Ensemble of Classifiers based on Multiobjective Genetic Sampling for Imbalanced Data

Everlandio R. Q. Fernandes, Andre C. P. L. F. de Carvalho, and Xin Yao

Abstract—Imbalanced datasets may negatively impact the predictive performance of most classical classification algorithms. This problem, commonly found in real-world, is known in machine learning domain as imbalanced learning. Most techniques proposed to deal with imbalanced learning have been proposed and applied only to binary classification. When applied to multiclass tasks, their efficiency usually decreases and negative side effects may appear. This paper addresses these limitations by presenting a novel adaptive approach, E-MOSAIC (Ensemble of Classifiers based on MultiObjective Genetic Sampling for Imbalanced Classification). E-MOSAIC evolves a selection of samples extracted from training dataset, which are treated as individuals of a MOEA. The multiobjective process looks for the best combinations of instances capable of producing classifiers with high predictive accuracy in all classes. E-MOSAIC also incorporates two mechanisms to promote the diversity of these classifiers, which are combined into an ensemble specifically designed for imbalanced learning. Experiments using twenty imbalanced multi-class datasets were carried out. In these experiments, the predictive performance of E-MOSAIC is compared with state-of-the-art methods, including methods based on presampling, active-learning, cost-sensitive, and boosting. According to the experimental results, the proposed method obtained the best predictive performance for the multiclass accuracy measures mAUC and G-mean.

Index Terms—Imbalanced datasets, ensemble of classifiers, evolutionary algorithm

1 Introduction

A Large number of real classification datasets present imbalanced class distribution, i.e., there are many more examples of some classes (majority classes) than others (minority classes). This imbalanced distribution occurs naturally in data from applications such as network intrusion detection, financial engineering, and medical diagnostics [1]. In such cases, imbalanced datasets can make many classical classification algorithms less effective, especially when predicting minority class examples. This is because most of the classical classification algorithms are designed to induce models that are able to generalize from the training data then return the simplest classification model that best fits the data. However, the simplest model pays less attention to rare cases, sometimes treating as noise [2] and the resulting classifier might lose its classification ability in this scenario.

The imbalanced learning problem is treated, in machine learning, in two distinct ways: at the data and the algorithm level [3]. However, most existing imbalanced learning techniques are only designed for and tested in two-class scenarios, i.e., binary datasets. Unfortunately, when a dataset with multiple classes are present, the literature solutions proposed for the binary case may not be directly applicable, or may achieve a lower performance than expected [4] [5]. In addition, a multiclass problem can have a different purpose. For example, in the binary case the researchers focus on the correct classification of the minority class, as the classifier is usually biased toward the majority class and the minority class is usually the most important. Datasets with several classes can have more than one main class, i.e., multiple classes that need to have a high degree of accuracy regarding the classifier.

A commonly strategy used to generate binary classification models when the training dataset is imbalanced is to select a balanced sample of the dataset. This means that the classes have the same number of examples. Thus, the model induced by this sample would not harm the minority class. Although this strategy be easily extended to multiclass classification problems, it may not be effective in some cases, as the generated classification model despises instances that are not part of the sample. Furthermore, the sample may not be truly representative. Such cases may lead to erroneous inferences or distort results, especially when the sample is randomly selected.

This approach raises important questions regarding classification in imbalanced datasets, like: which imbalance ratio of datasets really affects the predictive performance of classic learning algorithms? And, are all learning paradigms equally affected by class imbalance?

In [6] the authors present an extensive study with 22 binary datasets and seven learning algorithms from different paradigms. Given a database, part of the study is to generate several training set distributions with increasing degrees of class imbalance (50/50, 40/60, 30/70, 20/80, 10/90, 5/95 and 1/99). The 50/50 distribution represents a balanced distribution, 40/60 means that 40% of the instances in the dataset belongs to the minority class and 60% to the majority class, and so on. Next, the authors induce a classifier for each class distribution and compare its performance loss with the performance loss for the balanced distribution (50/50). According to the authors, most of the learning algorithms investigated had some degree of performance loss for every

- *E. Fernandes and A. de Carvalho are with the Instituto de Ciências Matemáticas e de Computação (ICMC) at University of São Paulo (USP), Brazil.*
- *X. Yao is with Southern University of Science and Technology Department of Compute Science and Engineering Shenzhen 518055, China.*

Manuscript received May 19, 2017; revised August 30, 2018.

non-balanced distribution. The losses start to be significant (5% or more) when the minority class represents at most 10% of the dataset. The study also shows that different learning paradigms are affected in different degrees by the class imbalance.

In opposition to the previous study, recently published studies have reported the successful use of ensembles of classifiers for classification with imbalanced datasets, where each classifier is induced by a different sample from the original dataset [7]. Ensembles are designed to increase the accuracy of a single classifier by separately inducing a set of hypotheses and combining their decisions using a consensus operator [8]. The generalization ability of an ensemble is usually higher than a single classifier. In [9] the authors present a formal demonstration of this. Although ensembles of classifiers tend to perform better than their members, their construction is not an easy task. According to [10], an ensemble with high accuracy implies two conditions: each base classifier has an accuracy higher than 50%; and they should be different from each other. Two classifiers are considered different from each other if their misclassifications are made at different instances in the same test set i.e., they should disagree as much as possible in their outcomes [11].

Therefore, *diversity* and *accuracy* are the two main criteria to be taken into account when generating an effective ensemble of classifiers. The literature has several examples were the use of diversity measures to select the base classifiers positively affects the ensemble predictive performance [12] [13]. Examples of diversity measures include Negative Correlation Learning (NCL) [14] and Pairwise Failure Crediting (PFC) [15].

Regarding the predictive accuracy of the base classifiers, a good accuracy in the minority classes is usually as important, or in some scenarios more important, than majority class accuracy. However, these learning objectives are usually in conflict; increasing the accuracy of some classes can result in lower accuracy in others. Multiobjective Evolutionary Algorithms (MOEA) can deal with this tradeoff, as they have been successfully worked with conflicting objectives in the learning process (e.g. predictive accuracy in each class). MOEA simultaneously evolve a set (or front) of non-dominated solutions over two or more objectives, without requiring the imposition of preferences on the objectives [16].

In this context, this paper proposes a new ensemble-based method, named E-MOSAIC (Ensemble of Classifiers based on Multiobjective Genetic Sampling for Imbalanced Classification), to deal with imbalanced multiclass classification tasks. For such, E-MOSAIC induced a set of classifiers from imbalanced datasets evolving balanced samples extracted from imbalanced datasets, guided by the class accuracy of the classifiers induced from these samples. It should be noted that this strategy allows the evolution of the samples, which may result in imbalanced samples, but which induce classifiers with high predictive accuracy for each class of the original dataset. In order to promote the diversity between the classifiers, the PFC diversity measure is used together with a process that eliminates similar solutions after crossover process. PFC is used as a secondary fitness that resolves tie issues in the selection process of the multiobjective genetic algorithm.

Important aspects in the E-MOSAIC and that differ it from the others genetic sampling methods for imbalanced classification is that the proposed approach does not have any mechanism to limiter the growing of amount of instances in each class. Balanced samples are randomly selected to form the initial population. This aim to eliminate the initial risk of some minority class of the dataset to receive less attention or to be treated as noise by leaner classifier. The combination of solutions in a ensemble of classifiers aims to reduce the loss of information inherent in the process of undersampling used to build the initial population. Experimental results for 20 multiclass imbalanced datasets from the UCI machine learning repository [17] show the advantages of the proposed approach over existing methods.

The remainder of this paper is structured as follows: Section 2 provides a review of related work. Section 3 explains the main ingredients of the E-MOSAIC approach. Section 4 shows the experimental analysis and Section 5 concludes the paper.

2 Related Works

In general, the classification of imbalanced datasets can be categorized into two primary levels: (*i*) the data level and (*ii*) the algorithm level. In the first, the objective is primarily to balance the class distribution [2], [5], [18] whereas, in the second, algorithms are adapted to increase the importance of instances from the minority class for model optimization [19], [20]. There are also other approaches that focus on feature selection or work at the ensemble level.

2.1 Data level approaches

Several works can be found in the literature regarding resampling techniques that study the effect of changing the class distribution in imbalanced datasets [18], [21]. All works show, empirically, that applying a pre-processing step to rebalance class distribution is frequently very useful. Techniques are usually classified as *oversampling* and *undersampling* strategies, or a mixture of both. In oversampling, the number of instances of the minority class is grown until it reaches the size of the majority class and, in undersampling, the opposite takes place.

