takes a probabilistic approach to ensemble learning, averaging over a set of models with different architectures or hyperparameters, each with a weight given by the posterior probability of the model given the data [5]. Ensemble Selection: This method selects a subset of models from a pool of pre-trained models to form an ensemble [6].

One area of research in deep learning is multi-class classification, which involves assigning instances to one of multiple classes. To address this problem, researchers have proposed methods such as One vs All (OvA) and One vs One (OvO), which aim to reduce the number of classifiers needed while maintaining high performance. OvA involves training one classifier for each class, while OvO involves training one classifier for each pair of classes [7].

To provide context, we briefly review some related work on deep learning methods for imbalanced and multiclass imbalanced classification, several methods have been proposed in the literature to address the challenges of imbalanced and multiclass imbalanced classification. These methods typically involve oversampling or undersampling techniques to balance the distribution of classes in the dataset. However, many of these methods have limitations, such as overfitting, loss of information, and high computational cost. In the paper of "Imbalanced Learning: Foundations, Algorithms, and Applications" [8], the authors discuss the state-of-the-art research on imbalanced learning, including foundations, algorithms, databases, assessment metrics, and major categories of imbalanced learning methods. They also provide an overview of the challenges and opportunities in this field.

In "Class imbalance problem in data mining review" [9], the author presents a brief review of existing solutions to the class-imbalance problem proposed at both the data and algorithmic levels. The author notes that while a common practice to handle the problem of imbalanced data is to rebalance them artificially by oversampling and/or undersampling, some researchers have found that modified support vector machines, rough set based minority class oriented rule learning methods, and cost-sensitive classifiers perform well on imbalanced data sets. The author also notes that current research in imbalanced data is moving towards hybrid algorithms.

Krawczyk in "Learning from Imbalanced Data Sets: Open Challenges and New Trends" [10] discusses the open issues and challenges that need to be addressed to further develop the field of imbalanced learning. The author identifies seven vital areas of research in this topic, covering the full spectrum of learning from imbalanced data such as classification, regression, clustering, data streams, big data analytics and applications. Fanny et al. [11], Ming et al [12], Zhai et al. [13], and Mirza et al. [14] propose different deep learning approaches to address class imbalance, Fanny et al. [11] proposed a method based on Class Expert Generative Adversarial Network (CE-GAN). In this approach, a GAN is trained for each minority class, with the generator network being conditioned on the class label. The discriminator network is shared across all classes. The CE-GAN approach improves the performance of minority classes by generating additional samples that are similar to the original minority samples. Ming et al. [12] proposed a Conditional Wasserstein Generative Adversarial Network-based approach (CW-GAN). The CW-GAN approach is designed to generate samples that are similar to the minority class samples. It works by learning a mapping from the minority class to a higher-dimensional space using a generator network, and then mapping back to the original space using a discriminator network. The CW-GAN approach can generate high-quality synthetic samples that are very similar to the original minority samples. Zhai et al. [13] proposed a diversity oversampling method using generative models. In this approach, generative models are used to generate additional samples for the minority class. However, instead of generating samples that are similar to the original minority samples, the generative models are designed to generate samples that are diverse and cover the entire feature space. This approach improves the performance of minority classes by increasing the diversity of the training data. Mirza et al. [14] proposed deep generative models to counter class imbalance. They proposed two approaches: the first approach involves using a variational autoencoder to generate synthetic samples for the minority class, while the second approach involves using a generative adversarial network to generate synthetic samples. Both approaches can generate high-quality synthetic samples that are similar to the original minority samples and can improve the performance of minority classes. Wang and Yao proposed an ensemble method called AdaBoost.NC [15]. This method is designed to handle multiclass and imbalance effectively and directly. The authors studied the impact of multiminority and multimajority on the performance of two basic resampling techniques and found that "multimajority" tends to be more harmful to the generalization performance. The authors claim that AdaBoost.NC is better at recognizing minority class samples and balancing the performance among classes in terms of G-mean without using any class decomposition. Guo et al. provided an in-depth review of rare event detection from an imbalanced learning perspective [16]. The authors collected 517 related papers that have been published in the past decade and reviewed them from both a technical and a practical point of view. They provided a comprehensive taxonomy of the existing application domains of imbalanced learning and detailed the applications for each category. The authors also suggested further research directions for the imbalanced learning and rare event detection fields. Rodríguez et al. proposed two approaches to extend Random Balance strategy (RandBal) to multiclass imbalance problems [17]. The first approach, called Multiple Random Balance (MultiRandBal), deals with all classes simultaneously. The second approach decomposes the multiclass problem into two-class problems using one-vs-one or one-vs-all and builds an ensemble of RandBal ensembles. The authors claim that both MultiRandBal and OvO/OvA-RandBal are viable extensions of the original two-class RandBal and consistently outperform acclaimed state-of-the-art methods for multiclass imbalanced problems. Li et al. proposed a data preprocessing-based method that combines a One vs One (OvO) decomposition of class pairs and a spectral clustering technique [18]. This method decomposes a multiclass dataset into several binary-class datasets, and then uses spectral clustering to divide the minority classes of binary-class subsets into subspaces and oversample them according to the characteristics of the data. The authors claim that their method has the best overall performance when compared to five state-of-the-art multiclass imbalanced learning methods on seven multiclass datasets.

Rifkin and Klautau [19] have proposed that this simple approach is as accurate as any other approach, assuming that the underlying binary classifiers are well-tuned regularized classifiers such as support vector machines (SVM). However, this thesis contradicts a large body of recent published work on multiclass classification that suggests using methods that are more sophisticated. Rifkin and Klautau support their position through a critical review of the existing literature, a substantial collection of carefully controlled experimental work, and theoretical arguments.

Our article introduces a novel approach to multi-class classification called Ensemble Partition Sampling (EPS). EPS utilizes oversampling and undersampling techniques to create binary training datasets and generate a more robust ensemble of classifiers. In the binary problem, the majority class is a class containing a large number of samples, and the class containing fewer instances known as the minority class. By minimizing the number of deletions for samples in the majority class and the number of syntheses for samples in the minority class, EPS addresses imbalanced and multiclass imbalanced datasets, resulting in improved classification performance. The resulting classifiers are then combined to make a final prediction, resulting in improved accuracy and performance compared to traditional One vs All (OvA) and One vs One (OvO) approaches. Overall, our proposed method demonstrates innovative and effective solutions for multiclass classification. To evaluate the effectiveness of EPS, we compare it to other methods proposed in the literature using benchmark classifiers such as CART, Random Forest, and SVM. The remaining sections of this article are organized as follows: The methodology section explains the two primary components of the EPS method, which involve the One vs All method and ensemble learning approaches. This section also provides a detailed description of how EPS uses oversampling and undersampling techniques to minimize and synthesize samples for both the majority and minority classes. In the results and discussion section, the study presents the experimental findings obtained using CART, Random Forest, and SVM classifiers on multiclass datasets from the public repository KEEL. The section also compares the performance of EPS with other state-of-the-art methods for imbalanced and multiclass imbalanced classification. Finally, the conclusion emphasizes the effectiveness of the EPS method in significantly improving classification performance in imbalanced and multiclass imbalanced datasets.

II. METHODOLOGY

This section describes the proposed approach for multiclass classification, which uses an ‘‘Ensemble Partition Sampling’’ within the ‘‘one-vs-all’’ (OvA) framework to address the issue of data imbalance. The proposed method comprises two main components: the generation of binary training datasets, and the creation and combination of binary classification models. We used the open-source software KEEL to access and utilize multiclass datasets to evaluate the performance of the proposed method. The datasets vary in size, the number of instances ranging from 100 to 57999, and differ in terms of the number of features, with some having only three features while others have as many as 59. The datasets contain both numeric and nominal attributes, and the imbalance ratio varies, with some having a ratio as low as 1.08 and others as high as 4559. We conducted experiments using 27 multiclass datasets from KEEL and tested them using CART, Random Forest, and SVM as benchmark classifiers.

We evaluated the performance of our methods using a specific measure and compared them to typical imbalanced learning methods in the OvA scheme and other methods for solving similar problems in the One vs One scheme or directly. We utilized the CART, Random Forest, and SVM classifiers as our base classifiers. For the CART classifier, we set the ‘min_samples_split’ parameter to 2 and the ‘min_samples_leaf’ parameter to 1. In the case of the Random Forest classifier, we set the ‘n_estimators’ parameter to 100 and the ‘criterion’ parameter to ‘gini’. As for the SVM classifier, we set the ‘C’ parameter to 1 and the ‘kernel’ parameter to ‘rbf’.

In this study, the imbalance ratio (IR) is calculated as the ratio of the number of majority class samples to the number of minority class samples. To ensure the accuracy of the results, all experiments were conducted using 20 iterations of five-fold cross-validation. This means that each experiment used four folds, containing 80% of the dataset samples, as the training set and the remaining 20% as the test set.

The first performance metric used in this paper, which we considered the primary metric in our study, is the Mean Average Value Accuracy (MAvA) [20], which is calculated by taking the ratio of the number of correct classifications in each class and the total number of instances in that class, and then dividing it by the total number of classes (formula 1). The standard deviation of the MAvA is also used.

$$MAvA = \frac{1}{m}\sum\nolimits_{i=1}^{m} Ac_i \tag{1}$$

where:

m: The total number of all classes.

$\mathrm{Ac_i}$: refers to the accuracy of each class, which is calculated by taking the ratio of the number of correct classifications in a particular class to the total number of instances in that class. In other words, it measures the proportion of correctly classified instances in a specific class. For example, if a model correctly classifies 90 out of 100 instances in a particular class, then the accuracy of that class would be 0.9 or 90%. The formula (2) for $\mathrm{Ac_i}$ can be expressed as:

$$Ac_i = \frac{\text{Number of correctly classified instances in class } i}{\text{Total number of instances in class } i} \tag{2}$$

We used the Holm-Bonferroni test with a significance level of $\alpha = 0.05$ to determine if our proposed method performed significantly better than other methods. This test allowed us to compare the performance of different methods in a pairwise manner and adjust for multiple comparisons, ensuring that any significant differences found were not due to chance [21], [22]. The corrected p-value is used to determine the significance of the comparison between the proposed method "EPS" and the other methods (imbalanced learning and multiclass imbalanced classification) for different models (CART, Random Forest, and SVM).

