Evolutionary ensembles for imbalanced learning

Everlandio Rebouças Queiroz Fernandes

Tese de Doutorado do Programa de Pós-Graduação em Ciências de
Computação e Matemática Computacional (PPG-CCMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Everlandio Rebouças Queiroz Fernandes

Evolutionary ensembles for imbalanced learning

Doctoral dissertation submitted to the Institute of Mathematics and Computer Sciences – ICMC-USP, in partial fulfillment of the requirements for the degree of the Doctorate Program in Computer Science and Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. André Carlos Ponce de Leon Ferreira de Carvalho

USP – São Carlos
October 2018

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

F363e Fernandes, Everlandio Rebouças Queiroz
Evolutionary ensembles for imbalanced learning /
Everlandio Rebouças Queiroz Fernandes; orientador
Andre Carlos Ponce de Leon Ferreira de Carvalho. --
São Carlos, 2018.
136 p.

Tese (Doutorado - Programa de Pós-Graduação em
Ciências de Computação e Matemática Computacional) --
Instituto de Ciências Matemáticas e de Computação,
Universidade de São Paulo, 2018.

1. Aprendizado de Máquina. 2. Problemas de
Classificação. 3. Aprendizado Desbalanceado. 4.
Ensemble de Classificadores. 5. Algoritmos
Evolutivos. I. Carvalho, Andre Carlos Ponce de Leon
Ferreira de , orient. II. Título.

Everlandio Rebouças Queiroz Fernandes

Comitês evolucionários para aprendizado desbalanceado

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. André Carlos Ponce de Leon Ferreira de Carvalho

USP – São Carlos
Outubro de 2018

Este trabalho é dedicado aos meus pais, irmãos e amigos.

ACKNOWLEDGEMENTS

My first thanks go to my earthly parents and my spiritual father. During all the uncertainties of life, I have always been able to count on their understanding and support.

I am immensely grateful to my advisor, Prof. Andre de Carvalho. His passion for research overflows to his students. Certainly, one of the most incredible people I've ever met.

My gratitude also goes to Prof. Xin Yao and Prof. Joost Kok for receiving me in their institutions and sharing their knowledge during my internship periods.

I would like to thank FAPESP for the financial support which made possible the development of this work (Grants 2013/11615-6, 2015/01370-1 and 2016/20465-6).

Finally, I would like to thank my friends and lab colleagues, they made the doctoral period pass very fast. Thanks to them for the discussions about "science", movies, parties, beers, sports, etc. They have contributed significantly to increasing my life skills.

*“Attitude is a little thing that makes
a big difference”
(Winston Churchill)*

ABSTRACT

FERNANDES, E. R. Q. **Evolutionary ensembles for imbalanced learning**. 2018. 136 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

In many real classification problems, the data set used for model induction is significantly imbalanced. This occurs when the number of examples of some classes is much lower than the other classes. Imbalanced datasets can compromise the performance of most classical classification algorithms. The classification models induced by such datasets usually present a strong bias towards the majority classes, tending to classify new instances as belonging to these classes. A commonly adopted strategy for dealing with this problem is to train the classifier on a balanced sample from the original dataset. However, this procedure can discard examples that could be important for a better class discrimination, reducing classifier efficiency. On the other hand, in recent years several studies have shown that in different scenarios the strategy of combining several classifiers into structures known as ensembles has proved to be quite effective. This strategy has led to a stable predictive accuracy and, in particular, to a greater generalization ability than the classifiers that make up the ensemble. This generalization power of classifier ensembles has been the focus of research in the imbalanced learning field in order to reduce the bias toward the majority classes, despite the complexity involved in generating efficient ensembles. Optimization meta-heuristics, such as evolutionary algorithms, have many applications for ensemble learning, although they are little used for this purpose. For example, evolutionary algorithms maintain a set of possible solutions and diversify these solutions, which helps to escape out of the local optimal. In this context, this thesis investigates and develops approaches to deal with imbalanced datasets, using ensemble of classifiers induced by samples taken from the original dataset. More specifically, this theses propose three solutions based on evolutionary ensemble learning and a fourth proposal that uses a pruning mechanism based on dominance ranking, a common concept in multiobjective evolutionary algorithms. Experiments showed the potential of the developed solutions.

Keywords: Imbalanced Learning, Data Classification, Ensemble of Classifiers, Evolutionary Algorithms.

RESUMO

FERNANDES, E. R. Q. **Comitês evolucionários para aprendizado desbalanceado**. 2018. 136 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

Em muitos problemas reais de classificação, o conjunto de dados usado para a indução do modelo é significativamente desbalanceado. Isso ocorre quando a quantidade de exemplos de algumas classes é muito inferior às das outras classes. Conjuntos de dados desbalanceados podem comprometer o desempenho da maioria dos algoritmos clássicos de classificação. Os modelos de classificação induzidos por tais conjuntos de dados geralmente apresentam um forte viés para as classes majoritárias, tendendo classificar novas instâncias como pertencentes a essas classes. Uma estratégia comumente adotada para lidar com esse problema, é treinar o classificador sobre uma amostra balanceada do conjunto de dados original. Entretanto, esse procedimento pode descartar exemplos que poderiam ser importantes para uma melhor discriminação das classes, diminuindo a eficiência do classificador. Por outro lado, nos últimos anos, vários estudos têm mostrado que em diferentes cenários a estratégia de combinar vários classificadores em estruturas conhecidas como comitês tem se mostrado bastante eficaz. Tal estratégia tem levado a uma acurácia preditiva estável e principalmente a apresentar maior habilidade de generalização que os classificadores que compõe o comitê. Esse poder de generalização dos comitês de classificadores tem sido foco de pesquisas no campo de aprendizado desbalanceado, com o objetivo de diminuir o viés em direção as classes majoritárias, apesar da complexidade que envolve gerar comitês de classificadores eficientes. Meta-heurísticas de otimização, como os algoritmos evolutivos, têm muitas aplicações para o aprendizado de comitês, apesar de serem pouco usadas para este fim. Por exemplo, algoritmos evolutivos mantêm um conjunto de soluções possíveis e diversificam essas soluções, o que auxilia na fuga dos ótimos locais. Nesse contexto, esta tese investiga e desenvolve abordagens para lidar com conjuntos de dados desbalanceados, utilizando comitês de classificadores induzidos a partir de amostras do conjunto de dados original por meio de metaheurísticas. Mais especificamente, são propostas três soluções baseadas em aprendizado evolucionário de comitês e uma quarta proposta que utiliza um mecanismo de poda baseado em ranking de dominância, conceito comum em algoritmos evolutivos multiobjetivos. Experimentos realizados mostraram o potencial das soluções desenvolvidas.

Palavras-chave: Aprendizado Desbalanceado, Classificação de Dados, Comitê de Classificadores, Algoritmos Evolutivos.

LIST OF FIGURES

Figure 1 – Class Imbalance Scenarios	29
Figure 2 – MOGASamp - Multiobjective Genetic Sampling	52
Figure 3 – E-MOSAIC - Ensemble of Classifier based on Multiobjective Genetic Sampling for Imbalanced Classification	68
Figure 4 – Mauc Data Level Methods	76
Figure 5 – G-Mean Data Level Methods	76
Figure 6 – Mauc Algorithm Level Methods	77
Figure 7 – Gmean Algorithm Level Methods	78
Figure 8 – Minimum Spanning Tree	97
Figure 9 – EVINCI's Workflow	98
Figure 10 – Crossover process	101
Figure 11 – Figure A Represents a Sample taken from Initial Population and Figure B a Sample From Fifth Generation	107
Figure 12 – Proposed method Workflow. Balanced samples are generated from the unbalanced dataset, each sample will be applied to a CNN, with the results obtained (accuracy and diversity) passed by a non-dominant rank followed by the application of the pruning technique to obtain the ensemble result. . .	115
Figure 13 – Pad absent	118
Figure 14 – Undamaged pad	118
Figure 15 – Damaged pad	118

LIST OF ALGORITHMS

Algorithm 1 – Evolutionary Algorithm	35
--	----

LIST OF TABLES

Table 1 – Databases Used for the Experimental Tests	51
Table 2 – AUC and classification accuracy of the Minority and Majority classes (average and standard deviation) using different resampling and classification techniques	55
Table 3 – Basic Characteristics of The Datasets (#F: The Number of Features, #C: The Number of Classes, #Inst.: The Total Number of Instances)	74
Table 4 – Parameters for MLP	75
Table 5 – Number of win-draw-lose between E-MOSAIC and the algorithm-level compared methods.	78
Table 6 – Accuracy for each class returned by E-MOSAIC on <i>Chess</i> , <i>Glass</i> , <i>Car</i> and <i>Conceptive</i> datasets.	81
Table 7 – N1byClass based on the MST shown in Figure 8	97
Table 8 – Basic Dataset Characteristics (#C: Number of Classes, #F: Number of Features, Imbalance Ratio, Class Distribution. Minority classes indicated by Equation 4.2 are in bold.)	103
Table 9 – G-mean Values Achieved by Different Methods in the Experiments over 30 Runs with their Ranks by Dataset (between parentheses), G-mean Average for Each Method, Ranking Count for Each Method, and Ranking Average	104
Table 10 – G-mean Achieved by Different Versions in the Experiments over 30 Runs, G-mean Average for Each Version, Ranking Count for Each Version, and Ranking Average	106
Table 11 – Comparison of approaches. The MPL, LeNet, CNN, and ILEC approaches were compared at different epochs according to the G-mean, Standard Deviation (SD) and accuracy of each of the classes.	120
Table 12 – G-mean Achieved by the Proposed Methods in the Experiments over 30 Runs with Population Size Equals to 30 and the Maximum Limit of 30 Generations. Also presents G-mean Average for Each Method, Ranking Count, and Ranking Average	132
Table 13 – MAuc Achieved by the Proposed Methods in the Experiments over 30 Runs with Population Size Equals to 30 and Maximum Limit of 30 Generations. Also presents MAuc Average for Each Method, Ranking Count, and Ranking Average	133

Table 14 – G-mean Achieved by the Proposed Methods in the Experiments over 30 Runs with Population Size Equals to 10 and Maximum Limit of 20 Generations. Also presents G-mean Average for Each Method, Ranking Count, and Ranking Average	134
Table 15 – MAuc Achieved by the Proposed Methods in the Experiments over 30 Runs with Population Size Equals to 10 and Maximum Limit of 20 Generations. Also presents MAuc Average for Each Method, Ranking Count, and Ranking Average	135

LIST OF ABBREVIATIONS AND ACRONYMS

CNN	Convolutional Neural Networks
E-MOSAIC	Ensemble of Classifiers based on Multiobjective Genetic Sampling for Imbalanced Classification
EA	Evolutionary Algorithms
EVEN	Evolutionary Ensemble
EVINCI	Evolutionary Inversion of Class Distribution for Imbalanced Learning
ILEC	Imbalanced Learning with Ensemble of Convolutional Neural Network
ML	Machine Learning
MOEA	Multiobjective Evolutionary Algorithm
MOGASamp	Multiobjective Genetic Sampling
NCL	Negative Correlation Learning
OSS	One-Sided Selection
PFC	Pairwise Failure Crediting
ROS	Random Oversampling
RUS	Random Undersampling
SMOTE	Synthetic Minority Oversampling Technique

CONTENTS

1	INTRODUCTION	27
1.1	Imbalanced Learning	28
1.2	Ensembles of Classifiers for Imbalanced Learning	31
1.3	Evolutionary-based Ensemble	34
1.4	Objectives	36
1.5	Hypothesis	36
1.6	Thesis Organization	37
1.6.1	<i>Chapter 2</i>	37
1.6.2	<i>Chapter 3</i>	38
1.6.3	<i>Chapter 4</i>	39
1.6.4	<i>Chapter 5</i>	40
1.6.5	<i>Chapter 6</i>	40
1.6.6	<i>Appendix A</i>	40
1.7	Bibliography	41
2	AN EVOLUTIONARY SAMPLING APPROACH FOR CLASSIFI- CATION WITH IMBALANCED DATA	47
2.1	Introduction	48
2.2	Literature Review	49
2.3	Performance Evaluation	50
2.3.1	<i>Accuracy</i>	50
2.3.2	<i>Diversity</i>	51
2.4	MOGASamp	52
2.4.1	<i>Sampling and the Training Models</i>	52
2.4.2	<i>Evaluation</i>	53
2.4.3	<i>Genetic Operators</i>	53
2.4.4	<i>Elimination of Identical Solutions</i>	53
2.4.5	<i>New Generation and Stop Criterion</i>	54
2.5	Experimental results	54
2.6	Conclusion	56
2.7	Bibliography	57

3	ENSEMBLE OF CLASSIFIERS BASED ON MULTIOBJECTIVE GENETIC SAMPLING FOR IMBALANCED DATA	61
3.1	Introduction	62
3.2	Related Works	64
3.2.1	<i>Data level approaches</i>	64
3.2.2	<i>Algorithm level approaches</i>	65
3.2.3	<i>Ensemble approaches</i>	66
3.3	The Proposed Method	67
3.3.1	<i>Sampling and the Training Models</i>	68
3.3.2	<i>Fitness Evaluation</i>	69
3.3.3	<i>Selection and Genetic Operators</i>	70
3.3.4	<i>Elimination of Identical Solutions</i>	70
3.3.5	<i>New Generation and Stop Criterion</i>	71
3.4	Experimental Study	72
3.4.1	<i>Compared Methods</i>	72
3.4.2	<i>Metrics</i>	72
3.4.3	<i>Experimental Setup</i>	73
3.4.4	<i>Experimental Results</i>	75
3.4.4.1	<i>Comparison with Data Level Methods</i>	75
3.4.4.2	<i>Comparison with Algorithm Level Methods</i>	77
3.4.5	<i>Further Analysis</i>	80
3.5	Conclusion	81
3.6	Bibliography	82
4	EVOLUTIONARY INVERSION OF CLASS DISTRIBUTION IN OVERLAPPING AREAS FOR MULTI-CLASS IMBALANCED LEARNING	89
4.1	Introduction	90
4.2	Unbalanced Datasets Methods and Issues	92
4.3	Imbalanced Ensemble Learning	95
4.4	N1byClass	96
4.5	Proposed Method	98
4.5.1	<i>Initial Population and Fitness</i>	100
4.5.2	<i>Selection and Reproduction</i>	100
4.5.3	<i>New Generation, Saved Ensemble and Stop Criteria</i>	101
4.6	Experiments	102
4.6.1	<i>Experimental Setup</i>	103
4.6.2	<i>Experimental Results - Compared Methods</i>	105
4.6.3	<i>Further Analysis</i>	105
4.7	Conclusion	107

4.8	Bibliography	108
5	AN ENSEMBLE OF CONVOLUTIONAL NEURAL NETWORKS FOR UNBALANCED DATASETS: A CASE STUDY WITH WAGON COMPONENT INSPECTION	113
5.1	INTRODUCTION	114
5.2	Related Work	115
5.3	Imbalanced Learning with Ensemble of Convolutional Neural Net- work (ILEC)	116
5.4	Statement of the problem	118
5.5	Experiments	119
5.5.1	<i>Database</i>	119
5.5.2	<i>Texture analysis</i>	119
5.5.3	<i>Deep Learning Approach</i>	120
5.5.3.1	<i>LeNet</i>	120
5.5.3.2	<i>CNN Architecture</i>	121
5.6	Results	121
5.7	Conclusions and Future Work	122
5.8	Bibliography	123
5.9	Future Work	129
5.10	Bibliography	130
A	COMPARISON OF THE PROPOSED METHODS	131
A.1	Experimental Setup	131
A.2	Experimental Results	133
A.3	Bibliography	136

INTRODUCTION

In supervised Machine Learning (ML), classification is a task in which a learning algorithm learns from a set of labeled instances. Thus, given a set of instances composed by attributes or characteristics and their corresponding labels, the algorithm induces a classification model able to predict the association of the attributes of a instance to its label (MITCHELL, 1997). Basic concepts in supervised learning tasks are training and test datasets. The former is the collection of instances from which a classification model is induced. The latter is a collection of instance similar to the training data, but not used during the learning process. It is used to evaluate the predictive accuracy of the induced model with known labeling instances that were not used for the model induction.