Random oversampling (ROS), a non-heuristic method that add instances through random replication of a minority class, is one of the simplest approaches. Interpolation techniques such as the Synthetic Minority Oversampling Technique (SMOTE) [18] are commonly used to generate synthetic data. SMOTE finds the k nearest neighbors of each instance from the minority class, then synthetically generates new instances in the line that connects that instance to its k nearest neighbors.

Depending on how instances are created, oversampling techniques generally increase the probability of overlapping between classes. Some techniques have been proposed to minimize this drawback, such as the Modified Synthetic Minority Oversampling Technique (MSMOTE) [22] and Adaptive Synthetic Sampling (ADASYN) [23]. Another aspect to address is that the replication of instances tends to increase the computational cost of the learning process [21] and can

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2019.2898861, IEEE Transactions on Knowledge and Data Engineering

generate data that would not be found in the investigated problem.

Conversely, random undersampling (RUS) is a simple strategy employed to shrink the majority class. Although of simple use, it may despise useful data. In order to overcome this problem, directed undersampling aims to detect and eliminate less representative instances from the majority class. This is the strategy used by the One-sided Selection (OSS) technique [24] which attempts to remove redundant, noisy and/or, close to the boundary instances from the majority class. Border instances are detected by applying Tomek links and instances distant from the decision boundary (redundant instances) are discovered by Condensed Nearest Neighbor (CNN) [25]. The elimination of instances from the majority class close to the separation boundary is also handled by the Majority Under-sampling Technique (MUTE) [26], which defines security levels for each instance from the majority class and uses these to propose undersampling.

2.2 Algorithm level approaches

Solutions proposed at the algorithm level are based on adapting existing classification algorithms to improve the overall accuracy of the classifier and number of positive classifications (detection of instances from the minority classes) at the same time. There are two major categories in this method, the recognition-based and cost-sensitive approaches.

The One-class SVM method [27] is a recognition-based example that considers only one class of examples during the learning process in order to recognize (or rebuild) the class of interest. The support vector model in One-class SVM is trained on data that has only one class, which is the normal class. It infers the properties of normal cases and from these can predict which examples are unlike the normal examples. This is useful for imbalanced datasets because the scarcity of training examples is what excludes the rare cases.

A dynamic sampling method (DyS) for multilayer perceptions (MLP) was proposed in [28]. In DyS, for each epoch of the training process, every example is fed to the current MLP, then the probability of it being selected for training the MLP is estimated. The selection mechanism can allay the effects of class imbalance and pay more attention to examples that are difficult to classify.

As pointed out by [21], solutions at the algorithm level are usually specific to the particular algorithm and/or problem. Therefore, they are only effective in certain contexts and usually require expertise in classification algorithms and their field of application.

2.3 Ensemble approaches

In contrast to the common approaches of machine learning that try to build a hypothesis about the training data, the ensemble of classifiers technique constructs a set of hypotheses and combines them through some method/operator consensus [8]. The ability to generalize in an ensemble is generally greater than the isolated classifiers that compose it, usually called base-classifiers. In [9] a formal demonstration of this is presented. Methods based on committees are attractive because they are able to boost weak classifiers, and this is better than guessing which classifiers can make more accurate predictions [8].

In recent years, several ensemble learning methods have been proposed as possible solutions to the task of classification with imbalanced datasets [29] [30] [31] [32]. The proposed solutions are based on a combination of: ensemble learning techniques, some resampling methods, cost-sensitive methods or adaption of some existing classification algorithms. However, most of them have been developed only to address the problem of binary classification.

Most methods use some variation of Bagging [33] and Boosting [34]. In Bagging, a set of base classifiers are trained with different samples from the training dataset. Sampling is carried out with replacement and each sample has the same size as in the original dataset. After base classifiers are created, a combination of classifiers responses by majority voting is performed and new input instances are assigned to the most voted-for class. The AdaBoost method [34] is the most typical algorithm in the Boosting family. It uses the whole training dataset to create classifiers after several iterations. At each iteration, instances incorrectly classified at the previous iteration are emphasized and used to create new classifiers. After obtaining the base classifiers, when a new instance is presented, each base classifier yields its vote (weighted by its overall accuracy) and the label for the new instance is determined by majority voting.

Although ensembles of classifiers usually present predictive performance better than their individual counterparts, their constructing is not an easy task. Commonly, an ensemble of classifiers with high accuracy is advocated to have two main characteristics: each base classifier has to have accuracy higher than 50% and the base classifiers should present high diversity among themselves [10]. Two classifiers are considered diverse when they disagree as much as possible or, in other words, when generating different misclassifications for different instances of the same test set [11].

Several methods that take into account diversity and accuracy of base classifiers have been proposed. Multi-objective Genetic Sampling (MOGASamp) [35], which is designed to handle only binary dataset, constructs an ensemble of classifiers induced from balanced samples in the training dataset. For this, a customized multiobjective genetic algorithm is applied, combining instances from balanced samples and guided by the performance of classifiers induced by those samples. This strategy aims to obtain a set of balanced samples from the imbalanced dataset and induce classifiers with high accuracy and diversity.

In [29], the authors developed a Multiobjective Genetic Programming (MOGP) approach that uses accuracies of the minority and majority classes as competing objectives in the learning process. The MOGP approach is adapted to evolve diverse solutions into an ensemble, aiming at improving the general classification performance.

In [36], the authors investigate two types of multiclass imbalance problems, i.e., multi-minority and multi-majority. First, they investigate the performance of two basic resampling techniques when applied to these problems. They conclude that in both cases the predictive performance of the methods decreases when the number of imbalanced classes increases. Motivated by these results, the authors investigate the two more popular ensemble approaches (Adaboost and Bagging) combining them with class decomposition (the one-against-all strategy) and using resampling techniques. According to their experimental results, the use of class decomposition did not provide any advantages in multiclass imbalance learning.

3 The Proposed Method

The main objective of the proposed method is to build an ensemble of classifiers with high accuracy and diversity for imbalanced multiclass classification, named E-MOSAIC. These base classifiers are induced by optimized samples from imbalanced datasets, without the need of empirical studies that are normally required to find an optimal class distribution. E-MOSAIC uses a multiobjective genetic algorithm based on NSGA-II [37] to evolve a combination of balanced samples, each sample used to induce a base classifier, and evaluate the classifiers induced by these samples regarding the predictive accuracy for each class. Tie issues in the selection process are resolved by the PFC diversity measure. The use of this metric, together with a mechanism to eliminate similar solutions after the crossover process, aim to promote the creation of diverse solutions in the evolutionary process.

Figure 1 outlines the proposed method, which is detailed in the next sections.

3.1 Sampling and the Training Models

First, n balanced samples are obtained from the training dataset. This means that each sample has the same number of instances of each class. The sample size is chosen based on the number of instances of the class that has fewer instances in training dataset, i.e., the most minority class. However, only 90% of the instances of the most minority class are used to compose the samples. Despite the small number of instances of the minority class in some datasets, this percentage was chosen to not compromise the diversity of the samples regarding the minority class.

Consider a dataset with 3 classes and 50 instances at the most minority class as an example. The sample size will be $0.9 * 50 * 3$, i.e., 45 instances of each class. Thus, 2 samples may be different with respect to the most minority class by up to 5 instances i.e., 11.11% of the sample. On the other hand, considering the majority classes, this difference can reach 100% depending on the number of instances of the majority classes.

Each sample represents an individual in the population of the Genetic Algorithm (GA). These samples are encoded by a binary vector where each cell represents one instance of the training dataset. The bits ”1” and ”0” indicate selected and ignored instances, respectively. After the sampling process, an MLP model is generated for each individual. This induction uses only the instances flagged as ”1” (selected) in the sample.

3.2 Fitness Evaluation

In order to evaluate each individual, the predictive model obtained by training a MLP network is validated using the entire training dataset. The predictive accuracy of this model for each class is estimated using the PPV metric (positive predictive value). The PPV of a classifier c with respect to a class i is calculated according to Equation 1.

$$PPV_{c,i} = \frac{\#true\_positives_i}{\#true\_positives_i + \#false\_positives_i} \quad (1)$$

where $\#true\_positives_i$ is the number of times the model correctly classifies instances from class i, and $\#false\_positives$ indicates the number of times the model classifies instances that are not from class i as belonging to this class. In this evaluation approach, these metrics are used as competing objectives in the learning process. Therefore, each individual is associated with the PPVs of its classification model.

The initial samples will be balanced, so the classifiers induced using these samples will not be affected by class imbalance. Since part of the examples from the majority classes will not be in these samples, only a part of the original dataset will be used and the predictive accuracy (PPV) for the minority classes will be overestimated, e.g., will be close to their accuracy if there were no imbalance.