In addition to these methods, we also utilized precision, recall, F1-score, and G-Mean as metrics to evaluate the performance of various multiclass imbalanced classification methods [23].

Precision refers to the percentage of correctly identified true positives out of all predicted positives. The formula (3) for precision is:

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

where TP stands for true positives and FP stands for false positives.

Recall, also known as sensitivity or true positive rate, measures the percentage of correctly identified true positives out of all actual positives. The formula (4) for recall is:

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

where FN stands for false negatives.

The F1-score is the harmonic mean of precision and recall, providing a balanced measure of both metrics. It ranges from 0 to 1, with a score of 1 indicating perfect precision and recall. The formula (5) for F1-score is:

$$F1 - score = 2 \times \frac{recall \times precision}{recall + precision} \tag{5}$$

G-Mean is a metric that measures the balance between classification performances on both the majority and minority classes. It takes into account both true positive and true negative rates and is especially useful for avoiding overfitting the negative class and underfitting the positive class. The formula for G-Mean (6) is the square root of sensitivity (recall) times specificity (true negative rate):

$$G - mean = \sqrt{sensitivity \times specificity} \tag{6}$$

A. FIRST PART OF OUR MODEL

The first part in the proposed approach is the creation of binary training datasets. This section explains the procedures for generating these datasets, including determining the number of samples to be taken and implementing the sampling method. The process of determining the sample numbers is outlined in the next section (A), while section (B) provides in-depth information on the sampling methods used.

(A) The method called the OvA approach used to enhance the performance of a classification model by dividing a dataset with multiple classes into several binary datasets [19]. Each binary dataset has a unique imbalance ratio and utilizing diverse training sets can significantly improve the classification effect. This method creates multiple training sets with increasing number of samples by utilizing the concept of incremental arithmetic progression to set the number of samples for each binary dataset. The original training set is repeatedly sampled with increasing sample numbers to generate multiple balanced training sets with different distributions. This method addresses the issue of varying imbalance ratios among binary datasets and increases the diversity of the model. The number of samples is calculated by determining the range of sample numbers and using a specific formula (7) to obtain a set of sample numbers.

$$num_{it} = \frac{N\,(more_i) - N(less_i)}{T - 1} \times\ (t - 1) + N\ (less_i) \tag{7}$$

- $less_i/more_i$ referring to minority and majority samples in a binary training dataset.
- $\mathrm{N}(less_i)$ is the total number of minority samples in the ith binary training dataset.
- $\mathrm{N}(more_i)$ is the total number of majority samples in the ith binary training dataset.
- $num_i = \{num_{i1}, num_{i2}, num_i, \ldots, num_{iT}\}$ is a set of the number of sampling, obtained where:
- num_i1 is the lower limit $\mathrm{N}(less_i)$ of the interval, and num_{iT} is the upper limit $\mathrm{N}(more_i)$ of the interval.
- T is the total number of classifiers (the number of training set generation).
- num_{it} is the number of sampling for the tth classifier corresponding to the ith binary dataset.

(B) The s-Random undersampling and br-SMOTE approaches are used to identify important samples in different regions by analyzing the neighborhood samples' classes [24]. This includes identifying safe samples, borderline samples, rare samples, and outliers. In this process, HVDM is used as a distance metric to identify the k closest samples to a given example from the training set. HVDM stands for Hellinger Distance-based Voting Metric, which is a statistical measure that calculates the similarity between probability distributions. After identifying the k closest samples, the relationship between their classes and the class of the current sample is analyzed to determine the type of the current sample. If at least 4 of the 5 nearest neighbors share the same class, the sample is considered safe. If 2 or 3 of the 5 nearest neighbors share the same class, the sample is considered borderline. If only one of the 5 nearest neighbors is the same class, it is considered a rare sample. If the sample is surrounded by samples from different classes, it is considered an outlier. Any outliers found in this process are removed.

This method enhances the performance of a classification model by dividing the original dataset into several binary datasets. For each binary dataset, the majority samples are reduced through random undersampling while the minority samples are increased through oversampling. The process of undersampling only removes the safe samples of the majority class *maj*, while retaining the borderline $maji_{\mathrm{b}}$ and rare samples $maji_{\mathrm{r}}$. The number of samples in the reduced majority class N(maj'_{is}) calculated using formula (8). If the number of safe samples is less than the number of other types of samples or the number of other types is greater than the number of samples, random undersampling is applied to all the majority samples. The details of this process are outlined in the next section. In the oversampling process, all safe samples of the minority class are retained while borderline and rare samples of the minority class are used to generate new samples using SMOTE as they are believed to have a significant impact on the classification results [25]. The number of samples in the oversampled minority class N($min'_{\ ir}$) + N(min'_{ib}) calculated using formula (9). If there are no borderline or rare samples, all minority samples are oversampled using br-SMOTE. The specific details of this process are outlined in the next section.

$$N\left(maj'_{\ is}\right) = num_{it} - N\left(maj_{ib}\right) - N\left(maj_{ir}\right) \tag{8}$$

$$N\left(min'_{ir} + min'_{\ ib}\right) = N\left(num_{it}\right) - N\left(min_{is}\right) \tag{9}$$

1) S-RANDOM UNDERSAMPLING

Algorithm 1 presents the function of s-Random undersampling, it takes in the original dataset of majority class D_maj and a number of samples to be selected num, and returns the undersampled dataset D_maj_prime. The "type" column in D_maj is used to distinguish between the different types of samples (safe, borderline, and rare).

The s_Random_undersampling function takes the majority class dataset D_maj and the desired number of samples num as inputs. First, it separates the safe samples from other samples. Then, it computes the number of safe and other samples in the dataset. If there are more other samples than safe samples or more than num samples are desired, the function samples num safe samples from the dataset. Otherwise, it samples num - N_other safe samples and concatenates them with all other samples to form the final dataset D_maj_prime. The function returns the balanced dataset with the same number of safe and other samples.

Algorithm 1 s_Random_Undersampling Function

```
Input: D_maj (dataframe), num (integer)
Output: D_maj_prime (dataframe)
1.    safe_samples = D_maj[D_maj["type"]==1] = other_samples-
      D_maj[D_maj["type"]!=1]
**Separate the safe samples from other samples**
2.    N_other = len(other_samples)
      N_safe = len(safe_samples)
**Compute the number of safe and other samples**
if N_other > N_safe or N_other > num:
D_maj_prime = safe_samples.sample(num)
else:
D_maj_s = safe_samples.sample(num - N_other)
D_maj_prime = pd.concat([D_maj_s, other_samples])
**If there are more other samples than safe samples or more than num samples
are desired, sample num safe samples. Otherwise, sample num - N_other safe
samples and concatenate them with all other samples to form the final dataset
D_maj_prime.**
Return D_maj_prime
```

2) BR-SMOTE

In the Algorithm 2, we present the function of the SMOTE, which is a function that implements Synthetic Minority Oversampling Technique (SMOTE) algorithm, which is used to oversample the minority class. It takes in two arguments, the dataset and the number of samples after oversampling. It returns a dataset with oversampled minority class. The function br_SMOTE takes in two arguments, the input dataset Dmin and the number of samples after oversampling num. It first counts the number of samples of type 1 (safe samples) and the number of samples of other types. If the number of samples of other types is 0, it applies the SMOTE algorithm to oversample the entire dataset and returns the oversampled dataset. If the number of samples of other types is not 0, it retains the samples of type 1, applies the SMOTE algorithm to oversample the samples of type 2 and 3, and concatenates the samples of type 1 with the oversampled samples of type 2 and 3. The function returns the oversampled dataset.

B. SECOND PART OF OUR MODEL

The goal of the second part of our approach is to aggregate and construct binary classification models. The proposed solution for solving the data imbalance problem in the OvA scheme is called the Ensemble Partition Sampling. It is used to build binary classification models. First, the training samples are divided into four categories: safe samples, borderline samples, rare samples, and outliers. The outliers are then removed. Next, the original dataset with m classes is split into m binary datasets. For each binary dataset, a set of sampling numbers, numi, is calculated using formula (7)

Algorithm 2 Br_SMOTE(Dmin, num)

```
Input: Dmin - dataset with minority class samples
num - number of synthetic samples to generate
Output: Dmin_prime - oversampled dataset
1. Select safe samples (type 1) from Dmin and create safe_samples.
2. Select not-safe samples (types 2 and 3) from Dmin and create not_safe_samples.
3. Get the number of safe and not-safe samples, N_safe and N_not_safe.
4. If N_not_safe is 0, oversample Dmin using SMOTE and assign the result to Dmin_prime.
5. If N_not_safe is not 0, oversample not_safe_samples using SMOTE with num=N_safe and assign the result to Dmin_br_prime.
6. Concatenate safe_samples and Dmin_br_prime and assign the result to Dmin_prime.
7. Return Dmin_prime.
```

within the range $[N(\mathrm{less_i}), N(\mathrm{more_i})]$. Then, a binary classification model with T sub-classifiers is created for each binary classification problem using Bagging. In each iteration, the number of majority samples is reduced and the number of minority samples is increased to create a balanced subset for training a binary classifier using br-SMOTE and s-Random undersampling. If there is no imbalance between the number of positive and negative samples in the binary training subset, T = 1 and the original binary training set is used to train a binary model. Finally, these binary classifiers are combined using the simple averaging method to create the final binary classification model. Algorithm 3 provides more details on how to construct the binary classification models.

The above algorithm is just a skeleton structure of the algorithm, it might not run without implementing the helper function calls like divide_samples, create_binary_datasets, calculate_sampling_numbers, oversample, undersample, train_classifier, and get_final_model.