In a desired scenario, the training data used by a classification algorithm to induce the classification model should contain instances representing the task to be solved and similarly distributed among the dataset classes (VLADISLAVLEVA; SMITS; HERTOOG, 2010). Examples of classification algorithms include decision tree induction algorithms, support vector machines, multilayer perceptron neural networks, Bayesian networks, and k -nearest neighbors. However, in some datasets, some areas of the feature space can often have an abundance of instances, while a few instances populate other regions. When this occurs, the dataset is considered imbalanced. For example, one can observe this behavior in a study of a rare disease in a given population. The number of instances available representing sick people (minority class) can be much lower than the number of instances available from healthy people (majority class).

In these cases, imbalanced datasets can make many classical classification algorithms less effective, especially when predicting minority class instances. This occurs because these algorithms are designed to induce models that generalize the training data, then return the simplest classification model, and that best fits the data. Although these algorithms can produce classification models with high overall accuracy, they often tend to undermine the identification of instances belonging to the minority classes, since simpler models give less attention to rare cases, treating them as noise sometimes (SUN et al., 2007). As a result of data imbalance, the

resulting classifier might lose its classification ability in such scenarios.

Consider, for example, the k -nn classification algorithm with k equals 1. This algorithm labels a new instance with the same class as its nearest neighbor in the training dataset. If the training data contains very few instances of the minority class, it is likely that the nearest neighbor of a new minority class instance belongs to the majority class, producing a misclassification. These situations represent the area of machine learning known as imbalanced learning, which is the object of study of this thesis. Imbalanced learning has been identified as one of the most challenging problems in machine learning and data mining due to its significant effects on classifier construction and predictive performance (HAIXIANG et al., 2017).

In this chapter, Sections 1.1 to 1.3 state the problem addressed in this thesis. Section 1.4 presents the main objectives set during the time of the doctorate and 1.5 defines the thesis hypothesis. Finally, Section 1.6 present the organization of this thesis.

1.1 Imbalanced Learning

In imbalanced learning, there are two distinct scenarios: when the dataset is binary and when it is multi-class. Figure 1 shows the data distribution in these two imbalanced scenarios. In Figure 1a (a binary dataset), it is easy to identify that the minority class is the class with less elements, the other being the majority class. This well-defined relationship between the classes allows us to determine the potential bias of the classification algorithms and counterbalance it towards the minority class, since this is usually the class of most interest. In a multi-class dataset (e.g., Figure 1b), the relationship between a pair of classes does not adequately reflect the whole imbalance problem. A class can at the same time be a majority for one group of classes and minority for another or even have a similar distribution for another group (SÁEZ; KRAWCZYK; WOŹNIAK, 2016). Besides, multi-class datasets may have more than one class of interest, i.e., multiple classes in which the classifier must present a high predictive accuracy.

The induction of classification models from imbalanced datasets has been investigated in the machine learning literature for at least the last 20 years. However, most of the proposed techniques were designed and tested only for binary dataset scenarios. Unfortunately, when working with a multi-class dataset, several of the solutions proposed in the literature for binary classification may not be directly applicable or, when they can be applied, achieve predictive performance below expectation (FERNÁNDEZ et al., 2013).

Class decomposition is a technique commonly applied to multi-class datasets to allow the application of methods that were developed for binary datasets. One of the most common applications of this technique is known as one-against-all or one-vs-all, where a multi-class classification problem is transformed into a set of binary classification sub-problems. In this technique, one class of the multi-class dataset is chosen as the positive class, and all other classes are labeled as negative. This new labeling of the dataset is used to induce a binary

Figure 1 – Class Imbalance Scenarios

classifier. The process is repeated so that, at each round, a different class is labeled as positive (RIFKIN; KLAUTAU, 2004). For example, given a dataset with five classes, five binary classifiers are induced. However, as presented in (WANG; YAO, 2012), this technique presents some deficiencies when applied to multi-class imbalanced learning. Besides, it makes the sub-problems generated even more imbalanced.

The solutions proposed in the literature for imbalanced learning can be divided into two groups: at data and at algorithm levels. The first group, the resampling methods, which is the most popular in the literature, preprocesses the dataset before the application of a classification algorithm. Solutions from this group focus mainly on modifying the ration of instances in the classes, trying to reduce the bias of traditional classification algorithms to favor the majority class. Experiments associated with these proposals empirically show that the application of a preprocessing step to rebalance class distribution often improves minority class recognition. The resampling methods can be categorized into three groups:

1. **Oversampling** performs the replication of preexisting instances in the original dataset or synthetically generates new instances. The selection of the preexisting instances can occur randomly (e.g. Random Oversampling (ROS)) or directed by subconcepts that compose the class. Regarding the generation of synthetic data, the interpolation technique is commonly used for this purpose, as is the case of Synthetic Minority Oversampling Technique (SMOTE) (CHAWLA et al., 2002). Due to the increasing number of instances in the training dataset, the oversampling methods usually increase the computational cost for the classifier induction. Besides, it can create instances that would never be found in the investigated problem.
2. **Undersampling** the opposite of the previous group, uses only a subset of the majority class and all instances of the minority class as training dataset. The well-known Random

Undersampling (RUS) is a simple method employed to shrink the majority class by randomly discarding some of its instances. Although it is simple to use, there may be a loss of relevant information from the classes that have been reduced. Directed or informative undersampling attempts to work around this problem by detecting and eliminating a less significant fraction of the data (e.g. One-Sided Selection (OSS) (KUBAT; MATWIN, 1997)).

3. **Hybrid methods** combines the oversampling and undersampling methods in an attempt to benefit from both and as a result, reduce the drawbacks caused by each. Since oversampling, despite increasing the representativeness of the minority class, can aggravate the computational cost of the learning algorithm, and undersampling, in trying to reduce classifier bias, can eliminate representative instances of the majority class (BATISTA; PRATI; MONARD, 2004) (SEIFFERT; KHOSHGOFTAAR; Van Hulse, 2009).

The main advantage of the resampling methods is that they are independent of the classification algorithms used to induce predictive model from the dataset. Although they are easy to implement and use, there are reports of problems related to overfitting and over-generalization (WANG, 2011).

In the algorithm-level group, the solutions are based on the adaptation of an existing classification algorithm to reduce its bias towards the majority class or on the proposal of new algorithms that take into account the skewed distribution of the classes. There are two main categories in this group — Recognition-based and Cost-sensitive methods:

1. **Recognition-based methods** take the form of one-class learners. This is an extreme case in which only instances from one class are used to induce a classification model. One-class SVM (SCHÖLKOPF et al., 2001) is a classification-based method that, in order to recognize the class of interest, considers only minority class instances during the learning process. The support vector machine algorithm in One-class SVM is trained on data that has only the normal instances (minority class instances). It infers the properties of normal instances to induce a predictive model and use the induced model to predict which instances are unlike the normal class instances. However, as indicated by Ali, Shamsuddin and Ralescu (2015), some classification algorithms do not work with instances from only one class, which makes these methods unpopular and restricts them to certain learning algorithms.
2. **Cost-sensitive methods** change the cost function used by classification algorithms (existing algorithms or new proposals) so that the cost of labeling a minority class instance as being of the majority class has a higher penalty. This effect can be achieved, for example, by changing the tree splitting criterion in decision tree algorithms (DRUMMOND; HOLTE, 2000) (LING et al., 2004). However, a constant drawback of such methods is that the costs

corresponding to the wrong classification must be provided in advance, requiring prior knowledge of the problem which is not available in many real situations.

The main negative aspect of algorithm-level methods is that they are usually specific to certain classification algorithms and/or problems, which makes them effective only in particular domains (SUN; WONG; KAMEL, 2009). Besides, to develop a solution at the algorithm-level, extensive knowledge about the classification algorithm and application domain is necessary.

In addition to the aforementioned data-level and algorithm-level methods, there has been a growing use of ensembles of classifiers as a possible solution for imbalanced learning (BHOWAN et al., 2013) (WANG et al., 2013) (YIN et al., 2014) (QIAN et al., 2014). Ensemble learning has as main characteristic the induction of diversified classifiers, which are combined to form a new classification system with higher generalization ability than the individual classifiers that compose the ensemble. This generalization power of the ensembles has been a current focus of research in the imbalanced learning aiming to reduce the bias toward the majority classes. Due to the importance that ensemble learning has for the development of this thesis, the next section discusses the application of ensemble of classifiers for imbalanced learning.

1.2 Ensembles of Classifiers for Imbalanced Learning

Ensemble learning is a well known approach in the machine learning area. It has been successfully applied to many problems, such as — but not limited to — remote sensing, face and fingerprint recognition, intrusion detection in networks and medicine (OZA; TUMER, 2008). In contrast to other machine learning methodologies that construct a single hypothesis (model) of the training dataset, ensemble learning methods induce a set of hypotheses and combine them through some aggregation method or operator. They usually combine base classifiers known as weak learners. The primary motivation for combining classifiers in ensembles is to improve the predictive model generalization ability. Since the base classifiers can do some misclassifications, considering that a limited sample of the data induced them (use of different subsets of the original dataset or datasets with different predictive attributes), but these errors are not necessarily the same (KITTLER et al., 1998).

In fact, Dietterich (1997) discusses and provides an overview of why ensemble methods can outperform single-classifier methods in classification tasks. Additionally, Hansen and Salamon (1990) demonstrate that, under specific constraints, the expected error rate of an instance decreases to zero as the number of base classifiers increases. For this, the base classifiers must have an accuracy rate higher than 50% and be as diverse as possible. Classifiers are diverse when they commit misclassifications in different instances for the same test dataset. Thus, the central aspects of ensemble learning refer to the accuracy and diversity requirements of its base classifiers, which can be implemented through parallel or sequential heuristics. However, it is worth to note that high accuracy and diversity are conflicting requirements, since, as the accuracy

of classifiers increases, the number of misclassifications decreases, making it more challenging to have misclassifications in different instances by different classifiers.

Bagging (BREIMAN, 1996) and Boosting (FREUND; SCHAPIRE, 1997) are the two most popular ensemble learning algorithms proposed in the literature (BROWN, 2017). They provide a strategy for manipulating the training dataset before the induction of each base classifier, in order to promote the diversity required. In Bagging, different samples bootstrapped from the training dataset induce the set of base classifiers. This sampling is carried out with replacement, and each sample has the same size and class distribution as in the original dataset. When the ensemble is asked to label a new instance, each base classifier makes its prediction, and the new instance receives the label of the class with the highest number of votes (majority vote).

AdaBoost (FREUND; SCHAPIRE, 1997) is the most typical algorithm in the Boosting family. It uses the whole training dataset to create classifiers sequentially. Adaboost uses a weighting strategy of the training instances to indicate which of them should receive more attention when inducing a new classifier. At each training iteration, the efficiency of the generated classifier is verified in the complete training data and the instances that were incorrectly classified receive a larger weight. The training instances with the highest weights participate more effectively in the induction of the next classifier. When a new instance appears, each base classifier produces its weighted vote (weighted by its accuracy in the entire training dataset), and the label of the new instance is determined.

However, when applying ensemble learning to imbalanced datasets, it can produce the same bias toward the majority class found in the use of individual classification algorithms (GALAR et al., 2012). On the other hand, the promise of improvements in the generalization ability and accuracy offered by ensembles methods is very attractive in the context of imbalanced learning. This is the main motivation for research focused on combining ensemble learning with some method that deals with class imbalance problem. The publications of these researches present experimental results that show significant improvements in the correct classification of the instances of the minority classes. Thus, the proposed solutions are usually hybrid methods in which ensemble learning algorithms add some method of resampling or cost-sensitive or even adaptation of existing classification algorithms.

Galar et al. (2012) proposed a taxonomy for imbalanced ensemble learning, subdividing the solutions proposed in the literature into four groups — Cost-sensitive Boosting, Boosting-based, Bagging-based, and Hybrid methods. The Cost-sensitive Boosting methods are similar to the cost-sensitive methods, but the process of cost minimization is embedded into the Boosting algorithm. The other three groups share the characteristic of incorporating a resampling method into the ensemble learning process.

1. **Cost-sensitive Boosting** proposes to change the function that updates the weights of the instances in each iteration, taking into account the skewed distribution of the classes and

placing more load in the class associated with the higher importance of identification (SUN et al., 2007). For example, AdaCost (FAN et al., 1999) introduces a cost adjustment function β within the AdaBoost weighting function. This cost adjustment produces a change in the weighting function that causes instances from the minority class to receive more attention than those from the majority class when they are misclassified. On the other hand, in the case of correct classification, the weight of the instances from the minority class is decreased in a more conservative way than in those from the majority class.

2. **Boosting-based** applies a data preprocessing method to each AdaBoost iteration. A common feature in these methods is the rebalancing of the training dataset while retaining the original AdaBoost weighting function. Thus, each of the resampling methods presented early is a possible candidate to be integrated with AdaBoost and, as a result, generate a new boosting-based solution. Examples of methods from this group are SMOTEBoost (CHAWLA et al., 2003) and RUSBoost (SEIFFERT et al., 2010).
3. **Bagging-based** uses several samples bootstrapped from the training set to induce their base classifiers in parallel, like in the original Bagging algorithm. However, in the solutions that belong to this group, the samples undergo some process of rebalancing. Thus, the main factor is how to process the rebalancing of the samples that will be used to induce classifiers. As a result, different resampling methods for imbalanced learning lead to different bagging-based methods. For example, Wang and Yao (2009) proposed different bagging-based methods with the application of methodologies that aim to increase the diversity of the base classifiers. Examples of these methods are OverBagging, SMOTEBagging, and UnderOverBagging.
4. **Hybrid methods** tries to add the benefits of the Boosting and Bagging to a resampling method. The boosting-based methods have the characteristic of decreasing the bias of their base classifiers, diminishing their tendency to not learn correctly as a consequence of not taking into account all information of the dataset (underfitting). The bagging-based methods, on the other hand, are very effective in decreasing the variances of the classifiers. Thus, when the base classifiers suffer from overfitting, bagging methods tend to overcome this problem (GALAR et al., 2012). In this category are EasyEnsemble and BalanceCascade (LIU; WU; ZHOU, 2009), and RotEasy (YIN et al., 2014).