A main aspect in the multiobjective genetic algorithm is the concept of Pareto Dominance [38]. In Pareto Dominance, a solution x_1 is said to dominate the other solution x_2, if the solution x_1 is no worse than x_2 in all objectives, and x_1 is strictly better than x_2 in at least one objective [39]. This allows individuals to be ranked according to their performance on all the objectives with regard to all individuals in the population. Based on this, the accuracies associated with each individual are used to compose a nondominance rank of the solutions. Nondominance rank [40] is a common Pareto-based dominance metric that calculates the number of other solutions in the population that dominate a given solution. So, a non-dominated solution will have a best fitness of 0, while high fitness values indicate poor-performing solutions, i.e., solutions dominated by many individuals.

However, without an explicit objective of diversity in the evolutionary process to encourage optimized samples to produce classifiers that make different errors in different inputs, there is no guarantee of the diversity of the classifiers produced by the optimized samples. Therefore, E-MOSAIC incorporates a diversity of classifiers measure as secondary objective in the evolutionary process. The PFC diversity measure is used in this approach because of its good results with imbalanced classification presented in [29] and fernandes2015 and because it shows more compliance with the performed search than does the crowding distance metric used by NSGA-II. This is because the crowding distance is calculated taking into account the values of the objectives used in the evolutionary algorithm (i.e. predictive accuracy in each class), giving preference to solutions that are more distant from the others in the objective space. However, the PFC indicates the diversity of the classification model associated with an individual in relation to the other models of the population and we are looking for more diverse classification models, aiming at constructing an effective ensemble of classifiers.

PFC is calculated for each individual using a pair-wise comparison with all individuals of the current population.

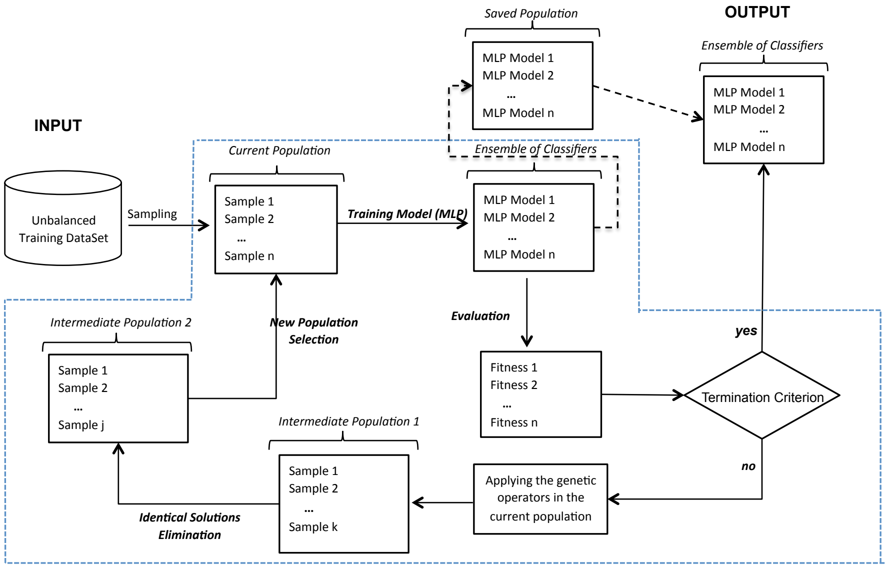


Fig. 1: E-MOSAIC - Ensemble of Classifier based on Multiobjective Genetic Sampling for Imbalanced Classification

The metric is used into the E-MOSAIC as a secondary fitness measure that resolves tie issues in the selection process (i.e., to apply the genetic operators of crossover/mutation and to build the next generation refer to the next step). This means that if two or more individuals have the same nondominance rank, the individual with the higher PFC is preferred. Solutions with higher PFC indicate that their nearest neighbors are far apart; these are preferred to smaller distance values.

3.3 Selection and Genetic Operators

Nondominance rank is used to select individuals that will breed a new generation using the genetic operators (reproduction and mutation). This selection is performed using a tournament of size 3. If a tie occurs, we consider the winner to be the one with the highest PFC. The quantity of parents selected will be equal to the quantity of individuals in the current population.

For each selected pair of parents, two new individuals are generated using the *one-point crossover* technique [41]. One-point or single-point crossover is a simple and frequently used method for genetic algorithms that selects a single crossover point on both parents' vectors and all data beyond this point, in either parent, is swapped between the two parents. The resulting vectors are the children. Mutation occurs in a percentage of generated offspring. The bits of a random portion of the vector that represents an individual are inverted.

Another important aspect is that from this point the number of instances of each class in the sample is no longer limiting. So, if after the crossover and mutation processes one sample is imbalanced, but it presets higher fitness than the other samples, it will be selected for the next generation.

3.4 Elimination of Identical Solutions

After applying the genetic operators, identical individuals can occur, especially when the imbalance ratio is not high. This fact was analyzed during our experimental tests. Identical individuals with high fitness have a higher probability of being selected for reproduction and for future generations, thereby increasing the number of identical solutions. However, the goal of this work is to have a diverse ensemble of classifiers with high accuracy. For this reason, after the reproduction stage, identical individuals are eliminated. After this elimination, if the number of individuals is less than the initial population size, new reproduction and mutation processes are performed.

3.5 New Generation and Stop Criterion

Selection of the individuals that comprise the new generation is based on the nondominance rank of each individual. First, individuals with higher levels of non-dominance are selected, then only those who are not dominated by the first, and so on, until the default population size is reached. The composition of the ensemble tries to mitigate the loss of information inherent to the sampling process, thus different classifiers may have different views of the dataset. This is encouraged by the mechanisms of diversity included into E-MOSAIC.

It is worth mentioning that even using elitism; there is no guarantee that the resulting ensemble of these individuals will have higher predictive accuracy than the ensemble from the previous generation. The reason is that even if the predictive accuracy of the models continues to improve over successive generations, the diversity between models can stagnate or even decrease, hampering the ensemble's predictive performance.

For this reason, in the initial population and after each generation, the classification models of all individuals in the current generation comprise an ensemble of classifiers representing the generation. This ensemble is evaluated based on the entire training dataset, and two accuracy measures are extracted from this evaluation, namely G-mean [30] and mAUC [42]. At first, the initial population and its accuracy measures (G-mean and mAUC) are saved

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2019.2898861, IEEE Transactions on Knowledge and Data Engineering

as *"Saved Population."* After each generation the G-mean and mAUC of the current population are compared with the metrics of the *"Saved Population."* If the current ensemble of classifiers presents improvement in their G-mean or mAUC and none of them are any worse, the current population replaces the *"Saved Population."*

The process stops after a fixed number of generations or after 5 generations without any replacement of the *"Saved Population"* or when Gmean or mAUC metrics reach their maximum value, i.e. max. G-mean = 1.0 and max. mAUC = 1.0. The classification models of all individuals in the final *"Saved Population"* compose the ensemble of classifiers. When a new example is presented to the classifiers, the class of this example is determined by the majority vote considering the output of each classifier.

4 Experimental Study

In this section, we present an empirical analysis of E-MOSAIC, including comparison with other approaches proposed for classification from imbalanced datasets. The experiments include a number of imbalanced datasets obtained from the UCI Machine Learning Database Repository [17]. The goal of these experiments is to verify whether E-MOSAIC actually offers some advantage in terms of overall performance and its effect during the learning process. The comparisons also allow us to determine the individual strengths and weaknesses of the proposed method compared to other existing approaches.

4.1 Compared Methods

A recent study [43] suggests that more elaborate methods of classification with imbalanced datasets do not have better performance than simple methods, such as ROS and RUS. Furthermore, E-MOSAIC incorporates an undersampling technique in its process, so it was first compared to the preprocessing methods. ROS and RUS be employed separately or used simultaneously to make a balanced dataset with the same number of instances as the original dataset. This method was also employed in our experiments and will be referred to as random fixed-size sampling (RFS) from now on. In addition, we applied no-sampling (NoS), in which the original training set without any resampling process was used to provide a baseline for our comparisons.

In addition to the data level approaches cited above, the performance of E-MOSAIC was compared with some algorithm level solutions and ensemble learning methods based on multiclass classification with imbalanced datasets found in the literature. DyS [28] is a recent method, closely related to active learning and boosting-type algorithms, for multiclass classification with imbalanced datasets. In the same study the authors presented MLP-based active learning (AL). Both methods were used in our experimental study.

For comparison with cost-sensitive learning, the minimization of misclassification cost (MMC) [44] and Rescale$_{\text{new}}$ [45] were chosen. Stagewise Additive Modeling using a Multiclass Exponential loss function (SAMME) [46] is a method that directly extends the AdaBoost algorithm to the multiclass case, but it was originally developed with decision trees as the base-classifiers. In order to make the comparison fairer, the SAMME was modified to use MLP as base-classifiers.