We use a method called Max strategy to combine all the individual binary models. Each binary classifier $h(x)$ (formula 10) will predict the unknown instance, and we will get an output vector $\boldsymbol{r} = \{r_1, r_2, r_3, \ldots, r_m\}$ where r_i is how confident the ith binary classifier is about the positive class. The label corresponding to the highest value in $\boldsymbol{r}$ is chosen as the final output.

$$\mathrm{h}\left(\mathrm{x}\right) = \frac{1}{T}\sum_{t=1}^{T} h_{it}\left(x\right) \tag{10}$$

III. RESULT AND DISCUSSION

In this section, we present and discuss the findings of our study on ensemble learning and multiclass classification utilizing the proposed Ensemble Partition Sampling (EPS) method. Specifically, we will compare the performance of our proposed method with imbalanced learning methods (A) and multiclass imbalanced classification methods (B). To present a clear and comprehensive comparison, we have used the MAvA metrics as the primary comparison metric, which is presented in tables, while other metrics are illustrated in figures. We will analyze and discuss the findings in detail in the following subsections.

Algorithm 3 Diff_Partition_Sampling_Ensemble(D, T)

```
Input:
  a. D: dataset
  b. T: number of iterations
Divide samples into safe, borderline, rare, and outliers and delete outliers:
  a. divide_samples(D)
  b. D = D.drop(outliers)
Set the number of binary datasets to the number of classes in D:
  a. m = len(D.classes)
Create binary datasets Di:
  a. Di = create_binary_datasets(D)
For each binary dataset Di:
  a. If count(D_1 [1], 'p') != count(D_1[i], 'n'):
    i. If count(D_1 [1], 'p') < count(D_1 [1], 'n'):
      1. more = count(D_i[i], 'n')
      2. less = count(D_1 [1], 'p')
    ii. Else:
      1. more = count(D_i[i], 'p')
      2. less = count(D_1 [1], 'n')
  b. Calculate the sampling numbers using the oversampling method 'br-SMOTE' for less and the undersampling method 'S-Random' for more:
    i. num = calculate_sampling_numbers(less, more)
  c. For each iteration t in range(T):
    i. Sample a number of instances equal to sample_num from the minority class using the oversampling method 'br-SMOTE':
      1. less = oversample(less, sample_num, method= "br-SMOTE")
    ii. Sample a number of instances equal to sample_num from the majority class using the undersampling method 's-Random':
      1. more = undersample(more, sample_num, method= 's-Random')
    iii. Combine the oversampled minority class and undersampled majority class to create a new dataset D_1_less_more:
      1. D_1_less_more_ = combine_classes(D_1_1, less, more)
    iv. Train a classifier on D_1_less_more:
      1. $h_{i\_t}$ = train_classifier(D_1_less_more_)
    v. Store the trained classifier $h_{i\_t}$
  d. Get the final model $h_i$:
    i. $h_i$ = get_final_model($h_{i\_t}$)
Return $h_i$.
```

A. COMPARISON OF THE PROPOSED METHODS WITH IMBALANCED LEARNING METHODS

In this section, we aimed to evaluate the effectiveness of the proposed Ensemble Partition Sampling (EPS) method

TABLE 1. Comparison of results of imbalanced learning methods using cart with mava metric.

Datasets	EPS	OvA	k-means-SMOTE	SMOTE	Bagging-RB
automobile	**80.96±0.75**	70.13±0	66.61±2.8	67.37±2.28	73.65±3.47
balance scale	58.1±0.44	55.4±0	55.22±1.36	54.99±1.68	**58.24±1.65**
car	**98.3±0.24**	94.81±0	86.1±1.4	84.43±1.27	93.93±2.35
cleveland	**31.59±1.63**	28.07±0	30.1±3	28.97±1.95	30.7±2.2
contraceptive	**48.76±0.6**	45.94±0	45.05±0.86	45.38±0.72	47.4±1.4
dermatology	94.18±0.43	91.82±0	91.52±1.13	89.3±0.59	**94.62±0.94**
ecoli	59.49±2.32	50.95±0	55.72±1.57	48.1±1.77	**60.03±4.39**
flare	**62.27±0.55**	59.96±0	60.49±0.78	60.28±0.74	60.27±2.06
glass	**71.22±1.73**	64.35±0	61.68±2.78	58.79±2.16	67.38±3.71
hayes-roth	**86.25±1.2**	74.5±0	74.54±1.84	74.5±1.64	83.33±2.9
led7digit	**70.52±0.59**	70.02±0	70.14±0.41	69.94±0.32	66.15±0.83
lymphography	**83.04±0.76**	72.52±0	70.68±3.82	68.24±3.58	75.65±3.98
newthyroid	**94.03±0.55**	90.79±0	90.86±0.49	91.32±1.2	92.21±1.76
page-blocks	89.01±0.32	80.29±0	80.23±1.06	79.75±0.76	**89.64±2.69**
penbased	**98.68±0.05**	96.8±0	95.79±0.19	94.51±0.19	97.83±0.35
satimage	**89.12±0.17**	85.93±0	85.38±0.36	80.88±0.28	86.38±1.68
segment	**97.68±0.1**	96.71±0	97.64±0.23	95.57±0.32	97.03±0.38
shuttle	95.88±0.01	93.8±0	93.82±0.02	93.89±0.03	**96.84±0.18**
splice	**93.86±0.2**	89.11±0	89.17±0.31	89.15±0.27	92.28±0.42
thyroid	**95.07±0.83**	93.98±0	93.09±0	93.25±0.34	95.06±2.38
vehicle	**74.67±0.39**	71.02±0	66.21±1.2	65.86±1.2	73.15±1.21
vowel	**89.62±0.46**	86.16±0	86.34±0.97	85.23±1.06	81.82±2.05
wine	94.19±0.6	90.09±0	90.01±1.09	90.65±1.38	**94.78±1.66**
winequality-red	**39.38±0.6**	34.25±0	35.5±2.02	36.4±2.43	36.32±2.16
winequality-white	**43.04±0.72**	38.76±0	38.75±1.38	38.51±1.29	39.01±2.03
yeast	**54.13±0.67**	52.49±0	49.12±1.1	45.69±0.81	52.68±1.55
zoo	**92.35±1.18**	82.22±0	81.56±1.69	82.22±0	88.44±3.45
Average	**77.23±0.67**	72.61±0	71.90±1.25	70.71±1.12	74.99±1.99
Ranking(p=7.13e−14)	1.22	3.63	3.74	4.22	2.19

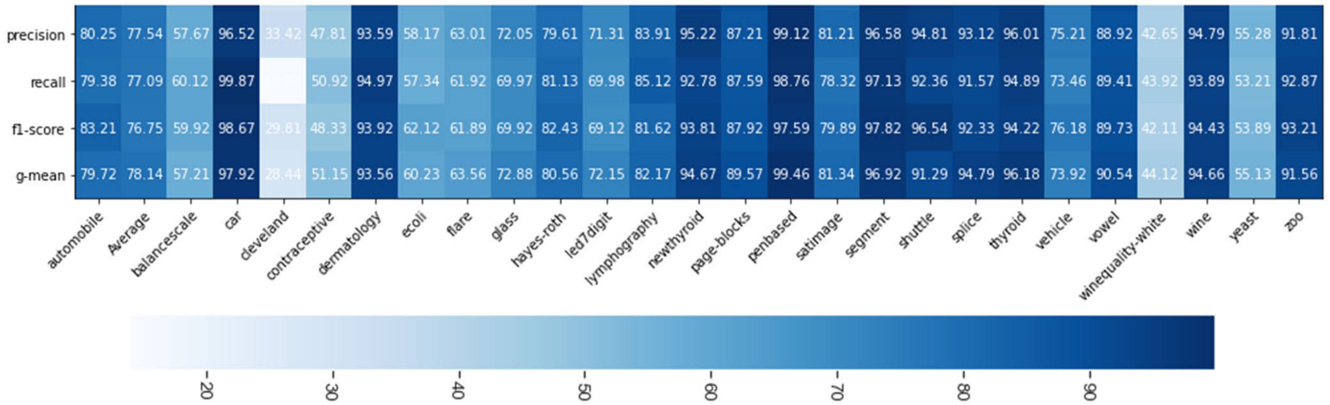


FIGURE 1. Results of imbalanced learning methods using CART with different metrics.

by comparing it to other imbalanced learning methods that have been previously proposed in the literature. To do this, we selected a number of representative methods from existing literature, including: OvA, SMOTE [26], k-means-SMOTE [27], and Bagging-RB [28]. The table 1 presents results of imbalanced learning methods using CART as a classifier on different datasets. It appears that EPS method performed best, followed by Bagging-RB, k-means-SMOTE, SMOTE, and OvA. From the figure 1, we observe that the performance of the proposed method varies across datasets. Some datasets, such as the car dataset, show consistent high performance across all metrics, while others, such as the winequality-red dataset, have consistently average performance across all metrics. For some datasets, there is a significant difference in performance across different metrics. For example, the contraceptive dataset shows high precision indicating that the model performs well in identifying positive instances but has. The results suggest that the proposed method may be more effective on certain types of datasets, such as those with well-separated classes and a clear decision boundary.

The table 2 presents the results of different imbalanced learning methods using Random Forest as a classifier on various datasets. The highest ranked method is EPS, the results

TABLE 2. Comparison of results of different imbalanced learning methods using random forest with MAvA metric.