As previously mentioned, accuracy and diversity are conflicting objectives in ensemble learning methods. This can be considered as one of the most significant trade-offs for building an effective ensemble of classifiers, i.e., finding the boundary between high accuracy and diversity. In fact, many real-world problems incorporate multiple performance measures (or objectives), which must be improved (or attained) simultaneously. Often, the process of optimizing one measure interferes negatively with another, making the appropriate solution for one objective a poor or unacceptable solution to another.

Evolutionary algorithms are particularly suited to deal with multiobjective optimization problems, since they deal simultaneously with a set of solutions (population) that allows finding a complete set of acceptable solutions in a single algorithm execution, rather than performing a series of separate runs (COELLO, 1999). Besides, although evolutionary-based ensembles are not among the most popular ensembles methods, the metaheuristic optimization of evolutionary algorithms has many applications for ensemble learning, as can be seen in the next section, which also discusses the application of evolutionary-based ensembles for imbalanced learning.

1.3 Evolutionary-based Ensemble

Evolutionary Algorithms (EA) represent a group of population-based search and optimization algorithms that simulate the evolution of individual solutions through inter-relational processes of selection, reproduction, and mutation. Its optimization ability has been highlighted due to its high adaptability to provide good solutions to problems from different application domains. These domains include, mechanical design, environmental protection, and finance, just to name a few (WONG, 2015). As presented by Zitzler, Laumanns and Bleuler (2004), three aspects categorize an evolutionary algorithm: *i*) a set of possible solutions is maintained; *ii*) a selective breeding process is carried out in this set; *iii*) solutions can be combined to generate new solutions.

Evolutionary algorithms use many concepts coming from genetics, Darwin's evolutionary theory (DARWIN, 1859) and cellular biology, and consequently adopts much of their terminologies. Thus, a candidate solution to a problem represents an individual in a population of solutions at a given point in the processing of the evolutionary algorithm. The representation or coding of an individual is commonly called genome or chromosome, and, as in biology, a chromosome represents a sequence of characteristics of the individual, i.e., genes. The process of combining individuals for the generation of new solutions (offspring or child) can occur by swapping parts of the chromosomes of previously selected individuals (breeding or reproduction) or by inserting a perturbation in a chromosome, known as mutation. Each individual in a given population receives values that indicate the quality of the solution represented by it, symbolizing its fitness. As with the natural process of selecting the fittest individuals, the fitness of the solutions serves to choose which individuals will participate of the reproduction process and, after the offspring generation, which will compose the new generation of solutions. Finally, the entire process of searching and creating solutions that present better fitness represents the evolution of a population (KICINGER; ARCISZEWSKI; JONG, 2005). A canonical EA consists of the steps described in Algorithm 1.

The diversity or heterogeneity of the population is essential for the evolutionary algorithm to carry out an useful exploration of the solution space as it goes from one generation to another (GREFENSTETTE, 1987). Thus, the similarities between ensemble learning and

Algorithm 1 – Evolutionary Algorithm

```

1:  $t \leftarrow 0$ 
2: INITPOPULATION( $P(t)$ ) ▷ Generate the initial population
3: EVALPOPULATION( $P(t)$ ) ▷ Calculate fitness of the population individuals
4: while  $!(Termination_{condition})$  do
5:    $parents \leftarrow SELECTION(P(t))$  ▷ Selects the individuals that will breed
6:    $offspring \leftarrow GENETIC-OPERATORS(parents)$  ▷ Create new individuals
7:   EVALPOPULATION( $offspring$ )
8:   REPLACE( $P(t), offspring$ ) ▷ Replace part or entire population with offspring
9:    $t \leftarrow t + 1$ 
10: end while

```

evolutionary algorithms begin to become stronger, since, in addition to maintaining a set of possible solutions (or classifiers in case of ensembles), the diversity of solutions represents an escape from the local optimal for both methodologies. Besides, as listed by Kovacs (2012), evolution has many applications within ensembles: *i)* Voting: evolving the weighting of the votes of the base classifiers. For example, to optimize the weight distribution of classifier votes. *ii)* Generation and evolution of base classifiers: providing the ensemble with a set of candidate members. *iii)* Classifier selection: the winning classifiers of the evolutionary process are added to the ensemble. *iv)* Features or instance selection: generating different classifiers by training them in different and optimized groups of features or instances.

As stated earlier, the main aspects of ensemble learning refer to the requirements of accuracy and diversity, which are conflicting objectives. Because it is a population-based method, evolutionary algorithms can be customized to produce many solutions that can be evaluated under more than one aspect. This customization characterizes the Multiobjective Evolutionary Algorithm (MOEA) category. This category of EA uses the concept of dominant solutions by considering all predefined objectives to select the most appropriate solutions. In MOEA, a solution x_1 dominates another solution x_2 if it is no worse than x_2 in any objective, and x_1 is undoubtedly better than x_2 in at least one of them. This technique allows individuals to be ranked according to their performance on all objectives, when compared with all other individuals in the population. Thus, a non-dominated solution is better fitted to the problem than the solutions dominated by many other solutions.

In the area of imbalanced learning, evolutionary-based ensembles have shown good results, despite still being an area with a small number of works. The method proposed in (CHAWLA; SYLVESTER, 2007), named Evolutionary Ensemble (EVEN), addresses the weighting of votes of the base classifiers. The authors argue that the members of an ensemble do not contribute equally to the improvement of the classification's performance and propose the use of a genetic algorithm to search for an optimized weighting of votes for previously induced models.

Using MOEA, Bhowan et al. (2013) propose a method based on multiobjective genetic programming (MOGP). Genetic Programming is a category of evolutionary algorithms in

which the representation of individuals have a tree-like structure. Using this representation, the method by [Bhowan et al. \(2013\)](#) models each individual as a classifier represented by a mathematical expression, and as conflicting objectives guide the evolutionary process, it uses the accuracy of the individual in the majority and minority classes separately. The authors also adapt MOEA to promote the most diverse solutions, taking into account the diversity measures Negative Correlation Learning (NCL) ([LIU; YAO, 1997](#)) or Pairwise Failure Crediting (PFC) ([CHANDRA; YAO, 2006](#)).

In the EUS-Bag ([SUN et al., 2017](#)) method, as in Bagging, each base classifier is induced by a sampling of the training dataset. Besides, each of these samples is the result of an evolutionary process that seeks an optimized subset of majority class instances. The method uses as fitness an equation composed of three terms — The first refers to the accuracy of the model induced by the sample, the second evaluates the sample imbalance rate and the third estimates the diversity of the model generated. After each evolutionary process, the resulting classifier of the individual presenting the best fitness is added to the final ensemble.

1.4 Objectives

This section presents the main objectives of this PhD thesis. The main goal of this research was to investigate how ensembles of classifiers can learn models with high predictive performance from imbalanced classification datasets. More specifically, the candidate investigated solutions, through evolutionary algorithms and ensemble of classifiers techniques, able to reduce the bias that imbalanced dataset causes in the classical classification algorithms, harming the predictive performance of the induced models.

1. Identify the main causes of poor performance that classical classification algorithms can present when applied to imbalanced datasets.
2. Investigate the solutions proposed in the literature for imbalanced learning, in particular those based on ensemble.
3. Identify deficiencies or opportunities to improve existing ensemble methods and how the use of different data samples can lead to better ensembles.
4. Propose ensemble methods able to overcome the deficiencies identified and, as a result, improve the predictive performance for a group of imbalanced classification datasets.

1.5 Hypothesis

In order to reach these objectives, the candidate formulated the following hypotheses:

- It is possible to optimize unbalanced dataset sampling employing evolutionary algorithms, reducing the bias of classifiers induced by such samples, common when classical classification algorithms are applied to unbalanced datasets.
- Choosing appropriately the objectives of the evolutionary algorithm used, the optimized samples can generate an ensemble of classifiers with better predictive performance than the existing methods in the literature that deal with the problem of imbalanced learning.

1.6 Thesis Organization

This section presents the organization of this thesis. Each chapter is self-contained, providing all the information the reader needs to understand the investigated research issue. Therefore, it can be read in the sequence that the reader wants. However, the sequence of chapters 2, 3 and 4 shows how the research proposals resulting from this doctorate evolved. Initially, the candidate proposed and investigated a method for imbalanced binary datasets, which is followed in the two next chapters by two methods for imbalanced multiclass datasets, all using evolutionary-based ensembles. Chapter 5 presents a proposal for the recognition of images from a real-world imbalanced multiclass dataset, using Convolutional Neural Networks (CNN) as base classifiers. Due to the inherent computational cost of CNN training, the proposal presented in this chapter applies an alternative to generate a pool of base classifiers. For ease of reading, a summary of each chapter follows.

1.6.1 Chapter 2

Title: *"An Evolutionary Sampling Approach for Classification with Imbalanced Data"*.

This chapter is an article written in collaboration with Dr. André Coelho (University of Fortaleza) and Dr. André de Carvalho (University of São Paulo). It proposes the Multiobjective Genetic Sampling (MOGASamp) method, which deals with imbalanced binary datasets. MOGASamp evolves balanced portions of the dataset as individuals of a customized multiobjective genetic algorithm, guided by the accuracy and diversity of the model generated by each sample, using the AUC and PFC metrics, respectively. The classification models represented by all individuals in the final population compose an ensemble of classifiers. When the classification system receives a new instance, the instance class is defined by the majority vote considering the output of each classifier.

The main contributions of this chapter are:

- The evolutionary algorithm has two mechanisms to increase the diversity of the samples produced during the evolutionary process:

- A measure of diversity of classifiers, which is explicitly inserted as one of the objectives of the multiobjective evolutionary algorithm.
- A mechanism that recognizes and eliminates solutions with a high degree of similarity.
- The proposed method considers that the imbalance rate of the dataset is not the main reason for the low accuracy in the minority class, but it can aggravate other situations found in the dataset. For this reason, the evolutionary process begins with balanced samples, but does not have any restriction to the unbalanced growth of the samples, considering that these samples do not produce classifiers with low accuracy in the minority class.

Follows the reference of the published article:

FERNANDES, E. R. Q.; CARVALHO, A. C. P. L. F. de; COELHO, A. L. V. An evolutionary-based approach for classification with imbalanced data. In: IEEE. Neural Networks (IJCNN), 2015 International Joint Conference on. [S.I.], 2015.p. 1-7.

1.6.2 Chapter 3

Title: "*Ensemble of Classifiers based on Multiobjective Genetic Sampling for Imbalanced Data*".

This chapter proposes a new evolutionary-based ensemble method, named Ensemble of Classifiers based on Multiobjective Genetic Sampling for Imbalanced Classification (E-MOSAIC). E-MOSAIC is an extension of the MOGASamp for imbalanced multiclass datasets. Like MOGASamp, E-MOSAIC evolves samples, initially balanced, extracted from the imbalanced dataset using a customized MOEA. However, in this method, the MOEA is guided by the accuracy of the sample-induced model for each class of the dataset. To promote diversity among classifiers, E-MOSAIC uses the PFC classifier diversity measure along with a process that eliminates twins solutions after the crossover process. However, E-MOSAIC uses the PFC as a secondary fitness, to deal with tie issues in the selection process of the multiobjective genetic algorithm. In addition, E-MOSAIC adopts a mechanism to maintain the best ensemble of classifiers generated during the evolutionary process.

The main contributions of this chapter are:

- EMOSAIC defines the predictive accuracy of the classifier in each class as the conflicting objectives of the customized MOEA. The solution was designed to consider the existence of several classes, and that there may be several classes of interest (positive classes). In such situations, increasing accuracy in one class may impair the correct classification of instances from other classes. Thus, the search is for samples that generate classifiers that present high accuracy for all classes.

- The ensemble obtained at the end of the evolutionary process is not necessarily what was produced in the last generation, but rather, the ensemble that presented the best accuracy during the evolutionary process. This mechanism was adopted taking into consideration that, although the individuals selected for the next generation present fitness values better or equal to the individuals of the current generation, this does not guarantee that the resulting ensemble of these individuals has better accuracy than the ensemble produced in the past generations.

This chapter is an article written in collaboration with Dr. Xin Yao (University of Birmingham) and Dr. André de Carvalho (University of São Paulo). This article was submitted to the international journal IEEE Transactions on Knowledge and Data Engineering - TKDE in May 2017.

1.6.3 Chapter 4

Title: *"Evolutionary Inversion of Class Distribution in Overlapping Areas for Multi-Class Imbalanced Learning"*.

This chapter presents another evolutionary-based ensemble method for multi-class imbalanced learning, named Evolutionary Inversion of Class Distribution for Imbalanced Learning (EVINCI). The evolutionary guidance of the proposed method is based on studies that indicate that the main difficulty experienced by classification algorithms for imbalanced datasets is related to overlapping areas. To address this issue, a dataset complexity measure, N1byClass, was proposed for use by EVINCI, which produces a matrix of values that estimates the percentage of overlap in each pair of classes. With the help provided by N1byClass and the accuracy of the model induced by the samples, EVINCI selectively reduces the concentration of less representative instances of the majority classes in the overlapping areas, while selecting samples that produce more accurate models.

The main contributions of this chapter are:

- EVINCI is the first ensemble method to consider that less complex samples of the original dataset, if it also considers aspects of accuracy and diversity of the classifiers induced by these samples, results in classifier ensembles with higher predictive performance for the imbalanced learning problem.
- Development of an extension of the N1 complexity measure, called N1byClass, which estimates the overlap percentage of each pair of classes.
- Proposal of a measure based on the class distribution of the dataset to systematically decide which classes are majorities and minorities in a multiclass dataset.

This chapter is an article written in collaboration with Dr. André de Carvalho (University of São Paulo). This research was submitted to the international journal Information Sciences in March 2018.

1.6.4 Chapter 5

Title: *"An Ensemble of Convolutional Neural Networks for Unbalanced Datasets: A case Study with Wagon Component Inspection"*.

This chapter proposes a method to build an ensemble of convolution neural network to deal with imbalanced image datasets, named Imbalanced Learning with Ensemble of Convolutional Neural Network (ILEC). The proposed method uses a customized undersampling method to construct a series of classifiers and applies a new pruning method based on a ranking of non-dominance to make the ensemble more accurate and with higher generalization ability.