4.2 Metrics

When the task is to evaluate a classifier over imbalanced domains, classical ways of evaluating, such as overall accuracy, do not make sense. A standard classifier may ignore the importance of the minority classes because their representation inside the dataset is not strong enough. A typical example of this in a binary-class case is as follows: if the ratio of imbalance presented in the dataset is 1:100, the error of ignoring this class is only 1%. An effective metric for evaluating the performance of a classifier is the rate of classification errors made in each class [47]. Single-class performance measures evaluate how well a classifier performs in one class. However, the goal is to achieve good prediction in all classes. Therefore, it is necessary to combine individual metrics, as they are not useful when used alone.

The Receiver Operating Characteristic (ROC) curve [48] shows the relationship between the benefits and classification costs, in relation to the distribution of the data. So, we say that one classification model is better than another if its ROC curve dominates the other. When it is necessary to encode the ROC curve into single scalar value, the strategy is calculating the Area Under the ROC Curve (AUC) [49], which has been widely used to evaluate the performance of classifiers. Originally, AUC is only applicable to binary-class datasets. However, Hand and Till [42] extended AUC to multiclass problems and proposed a metric, called M, for multiclass classification problems (MAUC).

Furthermore, to evaluate the classification performance in detail, an extended version of the *Geometric Mean (G-mean)* [50] proposed by Sun, Kamel and Wang (2014) [30] will be employed as another performance metric in our experimental study. The G-mean metric to evaluate the performance of multiclass classifiers is defined in [30] as

$$G - mean = \left(\prod_{i=1}^{m} \frac{tr_i}{n_i} \right)^{\frac{1}{m}} \tag{2}$$

where m is the number of classes, n_i is the number of examples in class i, and tr_i is the number of correctly classified examples in class i.

4.3 Experimental Setup

In order to compare the performance of the proposed method with the other methods used in this experimental study, 20 datasets were obtained from the UCI Machine Learning Database Repository [17]. The basic characteristics of the datasets are presented in Table 1, including the number of features (#F), number of classes (#C), total of instances in the dataset (#Inst.) and class distribution.

The first 18 datasets are originally multiclass imbalanced datasets, but the numbers of classes are not very large. So, the letter-recognition dataset, which has 26 classes, was used to form two imbalanced datasets by randomly removing examples of some classes. The characteristics of the two resulting datasets are also presented in Table 1 (*Letter-1 and Letter-2*).

Dataset	#F	#C	#Inst.	Class Distribution
Abalone	8	18	4139	15: 57: 115: 259: 391: 568: 689: 634: 487: 267: 203: 126: 103: 67: 58: 42: 32: 26
Arrhythmia	259	7	416	245: 44: 15: 15: 25: 50: 22
Balance-scale	4	3	625	49: 288: 288
Car	6	4	1728	1210: 384: 65: 69
Chess	6	18	28056	2796: 27: 78: 246: 81: 198:471: 592: 683: 1433: 1712: 1985: 2854: 3597: 4194: 4553: 2166: 390
Contraceptive	9	3	1473	629: 333: 511
Dermatology	34	6	358	112: 61: 72: 49: 52: 20
Ecoli	6	5	327	143: 77: 35: 20: 52
Glass	9	4	192	70: 76: 17: 29
New-thyroid	5	3	215	150:35:30
Nursery	8	4	12958	4266: 4320: 328: 4044
Page-blocks	10	5	5473	4913: 329: 28: 88: 115
Satellite	36	6	6435	1533: 703: 1358: 1508: 626: 707
Soybean	35	17	661	20: 20: 20: 88: 44: 20: 20: 92: 20: 20: 20: 44: 20: 91: 91: 15: 16
Splice	60	3	3190	767: 768: 1655
Thyroid-allhypo	27	3	3770	3481: 194: 95
Thyroid-allrep	27	4	3772	3648: 38: 52: 34
Thyroid-ann	21	3	7200	166: 368: 6666
Letter-1	16	26	19221	10: 766: 736: 805: 768: 775: 773: 734: 755: 747: 739: 761: 792: 783: 753: 803: 783: 758: 748: 796: 813: 764: 752: 787: 786: 734
Letter-2	16	26	984	10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 10: 734

TABLE 1: Basic Characteristics of The Datasets (#F: The Number of Features, #C: The Number of Classes, #Inst.: The Total Number of Instances)

All the methods in this experimental study use, or were adapted to use, MLP as a base classifier and the *back-propagation* algorithm [51] was used to train the MLP. The parameters of MLP used here are the same used in [28] and are shown in Table 2, including the number of hidden nodes (#Hid. Nodes) and the number of training epochs (#Epoch). In addition to the values shown in Table 2, the learning rate was set to 0.1.

The E-MOSAIC and SAMME are methods that return an ensemble of classifiers. They need an input that informs the number of base classifiers that is returned from the learning process to comprise the ensemble. This parameter also means that E-MOSAIC will have 30 individuals in the population of the multiobjective genetic algorithm as each individual induces a classifier and all classifiers are used to compose the ensemble. In addition, being a method based on genetic algorithms, E-MOSAIC also has to set the reproduction and mutation rates of its reproduction process. The mutation rate was set at 0.1, this means that 10% of new individuals created by the reproduction process undergo the mutation process as explained in Subsection 3.3. Regarding the reproduction process, each pair of selected parents generates two new individuals. So the reproduction rate is 100%. The number of individuals generated in each generation is equal to the size of population, i.e., 30 individuals.

Dataset	#Hid. Nodes	#Epoch
Abalone	20	500
Arrhythmia	5	100
Balance-scale	15	500
Car	20	200
Chess	20	200
Contraceptive	15	200
Dermatology	2	1000
Ecoli	5	200
Glass	10	2000
New-thyroid	4	200
Nursery	20	100
Page-blocks	20	100
Satellite	15	100
Soybean	10	100
Splice	5	100
Thyroid-allhypo	10	200
Thyroid-allrep	10	100
Thyroid-ann	20	100
Letter-1	10	1000
Letter-2	10	1000

TABLE 2: Parameters for MLP

The results are reported after ten executions of each method using 5 trials of stratified 5-fold cross-validation. In this procedure, the original dataset is divided into 5 non-intersected subsets, each of which maintains the original class imbalance ratio. For each fold, each algorithm is trained with the examples of the remaining folds, and the prediction accuracy rate of the induced model tested on the current fold is considered to be the model predictive performance [31] [30].

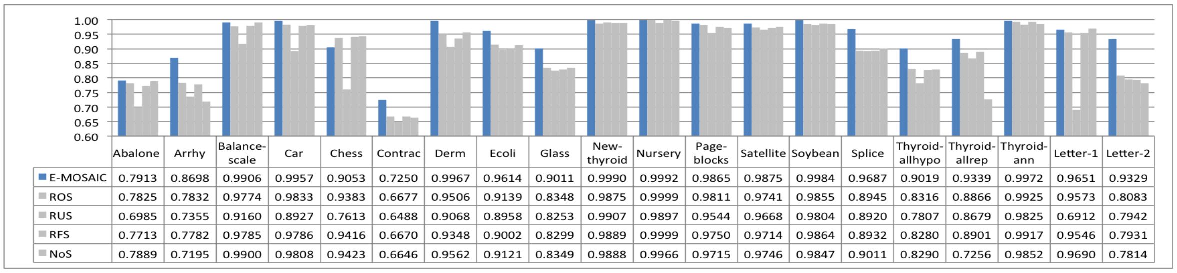
4.4 Experimental Results

4.4.1 Comparison with Data Level Methods

Figures 2 and 3 present, respectively, the average values for MAUC and G-mean obtained by E-MOSAIC, ROS, RUS, RFS and NoS for each dataset used. For each dataset, these figures also present a bar chart illustrating the comparative performance of the methods. The bars that represent the proposed method (blue bar) are highlighted, to evidence the difference between its performance and the performance obtained by the other methods.