Datasets	EPS	OvA	k-means-SMOTE	SMOTE	Bagging-RB
automobile	79.35±0.68	**79.43±0**	77.3±2.09	78.78±0.78	60.13±3.59
balancescale	60.35±0.1	**62.5±0**	61.22±0.33	61.66±0.26	59.89±1.77
car	**95.64±0.21**	94.44±0	92.68±0.81	93.08±1.23	87.67±5.07
cleveland	29.87±0.92	28.5±0	**31.27±1.22**	31.01±1.74	30.99±1.82
contraceptive	**50.51±0.13**	50.09±0	50.08±0.54	50.14±0.34	48.3±1.36
dermatology	**97.59±0.34**	97.5±0	97.48±0.07	97.5±2.28	96.65±0.81
ecoli	**71.77±0.51**	71.69±0	69.82±1.54	64.43±1.5	61.15±4.05
flare	**60.99±0.46**	60.08±0	59.95±0.44	59.69±0.34	58.26±2.01
glass	**77.61±0.88**	69.9±0	64.63±1.46	62.27±1.56	70.63±3.96
hayes-roth	**87.45±0.51**	86.18±0	85.02±0.49	84.01±0.54	81.41±3.22
led7digit	70.85±0.43	70.8±0	**71.21±0.58**	70.66±0.38	67.87±2.74
lymphography	**81.47±0.31**	78.62±0	76.58±2.44	75.74±2.5	72.01±4.21
newthyroid	**96.54±0**	94.25±0	94.98±0.75	95.99±0.6	95.54±1.57
page-blocks	**92.05±0.25**	84.27±0	89.05±0.72	90.88±0.37	89.77±2.59
penbased	99.11±0.07	99.1±0	99.17±0.04	**99.2±0.03**	98.38±0.37
satimage	**89.73±0.1**	89.08±0	89.56±0.1	89.69±0.13	87.31±1.55
segment	98.03±0.09	97.58±0	98.11±0.1	**98.22±0.11**	97.16±0.53
shuttle	94.93±0.74	93.42±0	**95.51±1.28**	94.51±0.67	93.51±0.84
splice	**95.58±0.03**	95.07±0	94.79±0.26	94.69±0.17	87.66±2.68
thyroid	**96.16±0.37**	86.51±0	92.59±0.75	93.16±1.23	93.62±2.32
vehicle	**77.3±0.27**	75.95±0	76.62±0.67	76.97±0.7	71.99±1.32
vowel	**96.09±0.36**	95.56±0	96.06±0.8	95.45±0.31	93.94±1.89
wine	98.78±0.79	98.63±0	98.79±0.22	**98.99±0.18**	95.79±1.12
winequality-red	**40.26±0.41**	35.96±0	40.1±1.12	39.78±0.48	35.82±1.85
winequality-white	**44.67±0.5**	39.06±0	41.8±0.31	42±0.22	34.69±1.94
yeast	54.93±0.54	**55.95±0**	54.59±1.62	53.39±1.4	45.91±1.34
zoo	**90±0.82**	87.14±0	89.14±2.09	**90±0**	88.03±2.86
Average	**78.8±0.4**	76.94±0	77.34±0.83	77.11±0.74	74.23±2.2
Ranking(p = 2.46e−09)	1.61	3.24	2.89	2.78	4.48

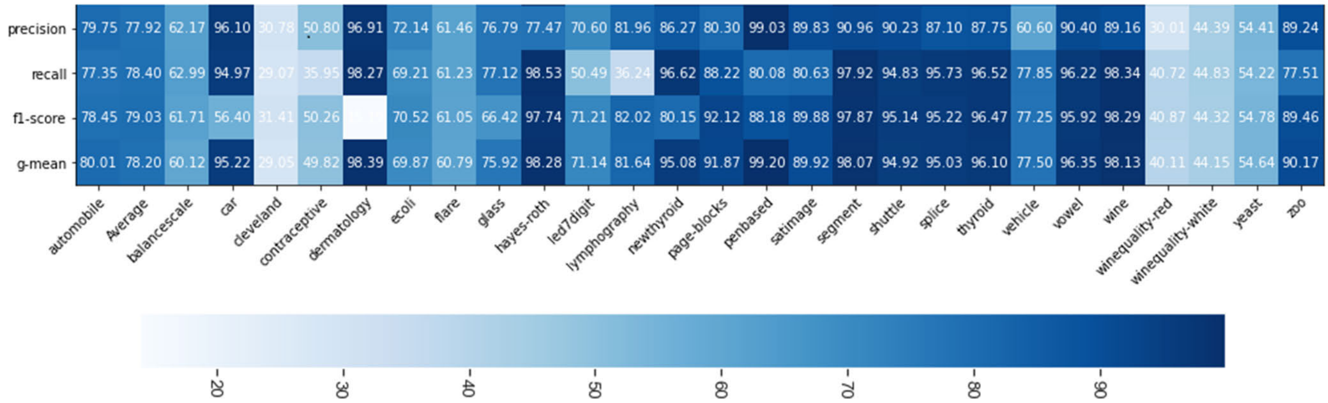


FIGURE 2. Results of different imbalanced learning methods using random forest with different metrics.

show that EPS performs well across most of the datasets. It has the highest average MAvA among all the methods at 76.94%, with a small standard deviation of 0.74%. It also has the lowest ranking ($p = 2.46e{-}09$) among all the methods at 1.61. Overall, EPS seems to perform well on these datasets when used with Random Forest classifier.

Figure 2 shows that the proposed method performs well for the majority of datasets, while it has lower performance for others. For instance, the proposed method achieves high accuracy, precision, recall, F1-score, and G-mean for datasets such as penbased, wine, segment, and car, while it has average performance for datasets such as cleveland, and yeast. It is also worth noting that some datasets have more consistent performance across the evaluation metrics than others. For example, the proposed method shows consistent performance across all evaluation metrics for the penbased dataset. In contrast, the performance for the zoo dataset varies significantly across different evaluation metrics.

The table 3 shows the mean average accuracy (MAvA) percentages of different imbalanced learning methods using SVM as a classifier. From the table, it can be seen that the proposed method has (EPS) the highest average MAvA among all the methods with 72.59%. OvA (Oversampling with Average) has the second-highest average MAvA with 72.24%. The

TABLE 3. Comparison of results of different imbalanced learning methods using SVM with MAvA metric.

Datasets	EPS	OvA	k-means-SMOTE	SMOTE	Bagging-RB
automobile	**59.4±1.3**	56.73±0	59.39±1.09	52.59±0.62	52.53±3.66
balancescale	**81.87±0.39**	70.33±0	81.21±1.07	79.58±1.11	76.58±3.63
car	**94.52±0.23**	91.83±0	92.66±1.15	92.34±0.8	92.28±2.19
cleveland	32.27±1.01	28.94±0	**36.23±1.92**	35.27±1.81	32.18±2.63
contraceptive	**56.1±0.22**	49.62±0	50.39±0.49	49.99±0.28	43.66±3.1
dermatology	97.15±0.1	96.85±0	97.25±0.18	**97.32±0.14**	96.48±0.66
ecoli	**71.89±0.15**	70.56±0	69.25±0.61	66.31±1.08	67.86±2.35
flare	**62.35±0.41**	58.92±0	59.21±0.85	59.47±0.23	57.54±2.26
glass	63.49±0.74	59.86±0	61.49±3.06	**64.11±1.75**	60.99±4.73
hayes-roth	**86.56±0.37**	86.06±0	60.27±2.5	58.02±1.11	77.55±4.91
led7digit	71.76±0.39	72.65±0	71.97±0.43	**73.81±0.6**	63.74±3.24
lymphography	76.73±2.37	74.35±0	**78.4±2.38**	77.49±2.32	71.08±5.52
newthyroid	94.33±0.49	94.22±0	87.76±1.67	**94.39±0.53**	95.84±2.22
page-blocks	**86.87±0.14**	71.93±0	79.53±2.25	75.19±0.67	83.32±3.33
penbased	99.37±0.01	**99.41±0**	99.31±0.15	97.99±0.02	99.21±0.11
satimage	87.1±0.05	87.49±0	**87.89±0.24**	82.85±0.09	85.20±1.53
segment	**93.78±0.07**	93.42±0	93.53±0.23	90.45±0.05	91.8±0.9
shuttle	90.03±0.05	77.89±0	93.8±0.04	**93.92±0.06**	82.65±0.09
splice	**85.54±0.06**	85.31±0	85.31±0.21	84.09±0.19	81.93±3.61
thyroid	**80.52±0.05**	64.87±0	59.78±1.12	61.07±0.6	75.52±3.9
vehicle	**78.15±0.28**	78.07±0	75.37±2.08	70.1±0.41	74.79±1.42
vowel	**93.59±0.17**	92.12±0	92.57±0.65	90.08±0.27	92.15±2.01
wine	97.52±0	97.52±0	98.35±0.69	**98.56±0.15**	97.67±0.81
winequality-red	35.92±0.38	28.82±0	**37.91±0.72**	36.51±0.8	32.41±2.11
winequality-white	**36.77±0.66**	25.5±01	31.52±0.43	30.45±0.41	30.08±1.87
yeast	**57.34±0.33**	54.9±0	54.48±0.71	48.21±1.01	55.57±1.29
zoo	91.75±0	91.75±0	**92.86±1.24**	90.2±1.72	88.04±1.78
Average	**76.39±0.39**	72.59±0	73.62±1.04	72.24±0.7	72.54±2.44
Ranking(p=9.71e−06)	1.89	3.57	2.46	3.22	3.85

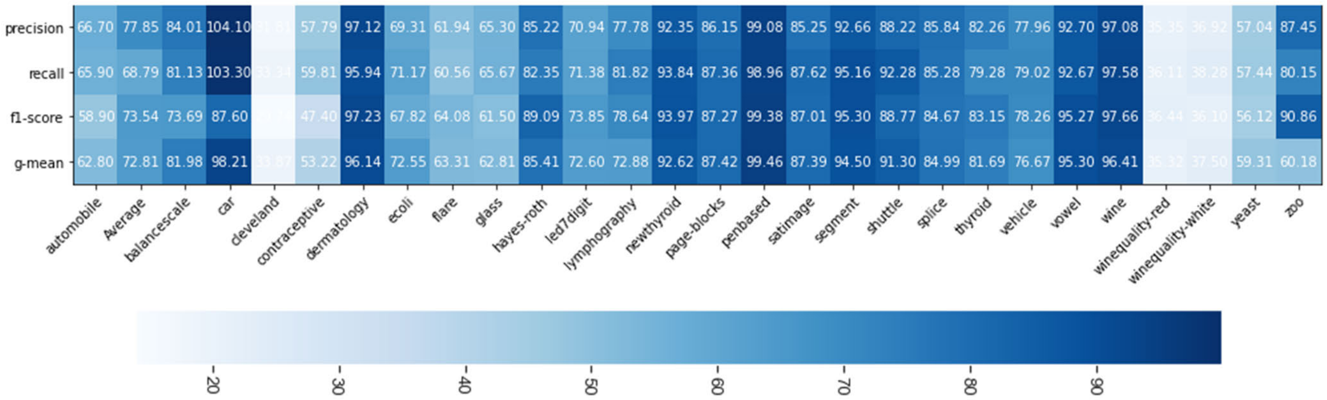


FIGURE 3. Results of different imbalanced learning methods using SVM with different metrics.

other methods, SMOTE (Synthetic Minority Over-sampling Technique), k-means-SMOTE, and Bagging-RB have average MAvA of 73.62%, 72.54% and 72.59% respectively.