The main contributions of this chapter are:

- Samples are generated by repeatedly applying random undersampling to the training dataset. However, to reach a set of diverse samples, even considering the minority class, ILEC selects only 80% of the most minority class in each sample.
- The method proposes a new pruning mechanism for ensembles of classifiers based on the ranking of non-dominance between the accuracy of the model generated by the sample and its diversity concerning the other models composing the ensemble.

This chapter is a paper written in collaboration with Rafael Rocha (Vale Technology Institute), Bruno Ferreira (SENAI Innovation Institute for Mineral Technologies), Eduardo Carvalho (SENAI Innovation Institute for Mineral Technologies), Ana Carolina Siravenha (SENAI Innovation Institute for Mineral Technologies), Ana Claudia Gomes (SENAI Innovation Institute for Mineral Technologies), Schubert Carvalho (Vale Technology Institute) and Cleidson de Souza (Federal University of Pará). It was submitted and accepted for oral presentation and publication at International Joint Conference on Neural Networks - IJCNN 2018 that will take place in July 2018.

1.6.5 Chapter 6

Chapter 6 presents the main conclusions from the research carried out in this thesis, discusses the main contributions from this thesis and points out future work directions.

1.6.6 Appendix A

In Appendix A, there is a comparison between the methods we propose to deal with the problem of imbalanced learning. Namely, MOGASamp (Chapter 2), EMOSAIC (Chapter 3)

and EVINCI (Chapter 4). The ILEC method, which will be presented in Chapter 5, deals with a dataset of images that presents an imbalance in the distribution of images by class. For this reason, the ILEC method is not part of the experiments presented in this Appendix.

1.7 Bibliography

ALI, A.; SHAMSUDDIN, S. M. H.; RALESCU, A. L. Classification with class imbalance problem: A review. In: . [S.l.: s.n.], 2015. Citations on pages 30 and 94.

BATISTA, G. E. A. P. A.; PRATI, R. C.; MONARD, M. C. A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explor. Newsl.*, ACM, New York, NY, USA, v. 6, n. 1, p. 20–29, Jun. 2004. ISSN 1931-0145. Available: <<http://doi.acm.org/10.1145/1007730.1007735>>. Citation on page 30.

BHOWAN, U. et al. Evolving Diverse Ensembles using Genetic Programming for Classification with Unbalanced Data. *IEEE Transactions on Evolutionary Computation*, v. 17, n. 3, p. 368–386, 2013. Citations on pages 31, 35, 36, 66, 67, 70, 94, 96, 98, 114, 116, and 117.

BREIMAN, L. Bagging predictors. *Machine Learning*, v. 24, n. 2, p. 123–140, 1996. Citations on pages 32, 50, 66, and 95.

BROWN, G. Ensemble learning. In: *Encyclopedia of Machine Learning and Data Mining*. [s.n.], 2017. p. 393–402. Available: <https://doi.org/10.1007/978-1-4899-7687-1_252>. Citation on page 32.

CHANDRA, A.; YAO, X. Ensemble learning using multi-objective evolutionary algorithms. *J. Math. Model. Algorithms*, v. 5, n. 4, p. 417–445, 2006. Available: <<http://dx.doi.org/10.1007/s10852-005-9020-3>>. Citations on pages 36, 48, 51, 63, 92, 98, and 117.

CHAWLA, N. et al. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, v. 16, p. 321–357, 2002. Citations on pages 29, 64, 65, 93, and 115.

CHAWLA, N. V. et al. Smoteboost: improving prediction of the minority class in boosting. In: *In Proceedings of the Principles of Knowledge Discovery in Databases, PKDD-2003*. [S.l.: s.n.], 2003. p. 107–119. Citation on page 33.

CHAWLA, N. V.; SYLVESTER, J. Exploiting diversity in ensembles: Improving the performance on unbalanced datasets. In: *Proceedings of the 7th International Conference on Multiple Classifier Systems*. Berlin, Heidelberg: Springer-Verlag, 2007. (MCS’07), p. 397–406. ISBN 978-3-540-72481-0. Available: <<http://dl.acm.org/citation.cfm?id=1761171.1761219>>. Citation on page 35.

COELLO, C. A. C. An updated survey of evolutionary multiobjective optimization techniques: state of the art and future trends. In: *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*. [S.l.: s.n.], 1999. v. 1, p. 13 Vol. 1. Citation on page 34.

DARWIN, C. *On the Origin of Species by Means of Natural Selection*. London: Murray, 1859. Or the Preservation of Favored Races in the Struggle for Life. Citation on page 34.

DIETTERICH, T. G. Machine-learning research – four current directions. *AI MAGAZINE*, v. 18, p. 97–136, 1997. Citations on pages 31, 48, 63, 67, 92, and 95.

DRUMMOND, C.; HOLTE, R. C. Exploiting the cost of (in)sensitivity of decision tree splitting criteria. In: *Proc. 17th International Conf. on Machine Learning*. [S.l.]: Morgan Kaufmann, San Francisco, CA, 2000. p. 239–246. Citation on page 30.

FAN, W.; STOLFO, S. J. Adacost: misclassification cost-sensitive boosting. In: *In Proc. 16th International Conf. on Machine Learning*. [S.l.]: Morgan Kaufmann, 1999. p. 97–105. Citations on pages 33 and 96.

FERNÁNDEZ, A. et al. Analysing the classification of imbalanced data-sets with multiple classes: Binarization techniques and ad-hoc approaches. *Knowledge-Based Systems*, v. 42, p. 97 – 110, 2013. ISSN 0950-7051. Available: <<http://www.sciencedirect.com/science/article/pii/S0950705113000300>>. Citations on pages 28, 62, and 92.

FREUND, Y.; SCHAPIRE, R. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, v. 55, n. 1, p. 119–139, 1997. Citations on pages 32, 50, 66, 79, 95, and 104.

GALAR, M. et al. A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, v. 42, n. 4, p. 463–484, Jul. 2012. Citations on pages 32, 33, 63, 95, and 114.

GREFENSTETTE, J. J. Incorporating problem specific knowledge into genetic algorithms. In: _____. *Genetic Algorithms and Simulated Annealing, London*. [S.l.: s.n.], 1987. p. 42–60. Citation on page 34.

HAIXIANG, G. et al. Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, v. 73, p. 220 – 239, 2017. ISSN 0957-4174. Available: <<http://www.sciencedirect.com/science/article/pii/S0957417416307175>>. Citation on page 28.

HANSEN, L. K.; SALAMON, P. Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.*, IEEE Computer Society, Washington, DC, USA, v. 12, n. 10, p. 993–1001, Oct. 1990. ISSN 0162-8828. Available: <<http://dx.doi.org/10.1109/34.58871>>. Citations on pages 31, 95, and 117.

KICINGER, R.; ARCISZEWSKI, T.; JONG, K. D. Evolutionary computation and structural design: A survey of the state-of-the-art. *Comput. Struct.*, Pergamon Press, Inc., Elmsford, NY, USA, v. 83, n. 23-24, p. 1943–1978, Sep. 2005. ISSN 0045-7949. Citation on page 34.

KITTLER, J. et al. On combining classifiers. *IEEE Trans. Pattern Anal. Mach. Intell.*, IEEE Computer Society, Washington, DC, USA, v. 20, n. 3, p. 226–239, Mar. 1998. ISSN 0162-8828. Available: <<http://dx.doi-org.ez67.periodicos.capes.gov.br/10.1109/34.667881>>. Citation on page 31.

KOVACS, T. Genetics-based machine learning. In: ROZENBERG, G.; BÄCK, T.; KOK, J. (Ed.). *Handbook of Natural Computing: Theory, Experiments, and Applications*. [S.l.]: Springer Verlag, 2012. p. 937–986. Citation on page 35.

KUBAT, M.; MATWIN, S. Addressing the curse of imbalanced training sets: One-sided selection. In: *In Proceedings of the Fourteenth International Conference on Machine Learning*. [S.l.]: Morgan Kaufmann, 1997. p. 179–186. Citations on pages 30, 49, 65, 93, and 116.

LING, C. X. et al. Decision trees with minimal costs. In: *Proceedings of the Twenty-first International Conference on Machine Learning*. New York, NY, USA: ACM, 2004. (ICML '04), p. 69–. ISBN 1-58113-838-5. Available: <<http://doi.acm.org/10.1145/1015330.1015369>>. Citation on page 30.

LIU, X.-Y.; WU, J.; ZHOU, Z.-H. Exploratory undersampling for class-imbalance learning. *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, v. 39, n. 2, p. 539–50, Apr. 2009. Citation on page 33.

LIU, Y.; YAO, X. Negatively correlated neural networks can produce best ensembles. *Australian Journal of Intelligent Information Processing Systems*, v. 4, n. 3/4, p. 176–185, 1997. Citations on pages 36, 48, and 63.

MITCHELL, T. M. *Machine Learning*. 1. ed. New York, NY, USA: McGraw-Hill, Inc., 1997. ISBN 0070428077, 9780070428072. Citation on page 27.

OZA, N. C.; TUMER, K. Classifier ensembles: Select real-world applications. *Information Fusion*, v. 9, p. 4–20, 2008. Citation on page 31.

QIAN, Y. et al. A resampling ensemble algorithm for classification of imbalance problems. *Neurocomputing*, Elsevier, v. 143, p. 57–67, Nov. 2014. Citations on pages 31, 66, 94, and 114.

RIFKIN, R.; KLAUTAU, A. In defense of one-vs-all classification. *J. Mach. Learn. Res.*, JMLR.org, v. 5, p. 101–141, Dec. 2004. ISSN 1532-4435. Available: <<http://dl.acm.org/citation.cfm?id=1005332.1005336>>. Citations on pages 29 and 93.

SÁEZ, J. A.; KRAWCZYK, B.; WOŹNIAK, M. Analyzing the oversampling of different classes and types of examples in multi-class imbalanced datasets. *Pattern Recognition*, v. 57, n. Supplement C, p. 164 – 178, 2016. ISSN 0031-3203. Available: <<http://www.sciencedirect.com/science/article/pii/S0031320316001072>>. Citation on page 28.

SCHÖLKOPF, B. et al. Estimating the support of a high-dimensional distribution. *Neural Computation*, v. 13, n. 7, p. 1443–1471, 2001. Citations on pages 30, 65, and 94.

SEIFFERT, C.; KHOSHGOFTAAR, T. M.; Van Hulse, J. Hybrid sampling for imbalanced data. In: *Integrated Computer-Aided Engineering*. [s.n.], 2009. v. 16, n. 3, p. 193–210. Available: <<http://www.scopus.com/inward/record.url?eid=2-s2.0-68249098324&partnerID=tZOtx3y1>>. Citation on page 30.

SEIFFERT, C. et al. RUSBoost: A Hybrid Approach to Alleviating Class Imbalance. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, v. 40, n. 1, p. 185–197, Jan. 2010. Citations on pages 33, 96, and 104.

SUN, B. et al. Evolutionary under-sampling based bagging ensemble method for imbalanced data classification. *Frontiers of Computer Science*, Jul 2017. ISSN 2095-2236. Available: <<https://doi.org/10.1007/s11704-016-5306-z>>. Citation on page 36.

SUN, Y. et al. Cost-sensitive boosting for classification of imbalanced data. *Pattern Recogn.*, Elsevier Science Inc., New York, NY, USA, v. 40, n. 12, p. 3358–3378, Dec. 2007. ISSN 0031-3203. Available: <<http://dx.doi.org/10.1016/j.patcog.2007.04.009>>. Citations on pages 27, 33, 62, 64, and 115.

SUN, Y.; WONG, A. K. C.; KAMEL, M. S. Classification of imbalanced data: a review. *IJPRAI*, v. 23, n. 4, p. 687–719, 2009. Available: <<http://dx.doi.org/10.1142/S02180014090007326>>. Citations on pages 31 and 49.

VLADISLAVLEVA, E.; SMITS, G.; HERTOOG, D. den. On the importance of data balancing for symbolic regression. *IEEE Trans. Evolutionary Computation*, v. 14, n. 2, p. 252–277, 2010. Citation on page 27.

WANG, J. et al. Ensemble of Cost-Sensitive Hypernetworks for Class-Imbalance Learning. In: *2013 IEEE International Conference on Systems, Man, and Cybernetics*. [S.l.]: IEEE, 2013. p. 1883–1888. Citations on pages 31, 66, 75, and 114.

WANG, S. Ensemble diversity for class imbalance learning. 2011. Available: <<http://etheses.bham.ac.uk/1793/>>. Citation on page 30.

WANG, S.; YAO, X. Diversity analysis on imbalanced data sets by using ensemble models. In: *CIDM*. [S.l.]: IEEE, 2009. p. 324–331. Citations on pages 33, 95, and 104.

WANG, S.; YAO, X. Multiclass imbalance problems: Analysis and potential solutions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, v. 42, n. 4, p. 1119–1130, Aug 2012. ISSN 1083-4419. Citations on pages 29, 67, 93, and 116.

WONG, K. Evolutionary algorithms: Concepts, designs, and applications in bioinformatics: Evolutionary algorithms for bioinformatics. *CoRR*, abs/1508.00468, 2015. Available: <<http://arxiv.org/abs/1508.00468>>. Citation on page 34.

YIN, Q.-Y. et al. A Novel Selective Ensemble Algorithm for Imbalanced Data Classification Based on Exploratory Undersampling. *Mathematical Problems in Engineering*, Hindawi Publishing Corporation, v. 2014, p. 1–14, 2014. Citations on pages 31, 33, 66, 71, 73, 75, 94, 100, 114, and 117.

ZITZLER, E.; LAUMANN, M.; BLEULER, S. A Tutorial on Evolutionary Multiobjective Optimization. In: GANDIBLEUX, X. (Ed.). *Metaheuristics for Multiobjective Optimisation*. [S.l.]: Springer, 2004. (Lecture Notes in Economics and Mathematical Systems). Citation on page 34.

AN EVOLUTIONARY SAMPLING APPROACH FOR CLASSIFICATION WITH IMBALANCED DATA

Authors:

Everlandio R. Q. Fernandes (everlandio@usp.br)

Andre C. P. L. de Carvalho (andre@icmc.usp.br)

Andre L. V. Coelho (acoelho@unifor.br)

Abstract

In some practical classification problems in which the number of instances of a particular class is much lower/higher than the instances of the other classes, one commonly adopted strategy is to train the classifier over a small, balanced portion of the training data set. Although straightforward, this procedure may discard instances that could be important for the better discrimination of the classes, affecting the performance of the resulting classifier. To address this problem more properly, in this paper we present MOGASamp (after Multiobjective Genetic Sampling) as an adaptive approach that evolves a set of samples of the training data set to induce classifiers with optimized predictive performance. More specifically, MOGASamp evolves balanced portions of the data set as individuals of a multiobjective genetic algorithm aiming at achieving a set of induced classifiers with high levels of diversity and accuracy. Through experiments involving eight binary classification problems with varying levels of class imbalance, the performance of MOGASamp is compared against the performance of six traditional methods. The overall results show that the proposed method have achieved a noticeable performance in terms of accuracy measures.