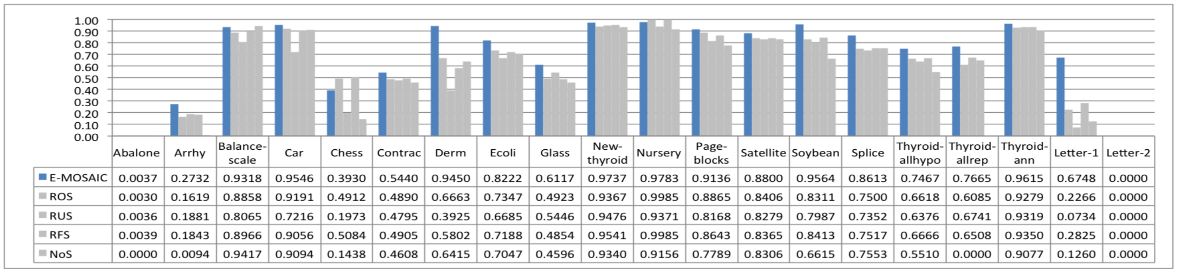
In order to provide some reassurance about the validity and non-randomness of the obtained results, we carried out statistical tests following the approach proposed by Demšar [52]. In brief, this approach seeks to compare multiple algorithms on multiple datasets, and is based on the Friedman test with a corresponding post-hoc test. The Friedman test is a non-parametric counterpart of the well-known ANOVA. If the null hypothesis, which states that the classifiers under study present similar performances, is rejected, then we proceed with the Nemenyi post-hoc test for pairwise comparisons.

According to the bar charts (Figures 2 and 3), E-MOSAIC outperforms the other methods in most datasets, presenting the best overall predictive performance. The ranking provided by the Friedman test supports this assumption, showing E-MOSAIC as the best-ranked method for MAUC and G-mean metrics. The Friedman test also indicates the rejection of the null-hypothesis, i.e., there is a statistically



	Abalone	Arrhy	Balance-scale	Car	Chess	Contrac	Derm	Ecoli	Glass	New-thyroid	Nursery	Page-blocks	Satellite	Soybean	Splice	Thyroid-allhypo	Thyroid-allrep	Thyroid-ann	Letter-1	Letter-2
E-MOSAIC	0.7913	0.8698	0.9906	0.9957	0.9053	0.7250	0.9967	0.9614	0.9011	0.9990	0.9992	0.9865	0.9875	0.9984	0.9687	0.9019	0.9339	0.9972	0.9651	0.9329
ROS	0.7825	0.7832	0.9774	0.9833	0.9383	0.6677	0.9506	0.9139	0.8348	0.9875	0.9999	0.9811	0.9741	0.9855	0.8945	0.8316	0.8866	0.9925	0.9573	0.8083
RUS	0.6985	0.7355	0.9160	0.8927	0.7613	0.6488	0.9068	0.8958	0.8253	0.9907	0.9897	0.9544	0.9668	0.9804	0.8920	0.7807	0.8679	0.9825	0.6912	0.7942
RFS	0.7713	0.7782	0.9785	0.9786	0.9416	0.6670	0.9348	0.9002	0.8299	0.9889	0.9999	0.9750	0.9714	0.9864	0.8932	0.8280	0.8901	0.9917	0.9546	0.7931
NoS	0.7889	0.7195	0.9900	0.9808	0.9423	0.6646	0.9562	0.9121	0.8349	0.9888	0.9966	0.9715	0.9746	0.9847	0.9011	0.8290	0.7256	0.9852	0.9690	0.7814

Fig. 2: Mauc Data Level Methods



	Abalone	Arrhy	Balance-scale	Car	Chess	Contrac	Derm	Ecoli	Glass	New-thyroid	Nursery	Page-blocks	Satellite	Soybean	Splice	Thyroid-allhypo	Thyroid-allrep	Thyroid-ann	Letter-1	Letter-2
E-MOSAIC	0.0037	0.2732	0.9318	0.9546	0.3930	0.5440	0.9450	0.8222	0.6117	0.9737	0.9783	0.9136	0.8800	0.9564	0.8613	0.7467	0.7665	0.9615	0.6748	0.0000
ROS	0.0030	0.1619	0.8858	0.9191	0.4912	0.4890	0.6663	0.7347	0.4923	0.9367	0.9985	0.8865	0.8406	0.8311	0.7500	0.6618	0.6085	0.9279	0.2266	0.0000
RUS	0.0036	0.1881	0.8065	0.7216	0.1973	0.4795	0.3925	0.6685	0.5446	0.9476	0.9371	0.8168	0.8279	0.7987	0.7352	0.6376	0.6741	0.9319	0.0734	0.0000
RFS	0.0039	0.1843	0.8966	0.9056	0.5084	0.4905	0.5802	0.7188	0.4854	0.9541	0.9985	0.8643	0.8365	0.8413	0.7517	0.6666	0.6508	0.9350	0.2825	0.0000
NoS	0.0000	0.0094	0.9417	0.9094	0.1438	0.4608	0.6415	0.7047	0.4596	0.9340	0.9156	0.7789	0.8306	0.6615	0.7553	0.5510	0.0000	0.9077	0.1260	0.0000

Fig. 3: G-Mean Data Level Methods

significant difference between the algorithms (MAUC: $p\text{-}value = 8.1349 \times e^{-14}$, G-mean: $p\text{-}value = 1.6887 \times e^{-10}$). Hence, we executed the Nemenyi post-hoc test for pairwise comparison. The proposed method outperforms all the data level methods on MAUC, measured with statistical significance at a 95% confidence level, except for the ROS method, which had a statistical significance at a 90%. Regarding the G-mean metric, the proposed method overcomes the ROS, RUS and NoS methods with statistical significance at a 95% confidence level.

We observe from Figure 2 that when comparing by the values of MAUC, E-MOSAIC outperforms other methods on most datasets (17 datasets). Only on three datasets, the proposed method does not achieve the best MAUC but it did not show the worst performance in any of them. E-MOSAIC is able to perform better overall due to its ability to induce an ensemble of classifiers with different views of the dataset. Also, due to the balanced way that the samples are collected and treated during the evolutionary process, the models of classification are generated in order to not harm the minority classes.

A similar situation can be seen in the Figure 3. On only a few datasets do the proposed method does not reach the highest G-mean value and none of them has the lowest value. However, in some datasets, such as *Abalone*, *Arrhythmia*, *Chess* and *Letter-2*, the G-mean value for all methods is very low (< 0.5). G-mean is the geometric mean of the classification accuracy of every class. Thus, poor accuracy of even one class will lead to poor G-mean. Therefore, a low value of G-mean value indicates that the classifier cannot effectively classify at least one class, which makes it less useful in practice.

4.4.2 Comparison with Algorithm Level Methods

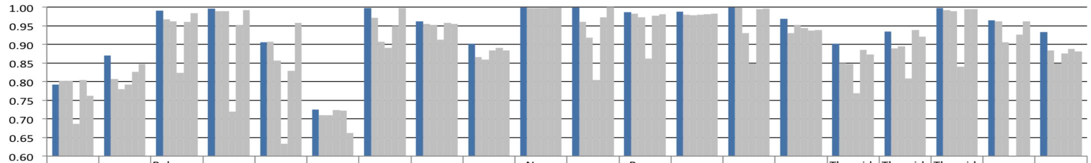
As in the previous subsection, Figures 4 and 5 show the average of MAUC and G-mean metrics, respectively, obtained by E-MOSAIC, DyS, AL, MMC, Rescale and SAMME methods in each dataset used here. Similarly, associated to each dataset, the figures also present a bar chart representing the comparative performance of the methods, the proposed method is highlighted by the blue bar. Table 3 shows the number of wins, draws and losses achieved by E-MOSAIC in a pairwise comparison with the algorithm level methods.

Metrics	**Methods**				
	DyS	**AL**	**MMC**	**Rescale**	**Samme**
MAUC	18-0-2	19-0-1	20-0-0	18-0-2	17-0-3
G-Mean	15-1-4	19-1-0	18-1-1	15-1-4	16-1-3

TABLE 3: Number of win-draw-lose between E-MOSAIC and the algorithm-level compared methods.

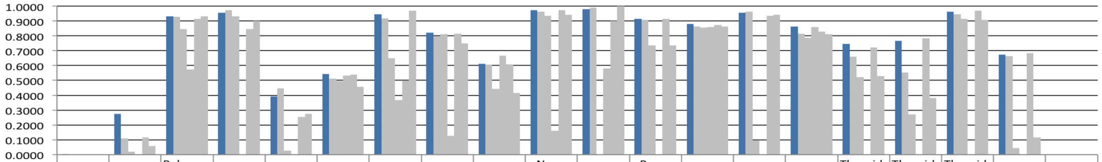
The results of statistical tests, following the methodology proposed by Demšar [52], suggest that E-MOSAIC achieved the best overall performance. The ranking provided by the Friedman test supports this assumption, indicating E-MOSAIC as the best-ranked method for MAUC and G-mean. The Friedman test also indicates the rejection of the null-hypothesis, i.e., there is a statistically significant difference among the algorithms (MAUC: $p\text{-}value = 1.6688 \times e^{-15}$, G-mean: $p\text{-}value = 2.3263 \times e^{-11}$). The application of Nemenyi post-hoc test revealed that the proposed method outperforms the Dys, AL, MMC and Rescale methods on MAUC metric with statistical significance at a 95% confidence level. Considering the G-mean metric the post-hoc test indicated that E-MOSAIC outperforms the AL, MMC, and SAMME methods with statistical significance at a 95% confidence level.