Looking at the figure 3 we can observe that the proposed method performs well on some datasets, while on others it has poor performance. For example, the method achieves high accuracy on the 'penbased', 'segment', and 'dermatology' datasets, while it has poor accuracy on 'yeast' and 'zoo' datasets. Similarly, we can see that the proposed method has a high recall on 'newthyroid', 'segment', and 'lymphography' datasets, while it has poor recall on 'zoo' and 'cleveland' datasets. The proposed method has a high precision on 'penbased', 'dermatology', and 'wine' datasets, while it has poor precision on 'cleveland' and 'zoo' datasets. In addition, the proposed method has a high F1-score on 'newthyroid', 'segment', and 'lymphography' datasets, while it has poor F1-score on 'zoo' and 'cleveland' datasets.

The table 4 presents the results of a Holm-Bonferroni multiple comparison test, which is used to compare the performance of the proposed (EPS) method with four other methods (OvA, SMOTE, k-means-SMOTE, and Bagging-RB) using different machine learning algorithms (CART, Random Forest, and SVM). The corrected p-value column shows the significance level of the difference in performance between EPS and each of the other methods. A smaller p-value indicates that there is a higher likelihood that the difference in performance is not due to chance. A p-value of less than 0.05 is considered statistically significant. From the table, it can be seen that the p-values for all comparisons are less than 0.05, which suggests that there is a statistically significant difference in performance between EPS and the other methods across all the machine learning algorithms used. The lowest p-value is for EPS vs Bagging-RB(CART) with 1.3e-04, it suggest that this comparison is the most statistically significant.

TABLE 4. **Comparison of the proposed method (EPS) with other methods using holm-bonferroni test ($\alpha = 0.05$).**

Comparison	Corrected *p*-value (Holm–Bonferroni)
Bagging-RB(CART) # EPS	**1.3e−04**
Bagging-RB(Random Forest)# EPS	**4.0e−05**
Bagging-RB(SVM) # EPS	**6.9e−05**
k-means-SMOTE(CART) # EPS	**2.2e−05**
k-means-SMOTE(Random Forest) # EPS	**2.0e−03**
k-means-SMOTE(SVM) # EPS	**3.7e−02**
OvA(CART) # EPS	**2.2e−05**
OvA(Random Forest) # EPS	**6.9e−04**
OvA(SVM) # EPS	**2.7e−04**
SMOTE(CART) # EPS	**2.2e−05**
SMOTE(Random Forest) # EPS	**2.0e−03**
SMOTE(SVM) # EPS	**4.6e−03**

B. COMPARISON OF THE PROPOSED WITH MULTICLASS IMBALANCED CLASSIFICATION METHODS

In this section, we aimed to compare the effectiveness of our proposed method, the Ensemble Partition Sampling (EPS) with typical multiclass imbalanced classification methods. To do this, we selected three representative methods from existing literature, namely DES-MI [25], OvO-EASY and OvO-SMB [29]. These methods were chosen as they have been shown to be effective in solving multiclass imbalanced classification tasks on public datasets. In addition, we also included our proposed method, OvO-EPS, as a comparison. We used CART, Random Forest and SVM as the base classifiers for our experiments and we used the same parameters for the comparison methods as those used in the literature.

From the table 5 below, it can be seen that the EPS method performs well in multiclass imbalanced classification, with an average *MAvA* of 77.24% across all datasets. This is the highest among all the methods compared, and it also has the lowest standard deviation among them. This indicates that EPS is able to achieve high accuracy in a consistent manner across different datasets. Furthermore, the ranking of the methods based on the p-value (p=2.53e−08) indicates that EPS has the lowest rank of 1.63. This implies that the EPS method outperformed all other methods and is the most robust and consistent among them. Overall, the results suggest that EPS is a highly effective method for multiclass imbalanced classification when using CART as a classifier. On the other hand, the OvO-SMB method has the worst performance with an average *MA*. The figure 4 shows that the proposed method performs differently for different datasets and evaluation metrics. The highest performance is achieved for the 'penbased' dataset, where all metrics are above 97%. On the other hand, the lowest performance is achieved for the 'cleveland' dataset, where all metrics are below 34%.

The proposed method generally performs better for datasets that have high accuracy, precision, recall, and F1-score, such as 'dermatology', 'newthyroid', 'segment', 'shuttle', 'splice', 'thyroid', and 'wine'. On the other hand, the proposed method performs worse for datasets that have low accuracy, precision, recall, and F1-score, such as 'cleveland' and 'yeast'.

From the table 6 it appears that our proposed method performs well in multiclass imbalanced classification using a Random Forest classifier. It has the second highest average *MAvA* percentage, with an average of 77.78%. It also has the second lowest p-value in the ranking column, indicating that it is a statistically significant method compared to the other methods. The figure 5 shows that the proposed method performs relatively well across the different datasets and metrics. However, there are some notable variations in performance across datasets and metrics. Looking at the different datasets, the method achieves the highest scores on car, led7digit, segment, and thyroid datasets, with scores ranging from 69% to 98%. On the other hand, the method performs relatively poorly on the contraceptive, dermatology, and lymphography datasets, achieving scores ranging from 16% to 51%. For the different metrics, the method performs best in terms of precision and recall, achieving scores above 90% in most cases. The F1-score and g-mean also show relatively high scores, with the former ranging from 50% to 98% and the latter ranging from 60% to 98%. However, the accuracy metric shows a wider variation in scores, ranging from 30% to 99%. Overall, our proposed method appears to be a strong performer in this specific context.

From the table 7, it can be seen that the EPS method generally performs well, with high accuracy scores on most of the datasets. However, it is not the best performer on all datasets, for example, on the car dataset, OvO-SMB performs better, and on the dermatology dataset, OvO-EASY performs better. From the figure 6, we can see that the proposed method performs differently for different datasets and metrics. For instance, the proposed method performs well on the car dataset for all metrics, achieving an accuracy of 94.52% and a g-mean of 98.92%. On the other hand, the proposed method does not perform well on the shuttle dataset, achieving an accuracy of only 18.82%. Additionally, we can see that some metrics perform better than others for certain datasets. For example, in the dermatology dataset, the proposed method performs poorly in terms of precision (30.87%) but performs very well in terms of recall (94.06%). Overall, it can be concluded that the EPS method is an effective solution for multiclass imbalanced classification tasks.

TABLE 5. Comparison of results of multiclass imbalanced classification methods using cart with MAvA metric.

DATASETS	EPS	OVO-EPS	OVO-EASY	OVO-SMB	DES-MI
automobile	80.96±0.75	79.98±1.89	70.98±2.19	73.50±3.14	**82.42±2.26**
balancescale	58.1±0.44	55.82±0.91	**58.74±2.02**	57.09±0.85	55.92±1.2
car	**98.3±0.24**	97.43±0.56	97.85±0.6	96.87±0.57	96.67±0.6
cleveland	**31.59±1.63**	29.87±0.92	27.97±1	29.69±3.88	30.07±1.99
contraceptive	48.76±0.6	48.25±0.69	**50.53±0.53**	48.05±0.65	47.92±0.87
dermatology	94.18±0.43	92.53±0.41	93.12±0.75	93.34±0.48	**96.64±0.88**
ecoli	**59.49±2.32**	55.67±1.61	50.32±2.22	55.74±1.19	57.99±1.32
flare	62.27±0.55	62.06±1.02	**62.3±1.25**	59.84±0.73	60.87±0.87
glass	**71.22±1.73**	69.01±2.32	70.27±2.04	68.18±2.53	70.95±2.57
led7digit	70.52±0.59	69.73±0.09	62.1±1.01	51.68±1.79	**71.39±0.7**
lymphography	**83.04±0.76**	70.39±3.22	71.79±5.15	73.88±4.11	79.28±2.77
newthyroid	**94.03±0.55**	92.25±1.2	93.09±1.31	92.38±1.55	93.77±0.83
page-blocks	89.01±0.32	86.3±0.71	**94.12±0.45**	86.18±0.85	86.37±0.68
penbased	**98.68±0.05**	96.99±0.05	91.63±0.25	85.35±0.09	98.05±0.09
winequality-red	39.38±0.6	42.28±1.94	**42.79±2.76**	38.44±1.21	39.16±1.51
roth	86.25±1.2	85.95±0.33	84.75±0.94	86.15±0.26	**86.95±0.69**
satimage	**89.12±0.17**	86.93±0.2	86.94±0.28	84.17±0.4	88.37±0.29
segment	**97.68±0.1**	96.61±0	97.42±0	96.32±0	97.4±0.19
shuttle	**95.75±0.01**	93.31±0.01	95.11±0.01	93.38±0.02	93.77±0.01
splice	**93.85±0.2**	92.58±0.17	91.03±0.27	90.91±0.31	93.74±0.27
thyroid	95.07±0.83	**97.99±1.81**	97.66±1.04	87.81±1.97	90.58±1.89
vehicle	**74.67±0.39**	69.41±0.48	69.56±0.68	68.15±0.79	74.29±0.78
vowel	**89.62±0.46**	79.8±0.64	88.78±0	88.79±0	88.98±0.6
winequality-white	43.04±0.72	**43.82±0.74**	42.01±3.01	39.69±0.76	41.01±0.64
wine	94.19±0.6	94.31±0.67	93.35±1.08	93.44±0.83	**96.17±0.8**
yeast	54.13±0.67	**54.87±0.82**	54.82±1.26	50.95±1.34	53.4±1.06
zoo	**92.35±1.18**	92.14±1.99	89.3±1.92	89.79±2.78	92.29±1.17
Average	**77.23±0.67**	75.42±0.94	75.12±1.26	73.32±1.22	76.46±1.02
Ranking(p=2.53e−08)	1.63	3.41	3.11	4.26	2.59