2.1 Introduction

Several classification problems present data with imbalanced class distributions. Such an imbalance occurs naturally in some practical applications, such as in financial data (MARQUÉS; GARCÍA; SÁNCHEZ, 2013), where the number of instances in the “default” class (minority class) is generally lower than the number of instances in the “non-default” class (majority class). Imbalanced data sets may affect the predictive performance of some classical classification algorithms because these algorithms assume that the data has a balanced distribution of classes and that the same cost of misclassification applies to all classes (HE; GARCIA, 2009).

A commonly strategy used for classification with imbalanced data sets is to select a balanced set of instances from each class. This means that the number of instances of the minority class will be equal to the number of instances of the majority class. This strategy is used to generate a classification model that is not detrimental to the minority class. However, this procedure may not be effective in some cases, since the final classification model may not take into account relevant instances for better discrimination between the classes, leading to a decrease in the predictive accuracy of the classifier.

In order to overcome this problem, ensembles of classifiers have been considered. Ensembles are designed to increase the accuracy of a single classifier by inducing separately a set of hypotheses and combining their decisions by some consensus operator (ZHOU, 2009). The generalization ability of an ensemble is usually higher than that of a single classifier. In (TUMER; GHOSH, 1996) the authors present a formal demonstration of this. Although ensembles tend to perform better than their members, their construction is not an easy task. According to (DIETTERICH, 1997), an ensemble of classifiers 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 in different instances of the same test set i.e., they should disagree as much as possible in their outcomes (KROGH; VEDELSBY, 1995).

Therefore, *diversity* and *accuracy* are the two main criteria that should be taken into account to generate an effective ensemble of classifiers. In this context, some metrics have been proposed to measure the diversity of the classifiers, such as the Pairwise Failure Crediting (PFC) (CHANDRA; YAO, 2006) and the negative correlation in the Negative Correlation Learning (NCL) approach (LIU; YAO, 1997). However, there is a trade-off on what should be the optimal measures of diversity and accuracy, since these are two conflicting criteria (CHANDRA; YAO, 2004). To handle this situation, the use of Multiobjective Evolutionary Algorithms (MOEA) seems to be an interesting solution, since MOEA can deal nicely with conflicting objectives in the learning process. Such algorithms evolve simultaneously a set (aka front) of non-dominated solutions over two or more objectives, without requiring the imposition of preferences on the objectives. In the case of ensembles of classifiers, the objectives are the two criteria of accuracy

and diversity.

In this context, this paper proposes a new approach, namely MOGASamp (*Multiobjective Genetic Sampling*), to deal with the problem of imbalanced data sets aiming at an ensemble of classifiers with high predictive performance. The goal of MOGASamp is to construct an ensemble of classifiers induced from balanced samples of the training data set. For this, a customized MOEA will evolve combinations of instances in balanced samples, guided by the performance of the classifiers induced by these samples. This strategy allows one to obtain a set of balanced samples from the imbalanced data set that induces classifiers with high accuracy and diversity.

In order to assess the novel approach, experimental tests were performed using several different imbalanced data sets. Comparative evaluations have demonstrated that MOGASamp can outperform traditional algorithms, such as AdaBoost and Bagging, being a good option for dealing with imbalanced data sets.

The remainder of this paper is structured as follows: Section 2 provides a review of related work. Section 3 discusses the evaluation metrics considered in this work. Section 4 introduces the main ingredients of the MOGASamp technique. Section 5 shows the experimental analysis and Section 6 concludes the paper.

2.2 Literature Review

Most of the studies found in the literature for classification with imbalanced data sets rely on two approaches (DEEPA; PUNITHAVALLI, 2010). The first approach allocates different costs to classes during the induction of the classification model (ZADROZNY; LANGFORD; ABE, 2003). The second approach is based on data resampling (subsampling or oversampling). In subsampling, instances from the majority class are removed, while oversampling the instances of the minority class are replicated or synthetic data are generated.

Although using a simple strategy, the subsampling approach when performed randomly may discard important data. To address this problem, a directed subsampling method can be used to detect and eliminate less representative portions of the data. This is the strategy used by the One-Sided Selection (OSS) technique (KUBAT; MATWIN, 1997), which removes instances from the majority class that are redundant, noisy, and/or close to the boundary between the classes. The border instances are detected by applying Tomek links, and the instances that are distant from the decision boundary (redundant instances) are discovered by Condensed Nearest Neighbor (CNN) (HART, 1968).

On the other hand, considering the oversampling approach, the replication of instances tends to increase the computational cost of the process (SUN; WONG; KAMEL, 2009). These approaches can be categorized into random or classic oversampling and synthetic oversampling.

The classic oversampling method is a non-heuristic method that add instances through the random replication of the minority class instances. This kind of oversampling sometimes creates very specific rules, leading to overfitting (HOLTE; ACKER; PORTER, 1989). However, synthetic oversampling methods add instances by generating synthetic minority class instances. The generated instances add essential information to the original dataset that may help improve the classifiers' performance. The interpolation technique is commonly used to generate synthetic data, such as in SMOTE (Synthetic Minority Oversampling Technique) (CHAWLA et al., 2002). SMOTE finds the k nearest neighbors of each instance of the minority class and, then, synthetic instances are generated along the line that connects the instance with its k nearest neighbors. Although SMOTE has proved to be an effective tool for handling the class imbalance, it may overgeneralize the minority class, once it does not consider the distribution of majority class neighbors. As a result, it may increase the overlapping between classes (GARCÍA; MARQUÉS; SÁNCHEZ, 2012).

Other studies use ensembles of classifiers to deal with imbalanced data set, such as Bagging (BREIMAN, 1996) and Boosting (FREUND; SCHAPIRE, 1997). The Bagging approach trains a set of base classifiers with different samples of the training data set. The sampling is performed with replacement and each sample has the same size of the original training data set. After obtaining the base classifiers, it combines them by majority voting, and the most voted class is predicted for a new instance.

The AdaBoost approach, the most representative algorithm in the Boosting family, uses the whole training data set to create classifiers serially. In each iteration, AdaBoost gives more emphasis to the instances that were incorrectly classified in the previous iteration. For this, the weights of incorrectly classified instances are increased and the weights of correctly classified instances are decreased. Finally, when a new instance is presented, each base classifier gives its vote, weighted by its overall accuracy, and the label of the new instance is selected based on the majority of votes.

2.3 Performance Evaluation

This section presents measures to evaluate the performance of the classifiers in imbalanced domains and to evaluate the diversity of an ensemble of classifiers.

2.3.1 Accuracy

An effective measure to evaluate the performance of a classifier is the rate of classification errors made in each class (HE; GARCIA, 2009). Such measure can be obtained using a confusion matrix. Each column of this matrix represents the instances in a predicted class, while each row represents the instances in an actual class. Elements along the main diagonal represent the correct classifications, number of true negatives (TN) and true positives (TP), while the

Database	YeastME1	YeastMit	YeastME3	Spect	Ion	German	Haberman	CMC
Imbalance Ratio	1:33	1:5	1:8	1:4	1:2	1:2	1:3	1:3
Total of Instances	1484	1484	1484	267	351	1000	306	1473

Table 1 – Databases Used for the Experimental Tests

off-diagonal elements represent the classification errors, number of false positives (FP) and false negatives (FN). From the confusion matrix, it is possible to extract two independent measures: True Positive Rate (Eq. 2.1) and True Negative Rate (Eq. 2.2). These two measures evaluate the performance on the positive (minority) and negative (majority) classes, respectively.

$$TP_r = \frac{TP}{TP + FN} \quad (2.1)$$

$$TN_r = \frac{TN}{TN + FP} \quad (2.2)$$

However, the goal is to achieve a good prediction in both classes (minority and majority) when dealing with a binary classification problem. So, it is necessary to combine these individual measures (TP_r and TN_r), since they are not useful when used alone. These measures are combined by the Receiver Operating Characteristic (ROC) curve (BRADLEY, 1997), which shows the relationship between the benefits and the classification costs in relation to the distribution of the data. So, we say that a classification model is better than another if its ROC curve dominates the other. When it is necessary to encode the ROC curve into a single scalar value, the most common strategy is to calculate the Area under the ROC Curve (AUC) (PROVOST; FAWCETT, 1997).

2.3.2 Diversity

If we have a perfect classifier that makes no errors, then we do not need an ensemble. If, however, the classifier does make some errors, then we can try to complement it with another classifier, which makes errors on different objects. Therefore, as mentioned earlier, the success of an ensemble depends on the diversity of the prediction errors generated by their base classifiers.

The diversity of an ensemble can be measured in two different ways: 1) considering the diversity of a pair of classifiers and then the average is obtained for all pairs diversity (Pairwise Measures); or 2) considering all the classifiers together and calculating a unique diversity value of the ensemble (Non-pairwise Measures) (KUNCHEVA, 2004).

The Pairwise Failure Crediting (PFC) measure (CHANDRA; YAO, 2006) calculates the distance between the failure patterns taking each pair of individuals. A failure pattern is a string of 0s and 1s indicating success or failure of the classifier. The accumulated differences on each individuals in the ensemble is used to compute the diversity of the individual members with respect to the ensemble, i.e. how different a member is with respect to others in the ensemble.

Figure 2 – MOGASamp - Multiobjective Genetic Sampling

The PFC measure has been employed in MOGASamp to achieve a set of samples that induce classifiers with good performance when dealing with imbalanced data sets.

2.4 MOGASamp

The goal of the proposed method is to generate a set of samples to induce base classifiers that will compose an ensemble of classifiers, so that that these classifiers have high accuracy with the most diversity as possible. For this purpose, MOGASamp adopts a multiobjective genetic algorithm to evolve a selection of samples and to evaluate the classifiers induced by these samples based on accuracy and diversity.

Figure 2 outlines the proposed method. In the following subsections, we will detail each of its main steps.

2.4.1 Sampling and the Training Models

In the first step we obtain n balanced samples from the training data set. This means that each sample will have an equal amount of instances of each class. The size of the samples is chosen based on the number of instances of the minority class. However, we use only 90% of the instances of the minority class to compose the samples. The remain 10% of the instances are used to perform the validation in the evolution process (refer to the next step). These samples are obtained without replacement. Each sample represents an individual of the population of the Genetic Algorithm (GA). These samples are encoded by a vector with dimensionality equal to the sample size. The cells of this vector represent the training instances that are part of the sample, and for each individual an SVM model is generated.

2.4.2 Evaluation

The obtained SVM model of each individual is validated using the instances that were not used to obtain the samples, i.e. the remaining 10% of the instances of the minority class and the other instances of the majority class. The AUC metric is calculated based on the performance of this model over the validation data. The PFC is also calculated for each individual using a pair-wise comparison with all individuals of the current population.

In this MOGASamp, the fitness of each individual is given by the AUC and PFC. These two metrics are used to compose a dominance rank (DEB et al., 2002) of the solutions. The dominance rank of a given solution is the number of other solutions in the population that dominate it. A solution x_1 is said to dominate another solution x_2 if 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 (DEB, 2001). A nondominated solution will have the best fitness of 0, while high fitness values indicate poor-performing solutions, i.e., solutions dominated by many individuals.

2.4.3 Genetic Operators

The dominance rank is used to select the 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 that the winner will be the one with highest AUC. The quantity of parents selected will be equal to the quantity of individuals of the current population. For each selected pair of parents, two new individuals are generated by merging the instances from the minority class of a parent with the instances from the majority class of another parent, and vice-versa. The mutation occurs in a percentage of the offspring generated. The instances of a random portion of the sample that represents an individual is changed by a new sampling, maintaining the proportion of classes.

2.4.4 Elimination of Identical Solutions

After applying the genetic operators, identical individuals can occur, especially when the imbalance ratio is not high (less than 1:6). This fact was observed in our experimental tests. Identical individuals with high fitness have a higher probability of being selected for reproduction and for future generations, increasing even further the number of replicated solutions. However, the goal of this work is to have a diverse ensemble of classifiers with a high accuracy. For this reason, after reproduction, identical individuals are eliminated. Afterwards, if the number of individuals is less than the initial population size, a new reproduction and mutation process is performed.

2.4.5 New Generation and Stop Criterion

The selection of the individuals that will compose the new generation is based on the non-dominance 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. This process repeats until the fixed number of generations and/or the maximum AUC value is reached, i.e. $AUC = 1.0$.

The classification models of all individuals in the final population compose the ensemble of classifiers. When a new instance is presented to the classifiers, its class is determined by majority voting considering the output of all classifiers.

2.5 Experimental results

Eight binary classification data sets with different imbalance ratio were used in our empirical assessment. These data sets were obtained from the UCI Repository ([BACHE; LICHMAN, 2013](#)) and are summarized in Table 3. For each data set, half of the instances of each class were randomly chosen for the training set and the other half as the test set. This ensures that both the training and test sets maintain the same class proportion as in the original data set.

MOGASamp was compared against six well-known resampling and classification techniques from the literature. The resampling techniques used were: SMOTE; Classical subsampling (random); Directed subsampling (OSS); and Classical oversampling. The classification techniques used were: Bagging and AdaBoost. For the resampling algorithms, after the process of rebalancing the classes, the SVM algorithm was used to generate the classification model.

MOGASamp was performed with a population of 40 individuals, a maximum of 20 generation and 5% as mutation rate. The SMOTE parameters used were: 200% as the percentage of oversampling and undersampling, 5 as the number of the neighbors. These are the default parameter values specified in the package used ([TORGO, 2010](#)). The classic Subsampling and Oversampling were performed until resulting in a balanced dataset. For the OSS technique, it is not necessary to set any parameter. The package used for OSS, classic Subsampling and Oversampling was that available in ([POZZOLO; CAELEN; BONTEMPI, 2014](#)). For Bagging and AdaBoost 100 iterations were used, as well as the standard configuration of the package used ([ALFARO; GÁMEZ; GARCÍA, 2013](#)).

Table 2 shows the AUC, True Positive rate (minority class) and True Negative rate (Majority class) values obtained by the evaluated techniques in each data set used. The values presented are the mean and standard deviation after running 30 times each algorithm. We highlight in bold the highest value for each measure.