Initially, we turn our attention to methods based on active learning, i.e., DyS and AL. Comparing the results shown in Figure 4 and Table 3, referring to MAUC, E-MOSAIC outperforms DyS on 18 datasets and the AL on 19 datasets, there are not draws. With regards to G-mean metric (Figure 5), the proposed method overcomes DyS on 15 datasets and AL on all datasets, with the exception of



	Abalone	Arrhy	Balance-scale	Car	Chess	Contrac	Derm	Ecoli	Glass	New-thyroid	Nursery	Page-blocks	Satellite	Soybean	Splice	Thyroid-allhypo	Thyroid-allrep	Thyroid-ann	Letter-1	Letter-2
E-MOSAIC	0.7913	0.8698	0.9906	0.9957	0.9053	0.7250	0.9967	0.9614	0.9011	0.9990	0.9992	0.9865	0.9875	0.9984	0.9687	0.9019	0.9339	0.9972	0.9651	0.9329
DyS	0.8015	0.8067	0.9669	0.9888	0.9073	0.7093	0.9708	0.9548	0.8663	0.9972	0.9603	0.9817	0.9800	0.9981	0.9307	0.8491	0.8892	0.9916	0.9617	0.8836
AL	0.8022	0.7796	0.9611	0.9889	0.8570	0.7102	0.9067	0.9521	0.8586	0.9974	0.9173	0.9721	0.9787	0.9302	0.9520	0.8468	0.8950	0.9896	0.9054	0.8487
MMC	0.6859	0.7925	0.8241	0.7195	0.6331	0.7234	0.8909	0.9130	0.8831	0.9974	0.8046	0.8615	0.9792	0.8485	0.9439	0.7685	0.8090	0.8395	0.5445	0.8761
Rescale	0.8038	0.8265	0.9610	0.9522	0.8286	0.7227	0.9500	0.9583	0.8902	0.9986	0.9724	0.9771	0.9809	0.9951	0.9373	0.8853	0.9390	0.9942	0.9259	0.8873
Samme	0.7617	0.8467	0.9831	0.9923	0.9582	0.6612	0.9980	0.9550	0.8838	0.9989	1.0000	0.9807	0.9819	0.9957	0.9382	0.8728	0.9206	0.9945	0.9618	0.8805

Fig. 4: Mauc Algorithm Level Methods



	Abalone	Arrhy	Balance-scale	Car	Chess	Contrac	Derm	Ecoli	Glass	New-thyroid	Nursery	Page-blocks	Satellite	Soybean	Splice	Thyroid-allhypo	Thyroid-allrep	Thyroid-ann	Letter-1	Letter-2
E-MOSAIC	0.0037	0.2732	0.9318	0.9546	0.3930	0.5440	0.9450	0.8222	0.6117	0.9737	0.9783	0.9136	0.8800	0.9564	0.8613	0.7467	0.7665	0.9615	0.6748	0.0000
DyS	0.0000	0.1104	0.9295	0.9716	0.4455	0.5112	0.9176	0.7974	0.6044	0.9639	0.9913	0.9085	0.8642	0.9608	0.8146	0.6591	0.5533	0.9451	0.6621	0.0000
AL	0.0000	0.0197	0.8470	0.9316	0.0279	0.4932	0.6504	0.8122	0.4440	0.9336	0.0000	0.7355	0.8554	0.0957	0.7878	0.5215	0.2703	0.9160	0.0433	0.0000
MMC	0.0000	0.0000	0.5750	0.0000	0.0000	0.5319	0.3666	0.1263	0.6670	0.1628	0.5798	0.0000	0.8586	0.0000	0.8579	0.0000	0.0000	0.0000	0.0000	0.0000
Rescale	0.0000	0.1170	0.9159	0.8448	0.2527	0.5385	0.4980	0.8159	0.6094	0.9711	0.9009	0.9144	0.8734	0.9348	0.8275	0.7201	0.7828	0.9690	0.6847	0.0000
Samme	0.0000	0.0566	0.9319	0.9017	0.2758	0.4553	0.9688	0.7502	0.4152	0.9428	0.9997	0.7350	0.8617	0.9408	0.8118	0.5296	0.3802	0.9083	0.1175	0.0000

Fig. 5: Gmean Algorithm Level Methods

Letter-2 where the G-mean values of E-MOSAIC, DyS and AL are 0.

Methods based on active learning select informative examples for training a classifier through some criterion, such as based on the distance from the decision hyperplane to the example. This decision criterion can suffer the influence of several factors, which complicates the selection of the parameters for the algorithms, sometimes requiring the aid of an expert in the data. E-MOSAIC tends to have better performance than these methods because the example selection process is embedded in the evolutionary process of the genetic algorithm, i.e., the selected sample to induce a classifier will be modified during the process, guided by the classifier performance and not by some pre-established factor, which may not be the best decision for all datasets.

MMC [44] is based on cost-sensitive methods. As with most of these methods, it needs a cost matrix to operate properly. The cost matrix used in these experiments was formulated the same as the one used in [44]. The results reached by MMC are presented in Figures 4 and 5, referring to MAUC and G-mean metrics, respectively. As we can see, MMC has in most cases obtained the bars with the lowest heights, indicating that this method has the worst results compared to other algorithm-level methods. The ranking provided by the Friedman test supports this assumption, indicating MMC as the worst-ranked method on MAUC and G-mean metrics. The probable reason is that this method is very dependent on the cost matrix formulation and when the dataset is imbalanced, the cost matrix should be adjusted for this kind of problem [28]. In practice, this is a very hard task, requiring deeper knowledge of the dataset or a trial and error process.

The following compares E-MOSAIC and another cost-sensitive method. *Rescaling* is possibly the most popular approach to cost-sensitive learning. In [45] the authors published a study using a rescaling approach to multiclass problems (referred here as Rescale$_{new}$) and it was also applied to pure class imbalanced problems. The results obtained by this method are presented in the ”Rescale” rows of the tables embedded in Figures 4 and 5, referring to the MAUC and G-mean metrics respectively. As we can see in Table 3, E-MOSAIC outperforms Rescale on 18 datasets and is outperformed by Rescale on only 2 datasets when comparing in terms of MAUC. In terms of G-mean, E-MOSAIC outperforms Rescale on 15 datasets, is outperformed by Rescale on 4 datasets and ties with Rescale on Letter-2 dataset where the G-mean values of both are 0.

Boosting [34] has been widely used for solving binary classification problems. In [45], the authors presented an extension of boosting technique for multiclass classification, named SAMME. In that study, the authors also performed experiments with imbalanced datasets, getting good results. For this reason and because SAMME is an ensemble-based method we compared the proposed method with it. MLPs with the same parameters as those given in Table 2 were used as the base classifiers for SAMME. The number of classifiers was set to 30, the same amount of classifiers used in the proposed method. The ”SAMME” rows of the tables embedded in Figures 4 and 5 refer to the results obtained for MAUC and G-mean metrics, respectively.

In Table 3, observing the comparison between E-MOSAIC and SAMME in terms of MAUC, we can see that E-MOSAIC outperforms SAMME on 17 datasets and is outperformed by SAMME on 3 datasets, there are no draws. In terms of the G-mean metric, the proposed method outperforms SAMME on 16 dataset, is outperformed by SAMME on 3 datasets and there is a tie on the Letter-2 dataset where the G-mean value of both is 0. Of the methods

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2019.2898861, IEEE Transactions on Knowledge and Data Engineering

used in our experimental study, SAMME most resembles the proposed method because of the amount of base-classifiers generated during the training process. However, SAMME generates a new classifier increasing focus on examples that were wrongly classified in the previous iteration. On the other hand, the aim of the proposed method is to find and optimize the selection of samples from the training data so that each classification model generated for these samples has high predictive accuracy and they are dissimilar as possible.

Moreover, E-MOSAIC produces and validates an ensemble of classifiers at each iteration (generation), and the one that has greater predictive accuracy with respect to the training data is the ensemble of classifiers resultant of the training process. On the other hand, SAMME generates a new classifier at each iteration with low dependence on previously generated classifiers without considering the resultant ensemble of classifiers. This is an important feature of E-MOSAIC as it is possible to generate an ensemble of classifiers that get results lower than its single classifiers [53].

4.5 Further Analysis

From the results above, we can conclude that E-MOSAIC outperforms the other methods on most datasets. However, on some datasets the proposed method did not achieved the best result of all the methods used. That happens particularly when the number of instances of the most minority class is very low in comparison with the one of the most majority class, such as the *Chess* dataset, i.e., when the undersampling technique is not a good option due the large amount of information lost by discarding instances of the majority classes.