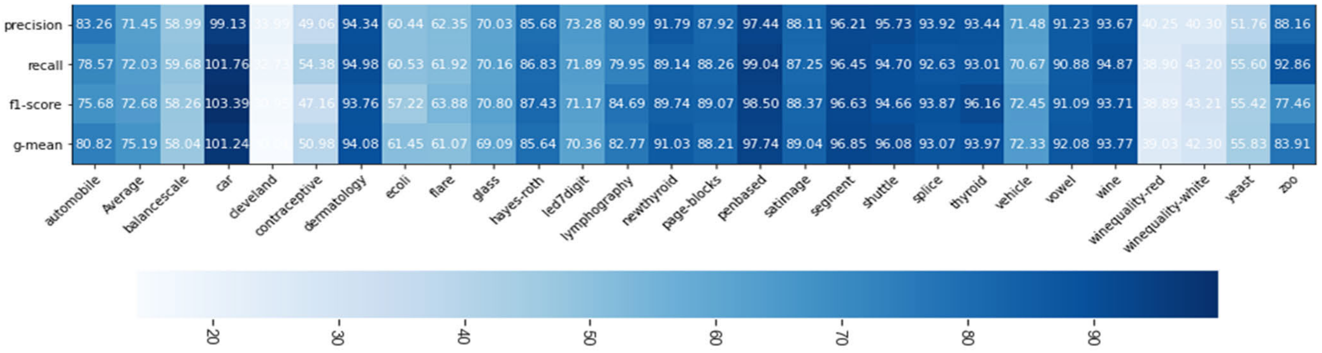


FIGURE 4. Results of multiclass imbalanced classification methods using CART with different metrics.

The Table 8 shows the results of a Holm-Bonferroni test for comparing the performance of the proposed method EPS method with three other methods (DES-MI, OvO-SMB, and OvO-EASY) and another variant of the EPS method (OvO-EPS) using three different machine learning algorithms (CART, Random Forest, and SVM). The corrected p-value is reported for each comparison.

For all comparisons using the CART algorithm, the corrected p-values are less than 0.05, indicating that there is a statistically significant difference in performance between the EPS method and the other methods or variant. Specifically, the p-value for the comparison between EPS and DES-MI is 0.01, indicating a stronger statistical significance than the other comparisons.

For the comparison using Random Forest algorithm, the corrected p-values are less than 0.05 for the comparisons with OvO-SMB and OvO-EASY, but greater than 0.05 for the comparisons with DES-MI and OvO-EPS, indicating that there is a statistically significant difference in performance between the EPS method and the OvO-SMB and OvO-EASY methods, but not with DES-MI and OvO-EPS.

For the comparison using SVM algorithm, the corrected p-values are all greater than 0.05, indicating that there is no

TABLE 6. Comparison of results of multiclass imbalanced classification methods using random forest with MAvA metric.

DATASETS	EPS	OVO-EPS	OVO-EASY	OVO-SMB	DES-MI
automobile	79.35±0.68	69.97±2.12	70.33±0.59	70.91±0.58	**80.64±0.95**
balancescale	**60.35±0.1**	58.08±0.58	59.36±2.38	59.73±0.68	57.28±0.98
car	95.64±0.21	**97.91±0.6**	97.41±0.57	94.42±0.67	96.84±0.44
cleveland	**29.87±0.92**	27.13±0.66	28.71±0.65	28.43±0.61	28.68±2.5
contraceptive	50.51±0.13	**51.21±0.43**	50.58±0.61	50.01±0.15	50.24±0.32
dermatology	**97.59±0.34**	97.53±0.22	93.39±0.26	97.41±0.73	96.85±0.4
ecoli	**71.77±0.51**	71.3±0.7	49.54±1.19	64.72±1.58	71.12±1.09
flare	60.99±0.46	62.22±0.51	**63.61±1.09**	58.39±0.75	60.77±0.81
glass	**77.61±0.88**	73.41±0.64	73.83±1.22	74.45±2.29	76.9±1.72
hayes-roth	**87.45±0.51**	85.3±0.56	84.18±0.73	84±0.59	86.18±0.92
led7digit	70.85±0.43	71.26±0.24	64.86±0.46	70.74±0.37	**71.8±0.42**
lymphography	81.47±0.31	77.19±0.3	69.64±4.56	72.53±2.99	**84.72±1.24**
newthyroid	96.54±0	**97.89±0.51**	94.73±1.2	94.98±1.02	96.29±1.08
page-blocks	92.05±0.25	92.14±0.28	**94±0.22**	89.09±0.7	89.38±0.82
penbased	99.11±0.07	98.88±0.02	90.75±0.07	98.88±0.03	**99.16±0.03**
satimage	89.73±0.1	89.92±0.04	87.06±0.07	89.72±0.19	**90.01±0.1**
segment	**98.03±0.09**	97.36±0	97.36±0	97.36±0	97.93±0.07
shuttle	94.93±0.74	93.57±0.93	**95.42±1.01**	95.13±0.96	93.1±0.82
splice	**95.58±0.03**	94.75±0.11	94.35±0.08	93.32±0.23	94.7±0.25
thyroid	96.16±0.37	95.94±0.78	**97.05±0.07**	90.78±0.93	95.9±1.25
vehicle	**77.3±0.27**	75.89±0.36	71.52±0.25	75.38±0.26	75.86±0.14
vowel	96.09±0.36	93.94±0.47	90.1±0	90.1±0	**97.43±0.24**
wine	**98.78±0.79**	98.63±0.33	92.81±0.52	98.77±0.27	98.72±0.39
winequality-red	40.26±0.41	41.7±0.65	**44.32±1.77**	41.54±0.37	40.51±1.45
winequality-white	44.67±0.5	**44.72±0.45**	40±1.61	42.63±0.19	43.7±0.38
yeast	54.93±0.54	51.99±0.35	55.82±0.35	52.57±1.22	**58.52±0.47**
zoo	90±0.82	90±0	**90.69±0.7**	90±1.28	89.43±0
Average	**78.8±0.4**	77.76±0.48	75.61±0.82	76.52±0.73	78.62±0.71
Ranking(p = 7.77e−04)	2.11	2.83	3.39	3.85	2.81

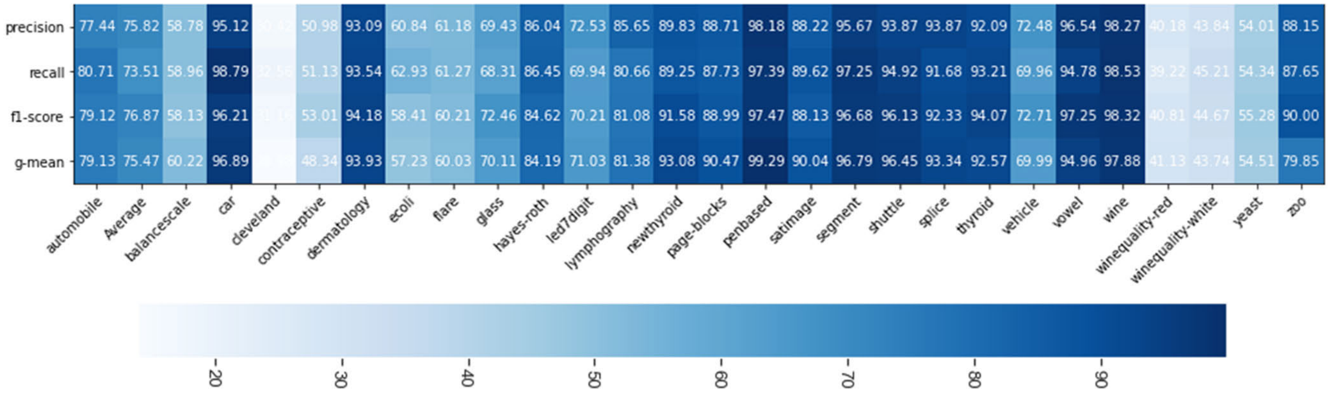


FIGURE 5. Results of multiclass imbalanced classification methods using Random Forest with different metrics.

statistically significant difference in performance between the EPS method and the other methods or variant.

In conclusion, the results suggest that the EPS method generally outperforms the other methods and variant when using the CART algorithm, but its performance is similar to the other methods and variant when using the Random Forest and SVM algorithms.

This study aimed to evaluate the effectiveness of the Ensemble Partition Sampling (EPS) method by comparing it to previously proposed methods in the literature. EPS was compared to OvA, SMOTE, k-means-SMOTE, and Bagging-RB using CART, Random Forest, and SVM as classifiers. The results demonstrated that EPS outperformed the other methods, with Bagging-RB, k-means-SMOTE, SMOTE, and OvA following in order. The study also compared EPS to typical multiclass imbalanced classification methods, including DES-MI, OvO-EASY, and OvO-SMB, and found that EPS performed well in multiclass imbalanced classification, achieving an average MAvA of 77.24% across all datasets. EPS reached 99% accuracy in many datasets, and other metrics, such as precision, F1-score, recall, and g-mean, also showed excellent results, exceeding 90% in several cases. A Holm-Bonferroni multiple comparison test was conducted to compare the performance of EPS to the

TABLE 7. Comparison of results of multiclass imbalanced classification methods using SVM with MAvA metric.