In order to statistically validate the obtained results, we present the results of statistical tests by following the approach proposed by Demšar ([DEMsAR, 2006](#)). In brief, this approach

		AUC	AccMin	AccMaj
YeastME1	MOGASamp	0.960 [0.001]	1.000 [0.000]	0.918 [0.003]
	SMOTE	0.960 [0.008]	0.995 [0.013]	0.929 [0.021]
	Over Sampling	0.831 [0.011]	0.675 [0.022]	0.988 [0.001]
	Under Sampling	0.951 [0.009]	1.000 [0.000]	0.902 [0.019]
	Bagging	0.772 [0.042]	0.547 [0.086]	0.997 [0.001]
	AdaBoost	0.835 [0.025]	0.675 [0.051]	0.995 [0.001]
	OSS	0.838 [0.000]	0.681 [0.000]	0.994 [0.000]
YeastMit	MOGASamp	0.764 [0.007]	0.627 [0.015]	0.900 [0.007]
	SMOTE	0.764 [0.007]	0.618 [0.019]	0.910 [0.008]
	Over Sampling	0.741 [0.005]	0.584 [0.010]	0.898 [0.010]
	Under Sampling	0.753 [0.008]	0.648 [0.029]	0.859 [0.022]
	Bagging	0.682 [0.009]	0.385 [0.019]	0.979 [0.001]
	AdaBoost	0.694 [0.008]	0.435 [0.016]	0.953 [0.003]
	OSS	0.736 [0.000]	0.508 [0.000]	0.964 [0.000]
YeastME3	MOGASamp	0.910 [0.004]	0.913 [0.010]	0.905 [0.008]
	SMOTE	0.885 [0.011]	0.822 [0.022]	0.949 [0.007]
	Over Sampling	0.862 [0.005]	0.785 [0.008]	0.938 [0.003]
	Under Sampling	0.896 [0.008]	0.894 [0.030]	0.898 [0.025]
	Bagging	0.885 [0.008]	0.802 [0.018]	0.968 [0.002]
	AdaBoost	0.858 [0.009]	0.751 [0.018]	0.966 [0.002]
	OSS	0.852 [0.000]	0.731 [0.000]	0.974 [0.000]
Spect	MOGASamp	0.680 [0.000]	0.408 [0.000]	0.953 [0.000]
	SMOTE	0.672 [0.006]	0.383 [0.018]	0.961 [0.006]
	Over Sampling	0.678 [0.005]	0.403 [0.011]	0.953 [0.002]
	Under Sampling	0.679 [0.005]	0.401 [0.013]	0.957 [0.005]
	Bagging	0.684 [0.010]	0.512 [0.045]	0.856 [0.037]
	AdaBoost	0.699 [0.011]	0.501 [0.022]	0.896 [0.019]
	OSS	0.666 [0.000]	0.370 [0.000]	0.962 [0.000]
Ion	MOGASamp	0.966 [0.003]	0.985 [0.003]	0.944 [0.004]
	SMOTE	0.967 [0.003]	0.996 [0.007]	0.938 [0.005]
	Over Sampling	0.959 [0.002]	0.950 [0.005]	0.968 [0.004]
	Under Sampling	0.949 [0.012]	0.986 [0.007]	0.912 [0.024]
	Bagging	0.906 [0.005]	0.872 [0.003]	0.941 [0.012]
	AdaBoost	0.924 [0.005]	0.865 [0.010]	0.984 [0.006]
	OSS	0.943 [0.000]	0.968 [0.000]	0.919 [0.000]
German	MOGASamp	0.960 [0.003]	1.000 [0.000]	0.917 [0.007]
	SMOTE	0.994 [0.001]	1.000 [0.000]	0.989 [0.002]
	Over Sampling	0.954 [0.005]	0.908 [0.011]	1.000 [0.000]
	Under Sampling	0.918 [0.048]	1.000 [0.000]	0.836 [0.090]
	Bagging	0.805 [0.005]	0.647 [0.008]	0.963 [0.004]
	AdaBoost	1.000 [0.000]	1.000 [0.000]	1.000 [0.000]
	OSS	0.994 [0.000]	0.995 [0.000]	0.993 [0.000]
Haberman	MOGASamp	0.632 [0.008]	0.526 [0.009]	0.735 [0.013]
	SMOTE	0.609 [0.019]	0.441 [0.057]	0.777 [0.037]
	Over Sampling	0.632 [0.019]	0.515 [0.044]	0.750 [0.060]
	Under Sampling	0.617 [0.029]	0.536 [0.049]	0.698 [0.087]
	Bagging	0.596 [0.009]	0.287 [0.019]	0.906 [0.006]
	AdaBoost	0.601 [0.013]	0.400 [0.025]	0.802 [0.013]
	OSS	0.645 [0.000]	0.425 [0.000]	0.866 [0.000]
CMC	MOGASamp	0.655 [0.005]	0.626 [0.011]	0.684 [0.009]
	SMOTE	0.643 [0.011]	0.511 [0.024]	0.775 [0.016]
	Over Sampling	0.646 [0.009]	0.512 [0.016]	0.779 [0.017]
	Under Sampling	0.648 [0.011]	0.620 [0.028]	0.676 [0.022]
	Bagging	0.581 [0.005]	0.210 [0.013]	0.952 [0.003]
	AdaBoost	0.596 [0.009]	0.315 [0.019]	0.878 [0.011]
	OSS	0.608 [0.000]	0.295 [0.000]	0.922 [0.000]

Table 2 – AUC and classification accuracy of the Minority and Majority classes (average and standard deviation) using different resampling and classification techniques

seeks to compare multiple algorithms on multiple data sets, and it is based on the use of 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.

The results suggest that MOGASamp achieved the best overall performance. The ranking provided by the Friedman test supports this assumption, showing MOGASamp as the best-ranked method on AUC and True Positive rate and sixth best-ranked method on True Negative rate. The Friedman test also indicates the rejection of the null hypothesis, confirming that the differences among the algorithms are statistically significant (AUC: $p\text{-value} = 0.0071$, AccMin: $p\text{-value} = 2.74 \times 10^{-5}$, AccMaj: $p\text{-value} = 9.40 \times 10^{-5}$). Hence, we have executed the Nemenyi post-hoc test for the purpose of pairwise comparison. The proposed method outperforms the Bagging and OSS on True Positive rate and outperforms the Bagging on AUC with statistical significance at a 95% confidence level.

AUC was used as a measure for evaluating the performance of each approach considering both classes. It can be seen that MOGASamp presented the best values in six of the data sets, and in the data set *Ion* it achieved a performance statistically similar to that of SMOTE. Another important aspect to be highlighted is that the proposed method did not present the worst AUC value in any of the evaluated data sets. This is an indication that MOGASamp can be applied to different data sets with different imbalance ratio, even without a priori knowledge.

As discussed previously, an effective way to evaluate a classifier with imbalanced data is using the rates of classification errors made in each class, since traditional classification algorithms tend to favor the majority class rather than the minority class. Based on this fact, it can be seen in Table 2 that the proposed method presented a trade-off between the rates of True Positives and True Negatives.

Analyzing the true positive values, MOGASamp achieved the best results in four data sets. In fact, considering the data sets YeastME3 and CMC, MOGASamp presented the highest values in experimental tests. These results were obtained without sacrificing the accuracy of the majority class (True Negative rate). The undersampling technique achieved good True Positive values, albeit it presented low True Negative values.

When analyzing the True Negative rate values, one can observe that Bagging presents significant results. However, Bagging does not present good values for True Positive rate. This fact indicates that this method does not have a good performance when dealing with imbalanced data. The main reason of this drawback is the fact that Bagging, as well as other classical classification algorithms, assumes that the data is balanced, then giving preference to the majority class. Similar situation can be observed in the AdaBoost and OSS methods.

2.6 Conclusion

In this paper, we presented a new evolutionary approach, called MOGASamp (Multiobjective Genetic Sampling), to address the problem of classification with imbalanced data sets.

This approach is based on a multiobjective genetic algorithm. It uses two metrics AUC and PFC to evolve a set of balanced samples of the training data set, until a set of classifiers with high accuracy and diversity is reached. The obtained classifiers are used as an ensemble of classifiers to predict new instances using majority voting.

Experimental tests were performed using eight data sets and the obtained results were compared with well known resampling and classification techniques from the literature. The experimental results has shown that MOGASamp presents high predictive accuracy, obtaining better results in six of the data sets. Furthermore, MOGASamp has also shown high stability to predict the data of both classes. This means that MOGASamp does not show noticeable differences in the rate of True Positives and True Negatives, which is the main drawback of classical algorithms when dealing with imbalanced data.

Acknowledgment

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

2.7 Bibliography

ALFARO, E.; GÁMEZ, M.; GARCÍA, N. *adabag: An R Package for Classification with Boosting and Bagging*. 2013. 1–35 p. Available: <<http://www.jstatsoft.org/v54/i02/>>. Citation on page 54.

BACHE, K.; LICHMAN, M. *UCI Machine Learning Repository*. 2013. Available: <<http://archive.ics.uci.edu/ml>>. Citations on pages 54, 64, 72, 73, and 103.

BRADLEY, A. P. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, v. 30, p. 1145–1159, 1997. Citations on pages 51 and 73.

BREIMAN, L.; BREIMAN, L. Bagging predictors. In: *Machine Learning*. [S.l.: s.n.], 1996. p. 123–140. Citations on pages 32, 50, 66, and 95.

CHANDRA, A.; YAO, X. DIVACE: diverse and accurate ensemble learning algorithm. In: *Intelligent Data Engineering and Automated Learning - IDEAL 2004, 5th International Conference, Exeter, UK, August 25-27, 2004, Proceedings*. [S.l.: s.n.], 2004. p. 619–625. Citation on page 48.

CHANDRA, A.; YAO, X. Ensemble learning using multi-objective evolutionary algorithms. *J. Math. Model. Algorithms*, v. 5, n. 4, p. 417–445, 2006. Citations on pages 36, 48, 51, 63, 92, 98, and 117.

CHAWLA, N. V. et al. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, v. 16, p. 321–357, 2002. Citation on page 50.

DEB, K. *Multi-Objective Optimization using Evolutionary Algorithms*. [S.l.]: John Wiley & Sons, Chichester, 2001. (Wiley-Interscience Series in Systems and Optimization). Citations on pages 53 and 69.

DEB, K. et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. *Trans. Evol. Comp.*, IEEE Press, Piscataway, NJ, USA, v. 6, n. 2, p. 182–197, Apr. 2002. ISSN 1089-778X. Citations on pages 53, 67, 98, and 117.

DEEPA, T.; PUNITHAVALLI, M. An analysis for mining imbalanced datasets. *International Journal of Computer Science and Information Security*, v. 8, p. 132–137, 2010. Citations on pages 49 and 62.

DEMsAR, J. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.*, JMLR.org, v. 7, p. 1–30, Dec. 2006. ISSN 1532-4435. Citations on pages 54, 76, 77, and 132.

DIETTERICH, T. G. Machine-learning research – four current directions. *AI Magazine*, v. 18, p. 97–136, 1997. Citations on pages 31, 48, 63, 67, 92, and 95.

FREUND, Y.; SCHAPIRE, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.*, Academic Press, Inc., Orlando, FL, USA, v. 55, n. 1, p. 119–139, Aug. 1997. ISSN 0022-0000. Citations on pages 32, 50, 66, 79, 95, and 104.

GARCÍA, V.; MARQUÉS, A. I.; SÁNCHEZ, J. S. Improving risk predictions by preprocessing imbalanced credit data. In: HUANG, T. et al. (Ed.). *ICONIP (2)*. [S.l.]: Springer, 2012. (Lecture Notes in Computer Science, v. 7664), p. 68–75. Citation on page 50.

HART, P. E. The condensed nearest neighbor rule (corresp.). *IEEE Transactions on Information Theory*, v. 14, n. 3, p. 515–516, 1968. Citations on pages 49, 65, and 93.

HE, H.; GARCIA, E. A. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, IEEE Computer Society, Los Alamitos, CA, USA, v. 21, n. 9, p. 1263–1284, 2009. ISSN 1041-4347. Citations on pages 48, 50, and 73.

HOLTE, R. C.; ACKER, L.; PORTER, B. W. Concept learning and the problem of small disjuncts. In: SRIDHARAN, N. S. (Ed.). *IJCAI*. [S.l.]: Morgan Kaufmann, 1989. p. 813–818. ISBN 1-55860-094-9. Citation on page 50.

KROGH, A.; VEDELSBY, J. Neural network ensembles, cross validation, and active learning. In: *Advances in Neural Information Processing Systems*. [S.l.]: MIT Press, 1995. p. 231–238. Citations on pages 48, 63, and 67.

KUBAT, M.; MATWIN, S. Addressing the curse of imbalanced training sets: One-sided selection. In: *In Proceedings of the Fourteenth International Conference on Machine Learning*. [S.l.]: Morgan Kaufmann, 1997. p. 179–186. Citations on pages 30, 49, 65, 93, and 116.

KUNCHEVA, L. I. *Combining Pattern Classifiers: Methods and Algorithms*. [S.l.]: Wiley-Interscience, 2004. ISBN 0471210781. Citation on page 51.

LIU, Y.; YAO, X. Negatively correlated neural networks can produce best ensembles. *Australian Journal of Intelligent Information Processing Systems*, v. 4, n. 3/4, p. 176–185, 1997. Citations on pages 36, 48, and 63.

MARQUÉS, A. I.; GARCÍA, V.; SÁNCHEZ, J. S. On the suitability of resampling techniques for the class imbalance problem in credit scoring. *Journal of the Operational Research Society*, v. 64, n. 7, p. 1060–1070, 2013. Citations on pages 48 and 62.

POZZOLO, A. D.; CAELEN, O.; BONTEMPI, G. *unbalanced: The package implements different data-driven method for unbalanced datasets*. 2014. Available: <http://cran.r-project.org/bin/windows/contrib/3.3/unbalanced_1.1.zip>. Citation on page 54.

PROVOST, F. J.; FAWCETT, T. Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In: HECKERMAN, D.; MANNILA, H.; PREGIBON, D. (Ed.). *KDD*. [S.l.: s.n.], 1997. p. 43–48. Citations on pages 51 and 73.

SUN, Y.; WONG, A. K. C.; KAMEL, M. S. Classification of imbalanced data: a review. *International Journal of Pattern Recognition and Artificial Intelligence*, v. 23, n. 4, p. 687–719, 2009. Citations on pages 31 and 49.

TORGO, L. *DMwR: Functions and data for “Data Mining with R”*. 2010. Available: <http://cran.r-project.org/bin/windows/contrib/3.3/DMwR_0.4.1.zip>. Citation on page 54.

TUMER, K.; GHOSH, J. Analysis of decision boundaries in linearly combined neural classifiers. *Pattern Recognition*, v. 29, p. 341–348, 1996. Citations on pages 48, 63, 66, 91, and 115.

ZADROZNY, B.; LANGFORD, J.; ABE, N. Cost-sensitive learning by cost-proportionate example weighting. In: *Proceedings of the Third IEEE International Conference on Data Mining*. [S.l.]: IEEE Computer Society, 2003. (ICDM '03), p. 435–442. ISBN 0-7695-1978-4. Citation on page 49.