The number of instances of the most minority class is part of the main process of E-MOSAIC. It defines the sample size that represents each individual in the population of the genetic algorithm. So, if there are very few instances of the most minority class the sample size will be proportional to this number. The problem is that a very small sample may not contain adequate representation of the dataset and thus induces classifiers with low overall accuracy. E-MOSAIC overcomes this problem on most of datasets inducing an ensemble of classifiers with different views of datasets and combining their decisions.

However, on few datasets, such as the *Chess* dataset, this does not seem to be enough to reach the best overall accuracy of all methods used in the experiments. One possible explanation is that the sample size is not large enough to contain a good representation of the majority classes. If this is true, the classifiers returned by the proposed method in that situation would have higher accuracy for the minority classes than for the majority classes. In order to verify this situation, the classification accuracy of each class was calculated for each classifier returned by E-MOSAIC over the 10 times five-fold cross validation on the studied datasets. Table 4 shows the classes, the amount of instances of each class (#Instances), and the Positive Predictive Value (PPV) for each class of the *Chess*, *Glass*, *Car*, and *Conceptive* datasets used here.

Comparing the class distribution of the datasets shown in Table 4, we can observe that the classification accuracies of smaller classes are usually higher than larger classes, particularly when the number of instances of the most minority class is very low. Taking the *Chess* dataset as an example, the most minority class (*zero* Class) has a PPV of 0.6, which is almost twice that reached on the most majority class (*fourteen* Class) at is 0.3768. This difference lessens when the disproportion between the most minority and the most majority classes is not so high. An example of this is the *Contraceptive* dataset, which has class distributions of 629: 333: 511 and the PPV of each class is 0.5644, 0.6084, and 0.4731, respectively.

Obviously, other factors may interfere with these results, such as the overlapping level between the classes. However, increasing the number of instances of classes with low predictive accuracy in the sample, so that the sample does not get a high level of imbalance, could improve the results in the majority classes without harming the minority classes, and therefore improve the overall accuracy. This will be studied in depth in our future work.

Chess			Glass		
Class	#Instances	PPV	Class	#Instances	PPV
draw	2796	0.2154	1	70	0.7093
eight	1433	0.3774	2	76	0.5391
eleven	2854	0.2494	3	17	0.8062
fifteen	2166	0.5110	4	29	0.9109
five	471	0.5161			
four	198	0.7076			
fourteen	4553	0.3768			
nine	1712	0.2942			
one	78	0.7007			
seven	683	0.2411			
six	592	0.4599			
sixteen	390	0.8090			
ten	1985	0.1366			
thirteen	4194	0.3070			
three	81	0.6493			
twelve	3597	0.2140			
two	246	0.5547			
zero	27	0.6000			
Car			Contraceptive		
Class	#Instances	PPV	Class	#Instances	PPV
acc	384	0.7616	1	629	0.5644
good	69	0.9454	2	333	0.6084
unacc	1210	0.8614	3	511	0.4731
vgood	65	0.9800			

TABLE 4: Accuracy for each class returned by E-MOSAIC on *Chess*, *Glass*, *Car* and *Conceptive* datasets.

5 Conclusion

In this paper we presented a new modeling approach, called E-MOSAIC (Ensemble of Classifiers based on Multiobjective Genetic Sampling for Imbalanced Classification), to address the problem of classification with multiclass imbalanced datasets. This approach is based on a multiobjective genetic algorithm and produced an ensemble of classifiers. For this, a customized MOEA evolved combinations of instances in balanced samples, guided by the performance of the classifiers induced by these samples for each class. In addition, the multi-objective fitness function incorporates a PFC diversity measure, which aims to encourage the diversity of classifiers from the learning process. In this way E-MOSAIC produces a set of classifiers with high accuracy and diversity. Then, the obtained classifiers are used as an ensemble of classifiers to predict new instances using majority votes.

Extensive experiments on 20 multiclass imbalanced datasets from the UCI machine learning repository showed that E-MOSAIC outperforms other relevant methods in most cases, including presampling, active learning, cost-sensitive, and boosting-type methods. In a few datasets the proposed method did not achieve the best result of the methods used in the experiments, though none of them showed the worst results.

In a further analysis we investigated such occurrences and identified that the proposed method may be harmed when the number of instances of the most minority class is low. This is because the size of samples that generate base-classifiers depends on the number of instances of that class. The problem is that very small samples may not contain proper representation of the dataset that affect, in this case, the majority classes. A possible solution would be to increase the number of instances of classes with low predictive accuracy in the samples. This will be studied in depth in our future work.

Although a MLP has been used as base classifier in this paper, the general idea of E-MOSAIC can be extended to any other learning algorithm, along with the measures of accuracy and diversity used in the fitness function of the genetic algorithm. Other, more recent, multiobjective evolutionary algorithms can also be used by E-MOSAIC, like Bi-Criterion Evolution [54] and Two_Arch2 [55]. The investigation of these algorithms is also part of the future work.

ACKNOWLEDGMENTS

The authors would like to thank FAPESP, CNPq, CAPES and Intel for their financial support.

REFERENCES

[1] A. I. Marqués, V. García, and J. S. Sánchez, "On the suitability of resampling techniques for the class imbalance problem in credit scoring," *JORS*, vol. 64, no. 7, pp. 1060–1070, 2013. [Online]. Available: http://dx.doi.org/10.1057/jors.2012.120

[2] Y. Sun, M. S. Kamel, A. K. C. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recogn.*, vol. 40, no. 12, pp. 3358–3378, Dec. 2007. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2007.04.009

[3] T. Deepa and M. Punithavalli, "An analysis for mining imbalanced datasets," *International Journal of Computer Science and Information Security*, vol. 8, pp. 132–137, 2010.

[4] A. Fernndez, V. Lpez, M. Galar, M. J. del Jesus, and F. Herrera, "Analysing the classification of imbalanced datasets with multiple classes: Binarization techniques and ad-hoc approaches," *Knowledge-Based Systems*, vol. 42, pp. 97 – 110, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950705113000300

[5] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 18, no. 1, pp. 63–77, Jan. 2006. [Online]. Available: http://dx.doi.org/10.1109/tkde.2006.17

[6] R. C. Prati, G. E. Batista, and D. F. Silva, "Class imbalance revisited: a new experimental setup to assess the performance of treatment methods," *Knowledge and Information Systems*, pp. 1–24, 2014. [Online]. Available: http://dx.doi.org/10.1007/s10115-014-0794-3

[7] M. Galar, A. Fernández, E. B. Tartas, H. B. Sola, and F. Herrera, "A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches." *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, vol. 42, no. 4, pp. 463–484, 2012.

[8] Z.-H. Zhou, "Ensemble learning." in *Encyclopedia of Biometrics*, S. Z. Li and A. K. Jain, Eds. Springer US, 2009, pp. 270–273. [Online]. Available: http://dblp.uni-trier.de/db/reference/bio/e.html#Zhou09

[9] K. Tumer and J. Ghosh, "Analysis of decision boundaries in linearly combined neural classifiers," *Pattern Recognition*, vol. 29, pp. 341–348, 1996.

[10] T. G. Dietterich, "Machine-learning research – four current directions," *AI MAGAZINE*, vol. 18, pp. 97–136, 1997.

[11] A. Krogh and J. Vedelsby, "Neural network ensembles, cross validation, and active learning," in *Advances in Neural Information Processing Systems*. MIT Press, 1995, pp. 231–238.

[12] L. I. Kuncheva and C. J. Whitaker, "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy," *Machine Learning*, vol. 51, no. 2, pp. 181–207, May 2003. [Online]. Available: https://doi.org/10.1023/A:1022859003006

[13] R. Lysiak, M. Kurzynski, and T. Woloszynski, "Optimal selection of ensemble classifiers using measures of competence and diversity of base classifiers," *Neurocomputing*, vol. 126, pp. 29 – 35, 2014, recent trends in Intelligent Data Analysis Online Data Processing. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S092523121300698X

[14] Y. Liu and X. Yao, "Negatively correlated neural networks can produce best ensembles," *Australian Journal of Intelligent Information Processing Systems*, vol. 4, no. 3/4, pp. 176–185, 1997.

[15] A. Chandra and X. Yao, "Ensemble learning using multi-objective evolutionary algorithms," *J. Math. Model. Algorithms*, vol. 5, no. 4, pp. 417–445, 2006. [Online]. Available: http://dx.doi.org/10.1007/s10852-005-9020-3

[16] K. Lwin, R. Qu, and G. Kendall, "A learning-guided multi-objective evolutionary algorithm for constrained portfolio optimization," *Applied Soft Computing*, vol. 24, pp. 757 – 772, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1568494614003913

[17] K. Bache and M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[18] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[19] J. R. Quinlan, "Improved estimates for the accuracy of small disjuncts," *Machine Learning*, vol. 6, no. 1, pp. 93–98, 1991.