DATASETS	EPS	OVO-EPS	OVO-EASY	OVO-SMB	DES-MI
automobile	**59.4±1.3**	55.9±0.56	53.6±1.76	56.91±4.02	56.36±1.59
balancescale	81.87±0.39	80.52±0.58	**86.39±1.04**	82.50±2.42	80.93±0.62
car	94.52±0.23	95.55±0.67	**97.76±2.33**	87.68±2.06	97.25±0.32
cleveland	**32.27±1.01**	32.19±1.56	23.04±1.65	30.35±1.81	29.47±1.77
contraceptive	**56.1±0.22**	54.85±0.32	50.28±0.63	48.26±0.9	54.27±0.28
dermatology	**97.15±0.1**	96.56±0.44	93.45±4.19	95.82±2.85	96.86±0.26
ecoli	71.89±0.15	**72.16±0.44**	57.35±0.77	68.29±1.74	64.99±0.69
flare	62.35±0.41	60.76±0.46	62.25±1.03	62.04±2.23	**62.5±0.88**
glass	63.49±0.74	64.4±0.49	62.68±3.92	61.25±2.83	**66.54±1.66**
hayes-roth	86.56±0.37	84.73±1.02	**86.59±1.22**	85.07±2.58	83.45±1.06
led7digit	71.76±0.39	70.78±0.12	63.23±1.84	51.68±3.17	**72.53±0.62**
lymphography	76.73±2.37	76.68±0.72	**78.05±2.49**	77.13±3.96	64.39±0.49
newthyroid	94.33±0.49	93.78±0	93.47±1.25	92.6±2.87	**94.89±0.89**
page-blocks	86.87±0.14	84.91±0.23	84.13±0.54	72.27±0.63	**92.84±0.44**
penbased	99.37±0.01	**99.41±0.01**	98.74±0.11	85.91±0.69	99.35±0.02
satimage	87.10±0.05	88.44±0.06	86.99±0.52	77.95±1.54	**88.51±0.17**
segment	93.78±0.07	**94.07±0**	**94.07±0**	**94.07±0**	93.66±0.26
shuttle	90.03±0.05	87.65±0.06	65.06±0.26	**95.82±0.19**	92.06±0.24
splice	85.54±0.06	86.35±0.06	92.63±0.27	91.67±0.39	**94.05±0.16**
thyroid	80.52±0.05	70.86±2.5	**97.69±1.74**	85.64±5.8	78.05±1.98
vehicle	**78.15±0.28**	75.53±0.24	70.81±0.67	68.79±2.61	75.72±0.54
vowel	93.59±0.17	**95.66±0**	94.04±0	**95.66±0**	94.16±0.37
wine	97.52±0	**98.26±0.36**	93.69±6.36	94.67±2.86	97.94±0.3
winequality-red	35.92±0.38	36.17±0.69	**41.57±2.65**	38.56±2.49	40.25±1.18
winequality-white	36.77±0.66	33.89±0.44	**44.46±2.52**	39.43±1.02	41.29±0.34
yeast	**57.34±0.33**	56.49±0.25	54.29±1.01	54.8±0.84	55.62±0.48
zoo	91.75±0	84.94±1.72	87.99±3.68	**92.86±2.34**	88.32±2.02
Average	**76.39±0.39**	75.24±0.52	74.60±1.65	73.62±2.03	76.16±0.73
Ranking(p=0.16)	2.59	3.09	2.7	3.3	2.59

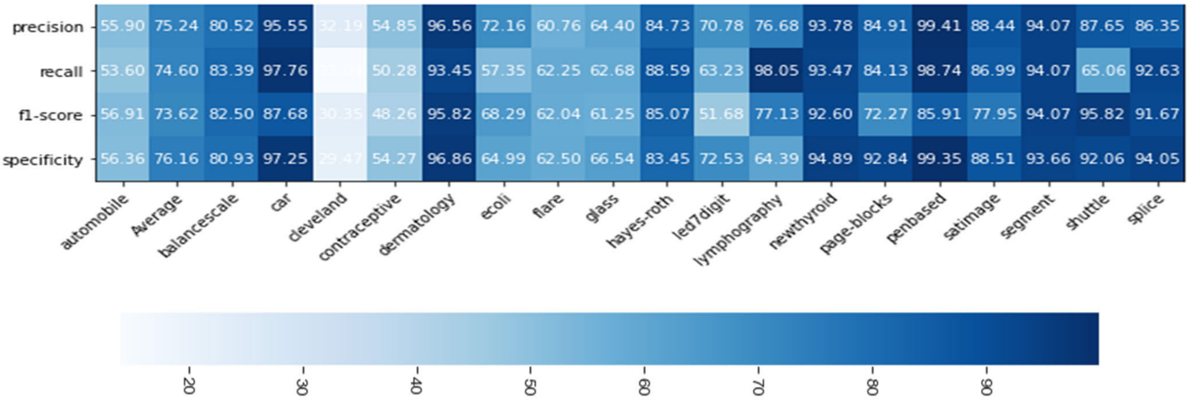


FIGURE 6. Results of multiclass imbalanced classification methods using SVM with different metrics.

TABLE 8. The results of a holm-bonferroni test for comparing the performance of EPS method with other methods.

COMPARISON	CORRECTED p-VALUE (HOLM–BONFERRONI)
DES-MI(CART) # EPS	**0.01**
OVO-SMB(CART) # EPS	**2.0E−04**
OVO-EASY(CART) # EPS	**0.02**
OVO-EPS(CART) # EPS	**5.1E−03**
DES-MI(RANDOM FOREST) # EPS	**0.26**
OVO-SMB(RANDOM FOREST) # EPS	**2.0E−04**
OVO-EASY(RANDOM FOREST) # EPS	**0.01**
OVO-EPS(RANDOM FOREST) # EPS	**0.07**
DES-MI(SVM) # EPS	**0.9**
OVO-SMB(SVM) # EPS	**0.16**
OVO-EASY(SVM) # EPS	**0.34**
OVO-EPS(SVM) # EPS	**0.23**

other methods, and the results indicated a statistically significant difference in performance between EPS and the other methods across all machine learning algorithms used. The tables and results figures highlighted this conclusion, proving that EPS's results were higher compared to other methods, with EPS's performance being the highest among all the methods compared.

IV. CONCLUSION

In this study, we proposed a new method called Ensemble Partition Sampling (EPS) for multiclass classification that utilizes ensemble learning and preprocessing techniques to address the problem of data imbalance. Our findings show that EPS outperforms other methods such as OvA, SMOTE, k-means-SMOTE, Bagging-RB, DES-MI, OvO-EASY, and OvO-SMB. Furthermore, our study contributes to the field of deep learning by presenting an innovative solution for multiclass classification that demonstrates improved results compared to existing methods.

Our study has highlighted the significance of ensemble learning and preprocessing techniques in improving the classification performance of imbalanced and multiclass imbalanced datasets. The use of EPS can help other researchers in the field of machine learning to tackle similar problems in their work. Future work can involve exploring the use of EPS in other domains and testing its robustness against larger and more complex datasets. While our proposed method of Ensemble Partition Sampling (EPS) shows promising results in improving classification performance in imbalanced and multiclass imbalanced datasets, there are still some limitations to our work.

One of the weaknesses of EPS is that it may not be as effective in extremely imbalanced datasets where one or more classes have significantly fewer instances compared to the majority class. This is because the minority classes may not have enough instances to generate a robust ensemble of classifiers, leading to lower accuracy and performance. Another limitation of our work is that we only used three classifiers, namely CART, Random Forest, and SVM. While these are popular classifiers in the field of machine learning, there may be other classifiers that could potentially outperform our proposed method. Further research can explore the effectiveness of EPS with a wider range of classifiers. Another limitation is the sensitivity of machine learning models to free parameters, which we did not optimize using an independent validation set in our study. Future research can address these limitations by exploring the use of EPS in other domains and testing its robustness against larger and more complex datasets.

ACKNOWLEDGMENT

The authors would like to thank the support of a team of distinguished experts who have made this research possible. This work is part of a larger scientific project aimed at improving algorithms of machine learning, and the authors are privileged to have the following institutions collaborating on this effort: This study is supported by a team of doctors from multiple institutions, which include the Department of Signal Theory and Communications, University of Valladolid, Spain, the LIMATI Laboratory, Sultan Moulay Slimane University, Morocco, the European University of the Atlantic, Spain, the International Iberoamerican University Arecibo, USA, and the University of la Romana, Dominican Republic.

REFERENCES

[1] T. Kam Ho, "The random subspace method for constructing decision forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 832–844, Aug. 1998, doi: 10.1109/34.709601.

[2] Y. Sun, Z. Liu, S. Todorovic, and J. Li, "Adaptive boosting for SAR automatic target recognition," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 1, pp. 112–125, Jan. 2007, doi: 10.1109/TAES.2007.357120.

[3] S. Lee, B. Kc, and J. Y. Choeh, "Comparing performance of ensemble methods in predicting movie box office revenue," *Heliyon*, vol. 6, no. 6, Jun. 2020, Art. no. e04260, doi: 10.1016/j.heliyon.2020.e04260.

[4] B. Lu, C. B. Moya, and G. Lin, "NSGA-PINN: A multi-objective optimization method for physics-informed neural network training," Mar. 2023, *arXiv:2303.02219*. Accessed: Apr. 5, 2023.

[5] A. E. Raftery and Y. Zheng, "Discussion: Performance of Bayesian model averaging," *J. Amer. Stat. Assoc.*, vol. 98, no. 464, pp. 931–938, Dec. 2003, doi: 10.1198/016214503000000891.

[6] E. Aguilar, B. Nagarajan, and P. Radeva, "Uncertainty-aware selecting for an ensemble of deep food recognition models," *Comput. Biol. Med.*, vol. 146, Jul. 2022, Art. no. 105645, doi: 10.1016/j.compbiomed.2022.105645.

[7] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, and F. Herrera, "An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes," *Pattern Recognit.*, vol. 44, no. 8, pp. 1761–1776, Aug. 2011, doi: 10.1016/j.patcog.2011.01.017.

[8] H. He and Y. Ma, Eds., *Imbalanced Learning: Foundations, Algorithms, and Applications*. Hoboken, NJ, USA: Wiley, 2013.