ZHOU, Z.-H. Ensemble learning. In: LI, S. Z.; JAIN, A. K. (Ed.). *Encyclopedia of Biometrics*. [S.l.]: Springer US, 2009. p. 270–273. ISBN 978-0-387-73003-5. Citations on pages 48, 63, 66, and 115.

ENSEMBLE OF CLASSIFIERS BASED ON MULTIOBJECTIVE GENETIC SAMPLING FOR IMBALANCED DATA

Authors:

Everlandio R. Q. Fernandes (everlandio@usp.br)

Andre C. P. L. de Carvalho (andre@icmc.ups.br)

Xin Yao (X.Yao@cs.bham.ac.uk)

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.

3.1 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 (MARQUÉS; GARCÍA; SÁNCHEZ, 2013). 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 (SUN et al., 2007) 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 (DEEPA; PUNITHAVALLI, 2010). 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 (FERNÁNDEZ et al., 2013) (ZHOU; LIU, 2006). 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 (PRATI; BATISTA; SILVA, 2014) 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 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 (GALAR et al., 2012). 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 (ZHOU, 2009). The generalization ability of an ensemble is usually higher than a single classifier. In (TUMER; GHOSH, 1996) 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 (DIETTERICH, 1997), 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 (KROGH; VEDELSBY, 1995).

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 where the use of diversity measures to select the base classifiers positively affects the ensemble predictive performance (KUNCHEVA; WHITAKER, 2003) (LYSIAK; KURZYNSKI; WOLOSZYNSKI, 2014). Examples of diversity measures include Negative Correlation Learning (NCL) (LIU; YAO, 1997) and Pairwise Failure Crediting (PFC) (CHANDRA; YAO, 2006).

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 trade-off, 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 (LWIN; QU; KENDALL, 2014).

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 ([BACHE; LICHMAN, 2013](#)) 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.

3.2 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 ([CHAWLA et al., 2002](#); [ZHOU; LIU, 2006](#); [SUN et al., 2007](#)) whereas, in the second, algorithms are adapted to increase the importance of instances from the minority class for model optimization ([QUINLAN, 1991](#); [ZADROZNY; ELKAN, 2001](#)). There are also other approaches that focus on feature selection or work at the ensemble level.

3.2.1 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 ([SUN; WONG; KAMEL, 2009](#); [CHAWLA et al., 2002](#)). 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) (CHAWLA et al., 2002) 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) (HU et al., 2009) and Adaptive Synthetic Sampling (ADASYN) (HE et al., 2008). Another aspect to address is that the replication of instances tends to increase the computational cost of the learning process (SUN; WONG; KAMEL, 2009) and can 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 (KUBAT; MATWIN, 1997) 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) (HART, 1968). The elimination of instances from the majority class close to the separation boundary is also handled by the Majority Under-sampling Technique (MUTE) (BUNKHUMPORNPAT; SINAPIROMSARAN; LURSINSAP, 2011), which defines security levels for each instance from the majority class and uses these to propose undersampling.

3.2.2 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 (SCHÖLKOPF et al., 2001) 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 (LIN; TANG; YAO, 2013). 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 (SUN; WONG; KAMEL, 2009), 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.

3.2.3 *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 (ZHOU, 2009). The ability to generalize in an ensemble is generally greater than the isolated classifiers that compose it, usually called base-classifiers. In (TUMER; GHOSH, 1996) 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 (ZHOU, 2009).

In recent years, several ensemble learning methods have been proposed as possible solutions to the task of classification with imbalanced datasets (BHOWAN et al., 2013) (YIN et al., 2014) (WANG et al., 2013) (GONG; HUANG, 2012) (SUN; SONG; ZHU, 2012) (QIAN et al., 2014) (WANG et al., 2013) (KOCYIGIT; SEKER, 2012). 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 (BREIMAN, 1996) and Boosting (FREUND; SCHAPIRE, 1997). 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 (FREUND; SCHAPIRE, 1997) 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 (DIETTERICH, 1997). 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 (KROGH; VEDELSBY, 1995).

Several methods that take into account diversity and accuracy of base classifiers have been proposed. Multiobjective Genetic Sampling (MOGASamp) (FERNANDES; CARVALHO; COELHO, 2015), 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 (BHOWAN et al., 2013), 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 (WANG; YAO, 2012), 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.

3.3 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 (DEB et al., 2002) to evolve a combination of balanced samples, each sample used to induce a base classifier, and evaluate the classifiers

Figure 3 – E-MOSAIC - Ensemble of Classifier based on Multiobjective Genetic Sampling for Imbalanced Classification

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

3.3.1 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.

3.3.2 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 3.1.

$$PPV_{c,i} = \frac{\#true_positives_i}{\#true_positives_i + \#false_positives_i} \quad (3.1)$$

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 (ZITZLER; LAUMANN; THIELE, 2001). 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 (DEB, 2001). 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 (FONSECA; FLEMING, 1993) 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 (BHOWAN et al., 2013) 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. 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.

3.3.3 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 (POLI; LANGDON, 1998). 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.

3.3.4 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.

3.3.5 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.

Despite the individuals selected for the next generation having fitness values (represented by non-dominance rank) better than or equal to the individuals of the current generation, this does not guarantee that the resulting ensemble of these individuals has better accuracy than the ensemble of the current generation. The reason is that, regardless of the accuracy of the models induced by the individuals continue to grow over successive generations, the diversity between models can stagnate or even decrease, harming the 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 (YIN et al., 2014) and mAUC (HAND; TILL, 2001). At first, the initial population and its accuracy measures (G-mean and mAUC) are saved 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.

3.4 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 (BACHE; LICHMAN, 2013). 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.

3.4.1 Compared Methods

A recent study (HULSE; KHOSHGOFTAAR; NAPOLITANO, 2007) 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 pre-processing 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 (LIN; TANG; YAO, 2013) 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) (KUKAR; KONONENKO, 1998) and Rescale_{new} (ZHOU; LIU, 2010) were chosen. Stagewise Additive Modeling using a Multiclass Exponential loss function (SAMME) (ZHU et al., 2009) 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.

3.4.2 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 (HE; GARCIA, 2009). 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 (BRADLEY, 1997) 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) (PROVOST; FAWCETT, 1997), which has been widely used to evaluate the performance of classifiers. Originally, AUC is only applicable to binary-class datasets. However, Hand and Till (HAND; TILL, 2001) 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)* (SUN; KAMEL; 0007, 2006) proposed by Sun, Kamel and Wang (2014) (YIN et al., 2014) 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 (YIN et al., 2014) as

$$G - mean = \left(\prod_{i=1}^m \frac{tr_i}{n_i} \right)^{\frac{1}{m}} \quad (3.2)$$

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 .

3.4.3 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 (BACHE; LICHMAN, 2013). The basic characteristics of the datasets are presented in Table 3, 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 3 (*Letter-1 and Letter-2*).

All the methods in this experimental study use, or were adapted to use, MLP as a base classifier and the *backpropagation* algorithm (RUMELHART; HINTON; WILLIAMS, 1988) was used to train the MLP. The parameters of MLP used here are the same used in (LIN; TANG;

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: 734

Table 3 – Basic Characteristics of The Datasets (#F: The Number of Features, #C: The Number of Classes, #Inst.: The Total Number of Instances)

YAO, 2013) and are shown in Table 4, including the number of hidden nodes (#Hid. Nodes) and the number of training epochs (#Epoch). In addition to the values shown in Table 4, 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 3.3.3. 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 4 – 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 (WANG et al., 2013; YIN et al., 2014).

3.4.4 Experimental Results

3.4.4.1 Comparison with Data Level Methods

Figures 4 and 5 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.

Figure 4 – Mauc Data Level Methods

Figure 5 – G-Mean Data Level 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 (DEM_sAR, 2006). 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 4 and 5), 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 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 4 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 5. 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.

3.4.4.2 Comparison with Algorithm Level Methods

As in the previous subsection, Figures 6 and 7 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 5 shows the number of wins, draws and losses achieved by E-MOSAIC in a pairwise comparison with the algorithm level methods.

Figure 6 – Mauc Algorithm Level Methods

The results of statistical tests, following the methodology proposed by Demšar (DEM_sAR, 2006), 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

Figure 7 – Gmean 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 5 – Number of win-draw-lose between E-MOSAIC and the algorithm-level compared methods.

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 6 and Table 5, 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 7), the proposed method overcomes DyS on 15 datasets and AL on all datasets, with the exception of 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 (KUKAR; KONONENKO, 1998) 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 (KUKAR; KONONENKO, 1998). The results reached by MMC are presented in Figures 6 and 7, 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 (LIN; TANG; YAO, 2013). 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 (ZHOU; LIU, 2010) 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 6 and 7, referring to the MAUC and G-mean metrics respectively. As we can see in Table 5, 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 (FREUND; SCHAPIRE, 1997) has been widely used for solving binary classification problems. In (ZHOU; LIU, 2010), 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 4 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 6 and 7 refer to the results obtained for MAUC and G-mean metrics, respectively.

In Table 5, 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 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 (SHIPP; KUNCHEVA, 2002).

3.4.5 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 6 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 6, 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 6 – Accuracy for each class returned by E-MOSAIC on *Chess*, *Glass*, *Car* and *Conceptive* datasets.

3.5 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. This will be another direction of our future work.

Acknowledgments

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

3.6 Bibliography

BACHE, K.; LICHMAN, M. *UCI Machine Learning Repository*. 2013. Available: <<http://archive.ics.uci.edu/ml>>. Citations on pages 54, 64, 72, 73, and 103.

BHOWAN, U. et al. Evolving Diverse Ensembles using Genetic Programming for Classification with Unbalanced Data. *IEEE Transactions on Evolutionary Computation*, v. 17, n. 3, p. 368–386, 2013. Citations on pages 31, 35, 36, 66, 67, 70, 94, 96, 98, 114, 116, and 117.

BRADLEY, A. P. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, v. 30, p. 1145–1159, 1997. Citations on pages 51 and 73.

BREIMAN, L. Bagging predictors. *Machine Learning*, v. 24, n. 2, p. 123–140, 1996. ISSN 1573-0565. Available: <<http://dx.doi.org/10.1023/A:1018054314350>>. Citations on pages 32, 50, 66, and 95.

BUNKHUMPORNPAT, C.; SINAPIROMSARAN, K.; LURSINSAP, C. MUTE: Majority under-sampling technique. In: *2011 8th International Conference on Information, Communications & Signal Processing*. [S.l.]: IEEE, 2011. p. 1–4. Citations on pages 65 and 93.

CHANDRA, A.; YAO, X. Ensemble learning using multi-objective evolutionary algorithms. *J. Math. Model. Algorithms*, v. 5, n. 4, p. 417–445, 2006. Available: <<http://dx.doi.org/10.1007/s10852-005-9020-3>>. Citations on pages 36, 48, 51, 63, 92, 98, and 117.

CHAWLA, N. V. et al. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, v. 16, p. 321–357, 2002. Citations on pages 29, 64, 65, 93, and 115.

DEB, K. *Multi-Objective Optimization using Evolutionary Algorithms*. [S.l.]: John Wiley & Sons, Chichester, 2001. (Wiley-Interscience Series in Systems and Optimization). Citations on pages 53 and 69.

DEB, K. et al. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Trans. Evol. Comp*, IEEE Press, Piscataway, NJ, USA, v. 6, n. 2, p. 182–197, Apr. 2002. ISSN 1089-778X. Available: <<http://dx.doi.org/10.1109/4235.996017>>. Citations on pages 53, 67, 98, and 117.

DEEPA, T.; PUNITHAVALLI, M. An analysis for mining imbalanced datasets. *International Journal of Computer Science and Information Security*, v. 8, p. 132–137, 2010. Citations on pages 49 and 62.

DEMsAR, J. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.*, JMLR.org, v. 7, p. 1–30, Dec. 2006. ISSN 1532-4435. Available: <<http://dl.acm.org/citation.cfm?id=1248547.1248548>>. Citations on pages 54, 76, 77, and 132.

DIETTERICH, T. G. Machine-learning research – four current directions. *AI MAGAZINE*, v. 18, p. 97–136, 1997. Citations on pages 31, 48, 63, 67, 92, and 95.

FERNANDES, E. R. Q.; CARVALHO, A. C. P. L. F. de; COELHO, A. L. V. An evolutionary sampling approach for classification with imbalanced data. In: *IEEE. Neural Networks (IJCNN), 2015 International Joint Conference on*. [S.l.], 2015. p. 1–7. Citations on pages 67, 96, 98, 116, and 117.

FERNÁNDEZ, A. et al. Analysing the classification of imbalanced datasets with multiple classes: Binarization techniques and ad-hoc approaches. *Knowledge-Based Systems*, v. 42, p. 97 – 110, 2013. ISSN 0950-7051. Available: <<http://www.sciencedirect.com/science/article/pii/S0950705113000300>>. Citations on pages 28, 62, and 92.

FONSECA, C. M.; FLEMING, P. J. *Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization*. 1993. Citation on page 69.

FREUND, Y.; SCHAPIRE, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.*, Academic Press, Inc., Orlando, FL, USA, v. 55, n. 1, p. 119–139, Aug. 1997. ISSN 0022-0000. Available: <<http://dx.doi.org/10.1006/jcss.1997.1504>>. Citations on pages 32, 50, 66, 79, 95, and 104.

GALAR, M. et al. A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, v. 42, n. 4, p. 463–484, 2012. Citations on pages 32, 33, 63, 95, and 114.

GONG, R.; HUANG, S. H. A Kolmogorov–Smirnov statistic based segmentation approach to learning from imbalanced datasets: With application in property refinance prediction. *Expert Systems with Applications*, v. 39, n. 6, p. 6192–6200, May 2012. Citation on page 66.

HAND, D. J.; TILL, R. J. A simple generalisation of the area under the roc curve for multiple class classification problems. *Machine Learning*, Springer Netherlands, v. 45, n. 2, p. 171–186, 2001. ISSN 0885-6125. 10.1023/A:1010920819831. Available: <<http://dx.doi.org/10.1023/A:1010920819831>>. Citations on pages 71 and 73.

HART, P. E. The condensed nearest neighbor rule (corresp.). *IEEE Transactions on Information Theory*, v. 14, n. 3, p. 515–516, 1968. Citations on pages 49, 65, and 93.

HE, H. et al. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In: *IEEE. Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*. [S.l.], 2008. p. 1322–1328. Citations on pages 65 and 94.

HE, H.; GARCIA, E. A. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, IEEE Computer Society, Los Alamitos, CA, USA, v. 21, n. 9, p. 1263–1284, 2009. ISSN 1041-4347. Citations on pages 48, 50, and 73.

HU, S. et al. Msmote: Improving classification performance when training data is imbalanced. In: *Computer Science and Engineering, 2009. WCSE '09. Second International Workshop on*. [S.l.: s.n.], 2009. v. 2, p. 13–17. Citations on pages 65 and 94.