[20] B. Zadrozny and C. Elkan, "Learning and making decisions when costs and probabilities are both unknown," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2001, pp. 204–213.

[21] Y. Sun, A. K. C. Wong, and M. S. Kamel, "Classification of imbalanced data: a review," *IJPRAI*, vol. 23, no. 4, pp. 687–719, 2009. [Online]. Available: http://dx.doi.org/10.1142/S0218001409007326

[22] S. Hu, Y. Liang, L. Ma, and Y. He, "Msmote: Improving classification performance when training data is imbalanced," in *Computer Science and Engineering, 2009. WCSE '09. Second International Workshop on*, vol. 2, Oct 2009, pp. 13–17.

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

[24] M. Kubat and S. Matwin, "Addressing the curse of imbalanced training sets: One-sided selection," in *In Proceedings of the Fourteenth International Conference on Machine Learning*. Morgan Kaufmann, 1997, pp. 179–186.

[25] P. E. Hart, "The condensed nearest neighbor rule (corresp.)." *IEEE Transactions on Information Theory*, vol. 14, no. 3, pp. 515–516, 1968.

[26] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "MUTE: Majority under-sampling technique," in *2011 8th International Conference on Information, Communications & Signal Processing*. IEEE, Dec. 2011, pp. 1–4.

[27] B. Schlkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution." *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[28] M. Lin, K. Tang, and X. Yao, "Dynamic sampling approach to training neural networks for multiclass imbalance classification." *IEEE Trans. Neural Netw. Learning Syst.*, vol. 24, no. 4, pp. 647–660, 2013. [Online]. Available: http://dblp.uni-trier.de/db/journals/tnn/tnn24.html#LinTY13

[29] U. Bhowan, M. Johnston, M. Zhang, and X. Yao, "Evolving Diverse Ensembles using Genetic Programming for Classification with Unbalanced Data," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 3, pp. 368–386, 2013.

[30] Q.-Y. Yin, J.-S. Zhang, C.-X. Zhang, and N.-N. Ji, "A Novel Selective Ensemble Algorithm for Imbalanced Data Classification Based on Exploratory Undersampling," *Mathematical Problems in Engineering*, vol. 2014, pp. 1–14, 2014.

[31] J. Wang, P.-l. Huang, K.-w. Sun, B.-l. Cao, and R. Zhao, "Ensemble of Cost-Sensitive Hypernetworks for Class-Imbalance Learning," in *2013 IEEE International Conference on Systems, Man, and Cybernetics*. IEEE, Oct. 2013, pp. 1883–1888.

[32] Y. Qian, Y. Liang, M. Li, G. Feng, and X. Shi, "A resampling ensemble algorithm for classification of imbalance problems," *Neurocomputing*, vol. 143, pp. 57–67, Nov. 2014.

[33] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996. [Online]. Available: http://dx.doi.org/10.1023/A:1018054314350

[34] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997. [Online]. Available: http://dx.doi.org/10.1006/jcss.1997.1504

[35] E. R. Q. Fernandes, A. C. P. L. F. de Carvalho, and A. L. V. Coelho, "An evolutionary sampling approach for classification with imbalanced data," in *Neural Networks (IJCNN), 2015 International Joint Conference on*. IEEE, 2015, pp. 1–7.

[36] S. Wang and X. Yao, "Multiclass imbalance problems: Analysis and potential solutions," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 4, pp. 1119–1130, Aug 2012.

[37] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *Trans. Evol. Comp*, vol. 6, no. 2, pp. 182–197, Apr. 2002. [Online]. Available: http://dx.doi.org/10.1109/4235.996017

[38] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization," in *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, K. C. Giannakoglou, D. T. Tsahalis, J. Périaux, K. D. Papailiou, and T. Fogarty, Eds. Athens, Greece: International Center for Numerical Methods in Engineering, 2001, pp. 95–100.

[39] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*, ser. Wiley-Interscience Series in Systems and Optimization. John Wiley & Sons, Chichester, 2001.

[40] C. M. Fonseca and P. J. Fleming, "Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization," 1993.

[41] R. Poli and W. B. Langdon, *Genetic Programming with One-Point Crossover*. London: Springer London, 1998, pp. 180–189. [Online]. Available: http://dx.doi.org/10.1007/978-1-4471-0427-8_20

[42] D. J. Hand and R. J. Till, "A simple generalisation of the area under the roc curve for multiple class classification problems," *Machine Learning*, vol. 45, no. 2, pp. 171–186, 2001, 10.1023/A:1010920819831. [Online]. Available: http://dx.doi.org/10.1023/A:1010920819831

[43] J. Van Hulse, T. M. Khoshgoftaar, and A. Napolitano, "Experimental perspectives on learning from imbalanced data," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 935–942.

[44] M. Kukar and I. Kononenko, "Cost-sensitive learning with neural networks," in *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI-98*. John Wiley & Sons, 1998, pp. 445–449.

[45] Z.-H. Zhou and X.-Y. Liu, "On multi-class cost-sensitive learning." *Computational Intelligence*, vol. 26, no. 3, pp. 232–257, 2010. [Online]. Available: http://dblp.uni-trier.de/db/journals/ci/ci26.html#ZhouL10

[46] J. Zhu, H. Zou, S. Rosset, and T. Hastie, "Multi-class adaboost," *Statistics and its Interface*, vol. 2, no. 3, pp. 349–360, 2009.

[47] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[48] A. P. Bradley, "The use of the area under the roc curve in the evaluation of machine learning algorithms," *Pattern Recognition*, vol. 30, pp. 1145–1159, 1997.

[49] F. J. Provost and T. Fawcett, "Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions." in *KDD*, D. Heckerman, H. Mannila, and D. Pregibon, Eds., 1997, pp. 43–48.

[50] Y. Sun, M. S. Kamel, and Y. Wang, "Boosting for learning multiple classes with imbalanced class distribution," in *Sixth International Conference on Data Mining (ICDM'06)*, Dec 2006, pp. 592–602.

[51] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Neurocomputing: Foundations of research," J. A. Anderson and E. Rosenfeld, Eds. Cambridge, MA, USA: MIT Press, 1988, ch. Learning Representations by Back-propagating Errors, pp. 696–699. [Online]. Available: http://dl.acm.org/citation.cfm?id=65669.104451

[52] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Dec. 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1248547.1248548

[53] C. A. Shipp and L. I. Kuncheva, "Relationships between combination methods and measures of diversity in combining classifiers," *Information Fusion*, vol. 3, no. 2, pp. 135 – 148, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1566253502000519

[54] M. Li, S. Yang, and X. Liu, "Pareto or non-pareto: Bi-criterion evolution in multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 645–665, Oct 2016.

[55] H. Wang, L. Jiao, and X. Yao, "Two_arch2: An improved two-archive algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 4, pp. 524–541, Aug 2015.

Everlandio R. Q. Fernandes is a Ph.D. in Computer Science and Computational Mathematics at the University of So Paulo (USP) 2018. He received in 2002 the B.Sc. degree in Computer Sciences from the Federal University of Rio Grande do Norte, Brazil and M.Sc. degree in Applied Informatics from the University of Fortaleza (2009). His main research interests are clustering, pattern recognition, data mining, ensembles of classifiers and evolutionary algorithms.

André C. P. L. F. de Carvalho is Full Professor in the department of Computer Science, University of So Paulo, Brazil. He was Associate Professor in the University of Guelph, Canada. He was visiting researcher in the University of Porto, Portugal and visiting professor University of Kent, UK. He is Assessor ad hoc for funding Agencies in Brazil, Canada, UK, Czech Republic and Chile. His main research interests are data mining, data science and machine learning. Prof. de Carvalho has more than 300 publications in these areas, including 10 paper awards from conferences organized by ACM, IEEE and SBC. He is the director of the Center of Machine Learning in Data Analysis, University of So Paulo.

Xin Yao received the B.Sc. and Ph.D. degrees from the University of Science and Technology of China, Hefei, China, in 1982 and 1990, respectively. He is a Chair Professor of computer science with the Southern University of Science and Technology, Shenzhen, China, and a Professor of computer science with the University of Birmingham, Birmingham, U.K. He has been researching on multiobjective optimization since 2003, when he published a well-cited EMO03 paper on many objective optimization. His current research interests include evolutionary computation, ensemble learning, and their applications in software engineering.