[9] R. Longadge and S. Dongre, "Class imbalance problem in data mining review," May 2013, *arXiv:1305.1707*. Accessed: Apr. 5, 2023.

[10] B. Krawczyk, "Learning from imbalanced data: Open challenges and future directions," *Prog. Artif. Intell.*, vol. 5, no. 4, pp. 221–232, Nov. 2016, doi: 10.1007/s13748-016-0094-0.

[11] A. Fanny and T. W. Cenggoro, "Deep learning for imbalance data classification using class expert generative adversarial network," *Proc. Comput. Sci.*, vol. 135, pp. 60–67, 2018, doi: 10.1016/j.procs.2018.08.150.

[12] M. Zheng, T. Li, R. Zhu, Y. Tang, M. Tang, L. Lin, and Z. Ma, "Conditional Wasserstein generative adversarial network-gradient penalty-based approach to alleviating imbalanced data classification," *Inf. Sci.*, vol. 512, pp. 1009–1023, Feb. 2020, doi: 10.1016/j.ins.2019.10.014.

[13] J. Zhai, J. Qi, and C. Shen, "Binary imbalanced data classification based on diversity oversampling by generative models," *Inf. Sci.*, vol. 585, pp. 313–343, Mar. 2022, doi: 10.1016/j.ins.2021.11.058.

[14] B. Mirza, D. Haroon, B. Khan, A. Padhani, and T. Q. Syed, "Deep generative models to counter class imbalance: A model-metric mapping with proportion calibration methodology," *IEEE Access*, vol. 9, pp. 55879–55897, 2021, doi: 10.1109/ACCESS.2021.3071389.

[15] S. Wang and X. Yao, "Multiclass imbalance problems: Analysis and potential solutions," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 4, pp. 1119–1130, Aug. 2012, doi: 10.1109/TSMCB.2012.2187280.

[16] G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue, and G. Bing, "Learning from class-imbalanced data: Review of methods and applications," *Expert Syst. Appl.*, vol. 73, pp. 220–239, May 2017, doi: 10.1016/j.eswa.2016.12.035.

[17] J. J. Rodríguez, J.-F. Díez-Pastor, Á. Arnaiz-González, and L. I. Kuncheva, "Random balance ensembles for multiclass imbalance learning," *Knowl.-Based Syst.*, vol. 193, Apr. 2020, Art. no. 105434, doi: 10.1016/j.knosys.2019.105434.

[18] Q. Li, Y. Song, J. Zhang, and V. S. Sheng, "Multiclass imbalanced learning with one-versus-one decomposition and spectral clustering," *Expert Syst. Appl.*, vol. 147, Jun. 2020, Art. no. 113152, doi: 10.1016/j.eswa.2019.113152.

[19] R. Rifkin and A. Klautau, "In defense of one-vs-all classification," *J. Mach. Learn. Res.*, vol. 5, pp. 101–141, Dec. 2004.

[20] D. P. Russo, K. M. Zorn, A. M. Clark, H. Zhu, and S. Ekins, "Comparing multiple machine learning algorithms and metrics for estrogen receptor binding prediction," *Mol. Pharmaceutics*, vol. 15, no. 10, pp. 4361–4370, Oct. 2018, doi: 10.1021/acs.molpharmaceut.8b00546.

[21] H. Abdi, "Holm's sequential Bonferroni procedure," in *Encyclopedia of Research Design*. Thousand Oaks, CA, USA: SAGE, 2010, doi: 10.4135/9781412961288.n178.

[22] E. Brookes, P. Vachette, M. Rocco, and J. Pérez, "U.S.-SOMO HPLC-SAXS module: Dealing with capillary fouling and extraction of pure component patterns from poorly resolved SEC-SAXS data," *J. Appl. Crystallogr.*, vol. 49, no. 5, pp. 1827–1841, Oct. 2016, doi: 10.1107/S1600576716011201.

[23] N. W. S. Wardhani, M. Y. Rochayani, A. Iriany, A. D. Sulistyono, and P. Lestantyo, "Cross-validation metrics for evaluating classification performance on imbalanced data," in *Proc. Int. Conf. Comput., Control, Informat. Appl. (IC3INA)*, Tangerang, Indonesia, Oct. 2019, pp. 14–18, doi: 10.1109/IC3INA48034.2019.8949568.

[24] R. M. Prabha and S. Sasikala, "A comprehensive analysis on multiclass imbalanced big data classification," in *Soft Computing and Signal Processing* (Advances in Intelligent Systems and Computing), vol. 1413, V. S. Reddy, V. K. Prasad, J. Wang, and K. T. V. Reddy, Eds. Singapore: Springer, 2022, pp. 315–325, doi: 10.1007/978-981-16-7088-6_28.

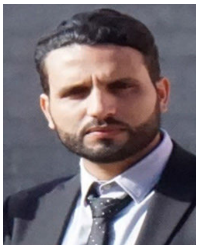
[25] S. García, Z.-L. Zhang, A. Altalhi, S. Alshomrani, and F. Herrera, ''Dynamic ensemble selection for multi-class imbalanced datasets,'' *Inf. Sci.*, vols. 445–446, pp. 22–37, Jun. 2018, doi: 10.1016/j.ins.2018.03.002.

[26] A. Fernandez, S. Garcia, F. Herrera, and N. V. Chawla, ''SMOTE for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary,'' *J. Artif. Intell. Res.*, vol. 61, pp. 863–905, Apr. 2018, doi: 10.1613/jair.1.11192.

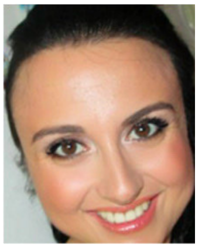
[27] Y. Chen and R. Zhang, ''Research on credit card default prediction based on *k*-means SMOTE and BP neural network,'' *Complexity*, vol. 2021, pp. 1–13, Mar. 2021, doi: 10.1155/2021/6618841.

[28] J. F. Díez-Pastor, J. J. Rodríguez, C. García-Osorio, and L. I. Kuncheva, ''Random balance: Ensembles of variable priors classifiers for imbalanced data,'' *Knowl.-Based Syst.*, vol. 85, pp. 96–111, Sep. 2015, doi: 10.1016/j.knosys.2015.04.022.

[29] Z.-L. Zhang, X.-G. Luo, S. González, S. García, and F. Herrera, ''DRCW-ASEG: One-versus-one distance-based relative competence weighting with adaptive synthetic example generation for multi-class imbalanced datasets,'' *Neurocomputing*, vol. 285, pp. 176–187, Apr. 2018, doi: 10.1016/j.neucom.2018.01.039.



BRAHIM JABIR received the degree in computer engineering from the Faculty of Sciences and Techniques, University Sultan Moulay Slimane, Beni Mellal, Morocco. He is currently a Professor in computer science with the Teacher Training College, Beni Mellal Khenifra. Since 2022, he has been the Leader of the LANDitic Research Group. He has also been responsible for the ICT Department, Teacher Training College. He is also an Associate Editor of the *International Journal of Information Technologies Systems* (IGI Global). His research interests include artificial intelligence (AI) and its various applications in different fields.



ISABEL DE LA TORRE DÍEZ is currently a Professor with the Department of Signal Theory and Communications and Telematics Engineering, University of Valladolid, Spain, where she is also the Leader of the GTe Research Group. Her research interests include design, development, and the evaluation of telemedicine applications, services and systems, e-health, m-health, electronic health records (EHRs), EHRs standards, biosensors, cloud and fog computing, data mining, the quality of service (QoS), and the quality of experience (QoE) applied to the health field.

ERNESTO FRANCISCO BAUTISTA THOMPSON received the master's degree in computer science from the National Autonomous University of Mexico (UNAM), and the Ph.D. degree in computer science from the European University of the Atlantic, in 2010. He is currently a renowned professor, a researcher, and a consultant with expertise in the fields of e-learning, data science, and information technology. He has made significant contributions to these fields through his research, teaching, and consulting work. He has worked at various institutions across the world, including the European University of the Atlantic, Spain, the International Iberoamerican University, Mexico, and the International University of Cuanza, Angola. He has held positions, such as a professor, a researcher, and a consultant, where he has contributed immensely to the advancement of e-learning, data science, and information technology.

DEBORA LIBERTAD RAMÍREZ VARGAS is a professor and postdoctoral researcher who has made significant contributions to the field of sustainability sciences, natural resource management, and fishing socio-ecological systems. She is affiliated with the Department of Engineering and Projects at the Universidad Internacional Iberoamericana in Campeche, Mexico. She is also associated with the Universidad Europea del Atlántico in Santander, Spain, and the Universidade Internacional do Cuanza in Cuito, Bié, Angola. She has previously held a scholarship from the Institute of Engineering at the National Autonomous University of Mexico (UNAM), where she obtained her Doctor of Engineering degree. She has also participated in various courses and workshops, ranging from Green Engineering to the conservation of Coastal Dunes. Notably, she attended a course-workshop on the simulation of natural and social phenomena (Net_Logo) at the Faculty of Engineering in Trento, Italy. Furthermore, she has taught subjects at the degree level at Mundo Maya University. Her research work primarily focuses on the processes of Coastal Oppression, Hydrological and Coastal Processes. Her publications on coastal oppression are characterized by the use of advanced mathematical methods, particularly in the context of fuzzy logic.

ÁNGEL GABRIEL KUC CASTILLA was born in Campeche, Mexico. He received the Ph.D. degree in engineering, with a specialization in marine and coastal engineering from the National Autonomous University of Mexico. He is affiliated with the Department of Engineering and Projects at the Universidad Internacional Iberoamericana in Mexico. Additionally, he holds affiliations with several esteemed institutions, including the International University Foundation of Colombia, Bogotá, Colombia, and the University of La Romana, La Romana, Dominican Republic. He is currently a professor and a researcher in the field of coastal engineering. His work focuses on the development of innovative solutions for the sustainable management of marine and coastal ecosystems. He has published numerous articles in top-tier journals, and his research has been funded by prestigious organizations, such as the National Science Foundation and the Mexican Council for Science and Technology.