HULSE, J. V.; KHOSHGOFTAAR, T. M.; NAPOLITANO, A. Experimental perspectives on learning from imbalanced data. In: *ACM. Proceedings of the 24th international conference on Machine learning*. [S.l.], 2007. p. 935–942. Citation on page 72.

KOCYIGIT, Y.; SEKER, H. Imbalanced data classifier by using ensemble fuzzy c-means clustering. In: *Proceedings of 2012 IEEE-EMBS International Conference on Biomedical and Health Informatics*. [S.l.]: IEEE, 2012. p. 952–955. Citation on page 66.

KROGH, A.; VEDELSBY, J. Neural network ensembles, cross validation, and active learning. In: *Advances in Neural Information Processing Systems*. [S.l.]: MIT Press, 1995. p. 231–238. Citations on pages 48, 63, and 67.

KUBAT, M.; MATWIN, S. Addressing the curse of imbalanced training sets: One-sided selection. In: *In Proceedings of the Fourteenth International Conference on Machine Learning*. [S.l.]: Morgan Kaufmann, 1997. p. 179–186. Citations on pages 30, 49, 65, 93, and 116.

KUKAR, M.; KONONENKO, I. Cost-sensitive learning with neural networks. In: *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI-98)*. [S.l.]: John Wiley & Sons, 1998. p. 445–449. Citations on pages 72 and 79.

KUNCHEVA, L. I.; WHITAKER, C. J. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, v. 51, n. 2, p. 181–207, May 2003. ISSN 1573-0565. Available: <<https://doi.org/10.1023/A:1022859003006>>. Citation on page 63.

LIN, M.; TANG, K.; YAO, X. Dynamic sampling approach to training neural networks for multiclass imbalance classification. *IEEE Trans. Neural Netw. Learning Syst.*, v. 24, n. 4, p. 647–660, 2013. Available: <<http://dblp.uni-trier.de/db/journals/tnn/tnn24.html#LinTY13>>. Citations on pages 66, 72, 74, 79, and 94.

LIU, Y.; YAO, X. Negatively correlated neural networks can produce best ensembles. *Australian Journal of Intelligent Information Processing Systems*, v. 4, n. 3/4, p. 176–185, 1997. Citations on pages 36, 48, and 63.

LWIN, K.; QU, R.; KENDALL, G. A learning-guided multi-objective evolutionary algorithm for constrained portfolio optimization. *Applied Soft Computing*, v. 24, p. 757 – 772, 2014. ISSN 1568-4946. Available: <<http://www.sciencedirect.com/science/article/pii/S1568494614003913>>. Citation on page 63.

LYSIAK, R.; KURZYNSKI, M.; WOLOSZYNSKI, T. Optimal selection of ensemble classifiers using measures of competence and diversity of base classifiers. *Neurocomputing*, v. 126, p. 29 – 35, 2014. ISSN 0925-2312. Recent trends in Intelligent Data Analysis Online Data Processing. Available: <<http://www.sciencedirect.com/science/article/pii/S092523121300698X>>. Citation on page 63.

MARQUÉS, A. I.; GARCÍA, V.; SÁNCHEZ, J. S. On the suitability of resampling techniques for the class imbalance problem in credit scoring. *JORS*, v. 64, n. 7, p. 1060–1070, 2013. Available: <<http://dx.doi.org/10.1057/jors.2012.120>>. Citations on pages 48 and 62.

POLI, R.; LANGDON, W. B. Genetic programming with one-point crossover. In: _____. *Soft Computing in Engineering Design and Manufacturing*. London: Springer London, 1998. p. 180–189. ISBN 978-1-4471-0427-8. Available: <http://dx.doi.org/10.1007/978-1-4471-0427-8_20>. Citations on pages 70 and 101.

PRATI, R. C.; BATISTA, G. E.; SILVA, D. F. Class imbalance revisited: a new experimental setup to assess the performance of treatment methods. *Knowledge and Information Systems*,

Springer London, p. 1–24, 2014. Available: <<http://dx.doi.org/10.1007/s10115-014-0794-3>>. Citations on pages 62, 90, and 100.

PROVOST, F. J.; FAWCETT, T. Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In: HECKERMAN, D.; MANNILA, H.; PREGIBON, D. (Ed.). *KDD*. [S.l.: s.n.], 1997. p. 43–48. Citations on pages 51 and 73.

QIAN, Y. et al. A resampling ensemble algorithm for classification of imbalance problems. *Neurocomputing*, Elsevier, v. 143, p. 57–67, Nov. 2014. Citations on pages 31, 66, 94, and 114.

QUINLAN, J. R. Improved estimates for the accuracy of small disjuncts. *Machine Learning*, Springer, v. 6, n. 1, p. 93–98, 1991. Citations on pages 64 and 115.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Neurocomputing: Foundations of research. In: ANDERSON, J. A.; ROSENFELD, E. (Ed.). Cambridge, MA, USA: MIT Press, 1988. chap. Learning Representations by Back-propagating Errors, p. 696–699. ISBN 0-262-01097-6. Available: <<http://dl.acm.org/citation.cfm?id=65669.104451>>. Citation on page 73.

SCHÖLKOPF, B. et al. Estimating the support of a high-dimensional distribution. *Neural Computation*, v. 13, n. 7, p. 1443–1471, 2001. Citations on pages 30, 65, and 94.

SHIPP, C. A.; KUNCHEVA, L. I. Relationships between combination methods and measures of diversity in combining classifiers. *Information Fusion*, v. 3, n. 2, p. 135 – 148, 2002. ISSN 1566-2535. Available: <<http://www.sciencedirect.com/science/article/pii/S1566253502000519>>. Citation on page 80.

SUN, Y.; KAMEL, M. S.; 0007, Y. W. Boosting for learning multiple classes with imbalanced class distribution. In: *ICDM*. IEEE Computer Society, 2006. p. 592–602. ISBN 0-7695-2701-9. Available: <<http://dblp.uni-trier.de/db/conf/icdm/icdm2006.htmlSunKW06>>. Citation on page 73.

SUN, Y. et al. Cost-sensitive boosting for classification of imbalanced data. *Pattern Recogn.*, Elsevier Science Inc., New York, NY, USA, v. 40, n. 12, p. 3358–3378, Dec. 2007. ISSN 0031-3203. Available: <<http://dx.doi.org/10.1016/j.patcog.2007.04.009>>. Citations on pages 27, 33, 62, 64, and 115.

SUN, Y.; WONG, A. K. C.; KAMEL, M. S. Classification of imbalanced data: a review. *IJPRAI*, v. 23, n. 4, p. 687–719, 2009. Available: <<http://dx.doi.org/10.1142/S0218001409007326>>. Citations on pages 64, 65, 66, 94, and 115.

SUN, Z.; SONG, Q.; ZHU, X. Using Coding-Based Ensemble Learning to Improve Software Defect Prediction. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, v. 42, n. 6, p. 1806–1817, Nov. 2012. Citations on pages 66 and 94.

TUMER, K.; GHOSH, J. Analysis of decision boundaries in linearly combined neural classifiers. *Pattern Recognition*, v. 29, p. 341–348, 1996. Citations on pages 48, 63, 66, 91, and 115.

WANG, J. et al. Ensemble of Cost-Sensitive Hypernetworks for Class-Imbalance Learning. In: *2013 IEEE International Conference on Systems, Man, and Cybernetics*. [S.l.]: IEEE, 2013. p. 1883–1888. Citations on pages 31, 66, 75, and 114.

WANG, S.; YAO, X. Multiclass imbalance problems: Analysis and potential solutions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, v. 42, n. 4, p. 1119–1130, Aug 2012. ISSN 1083-4419. Citations on pages 29, 67, 93, and 116.

YIN, Q.-Y. et al. A Novel Selective Ensemble Algorithm for Imbalanced Data Classification Based on Exploratory Undersampling. *Mathematical Problems in Engineering*, Hindawi Publishing Corporation, v. 2014, p. 1–14, 2014. Citations on pages 31, 33, 66, 71, 73, 75, 94, 100, 114, and 117.

ZADROZNY, B.; ELKAN, C. Learning and making decisions when costs and probabilities are both unknown. In: ACM. *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.], 2001. p. 204–213. Citations on pages 64 and 115.

ZHOU, Z.-H. Ensemble learning. In: LI, S. Z.; JAIN, A. K. (Ed.). *Encyclopedia of Biometrics*. Springer US, 2009. p. 270–273. ISBN 978-0-387-73003-5. Available: <<http://dblp.uni-trier.de/db/reference/bio/e.htmlZhou09>>. Citations on pages 48, 63, 66, and 115.

ZHOU, Z.-H.; LIU, X.-Y. Training cost-sensitive neural networks with methods addressing the class imbalance problem. *Knowledge and Data Engineering, IEEE Transactions on*, IEEE, v. 18, n. 1, p. 63–77, Jan. 2006. ISSN 1041-4347. Available: <<http://dx.doi.org/10.1109/tkde.2006.17>>. Citations on pages 62, 64, and 115.

ZHOU, Z.-H.; LIU, X.-Y. On multi-class cost-sensitive learning. *Computational Intelligence*, v. 26, n. 3, p. 232–257, 2010. Available: <<http://dblp.uni-trier.de/db/journals/ci/ci26.htmlZhouL10>>. Citations on pages 72 and 79.

ZHU, J. et al. Multi-class adaboost. *Statistics and its Interface*, v. 2, n. 3, p. 349–360, 2009. Citation on page 72.

ZITZLER, E.; LAUMANN, M.; THIELE, L. SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. In: GIANNAKOGLU, K. C. et al. (Ed.). *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*. Athens, Greece: International Center for Numerical Methods in Engineering, 2001. p. 95–100. Citation on page 69.

EVOLUTIONARY INVERSION OF CLASS DISTRIBUTION IN OVERLAPPING AREAS FOR MULTI-CLASS IMBALANCED LEARNING

Authors:

Everlandio R. Q. Fernandes (everlandio@usp.br)

Andre C. P. L. de Carvalho (andre@icmc.ups.br)

Abstract

Inductive learning from multi-class and unbalanced datasets is one of the main machine learning challenges. Most machine learning algorithms have their predictive performance negatively affected by imbalanced data. Although several techniques have been proposed to deal with this difficulty, they are usually restricted to binary classification datasets. Thus, one of the research challenges in this area is how to deal with imbalanced multiclass classification datasets. This challenge increases when classes containing fewer instances are located in overlapping regions of the data attribute space. In fact, several studies have indicated that the degree of class overlapping has a higher effect on the predictive performance than the global class imbalance ratio. This paper proposes a novel evolutionary ensemble-based method for multi-class imbalanced learning — the evolutionary inversion of class distribution for imbalanced learning (EVINCI). EVINCI uses a multiobjective evolutionary algorithm (MOEA) to evolve a set of samples taken from an unbalanced dataset. It selectively reduces the concentration of less representative instances of the majority classes in the overlapping areas while selecting samples that produce more accurate models. In experiments performed to evaluate its predictive accuracy, EVINCI was superior to state-of-the-art ensemble-based methods for imbalanced learning.

4.1 Introduction

Today, many data classification tasks involve unbalanced datasets, in which at least one class is underrepresented. This situation is found in many real-world problems, such as data analysis of fraudulent credit card transactions, disease diagnosis, image analysis of defective parts on production lines, and ecology, to name a few. In consequence, the development and investigation of techniques that can perform effective data classification in imbalanced datasets is currently one of the most compelling research issues in data mining and machine learning (LÓPEZ et al., 2013).

Most classical classification algorithms have difficulties in dealing with imbalanced datasets. These difficulties occur because, to improve the overall predictive accuracy of a training data subset, they usually induce classification models that tend to give lesser consideration to classes with few examples (minority classes). This situation becomes even more challenging when objects from minority classes are situated in overlapping regions of the data attribute space. In fact, several studies (GARCÍA; MOLLINEDA; SÁNCHEZ, 2008; LÓPEZ et al., 2013; PRATI; BATISTA; SILVA, 2014) have indicated that class distribution is not primarily responsible for hindering classifier performance, but rather it is the degree of overlap between the dataset classes.

In particular, (GARCÍA; MOLLINEDA; SÁNCHEZ, 2008) presented an interesting study of this subject. The authors proposed two different frameworks that focus on the performance of the K -NN classification algorithm. In the first framework, they investigate the situation in which the imbalance ratio in the overlap region is similar to the overall imbalance ratio. In the second, the imbalance ratio in the overlapping areas is inversely related to the overall imbalance ratio, that is, the minority class is locally denser in the overlapping regions. Their experimental results seemed to indicate that the behavior of the K -NN is more dependent on changes in the imbalance ratio in the overlapping region than on changes in the size of the overlapping area. Their results also indicated that the proportion of the local imbalance ratio and the size of the overlapping region are more important than the relationship with the global imbalance ratio. In (LÓPEZ et al., 2013), the authors indicate that, in the case of an overlapping region, most classification algorithms are not only unable to correctly discriminate between classes, but they also favor the majority classes, which leads to low overall classification accuracy.

Although many studies highlight the problem of the dominance of a class in the overlapping region, its treatment has received little attention in the imbalanced learning literature, as discussed in a recent work about data irregularities in classification tasks (DAS; DATTA; CHAUDHURI, 2018).

Various measures for estimating the complexity of the separation frontier and its overlapping rates have been investigated over the years. In particular, (HO; BASU, 2002) proposed and evaluated several measures that characterize the difficulty of a classification problem, focusing

on the geometric complexity of the frontier of classes' separation. One of these measures is based on the test proposed by (FRIEDMAN; RAFSKY, 1979), which verifies whether two samples, for example, two datasets collected at different times, come from the same data distribution. Ho and Basu used this test, which they named the N1 measure, to decide whether instances from different classes form separable distributions.

The N1 measure initially generates a minimum spanning tree (MST) that connects all instances of a dataset, taking into account the Euclidean distance and ignoring the class associated with each instance. Next, N1 counts the number of instances connected by an edge in the MST and belonging to different classes. These instances are considered to be close to the class boundary. As this measure was originally designed for binary datasets, it uses the ratio of the sum of these points to the total points in the data set as the measure of complexity that estimates the separability of classes, that is, the fraction of instances in the dataset that lie at the class boundary. High N1 values indicate the need for a more complex separation boundary for the dataset.

When classifying multi-class unbalanced datasets, it would be interesting to verify the value of this ratio for all classes and, in particular, between the majority and minority classes. By doing so, this measure could be used to optimize the sampling of the dataset so that the overlapping areas present a higher concentration of examples from the minority classes, thereby increasing their visibility to the classification algorithm. However, even though the overlapping regions have a higher concentration of minority classes, the resulting classifier may have low overall accuracy since relevant instances of the majority classes could be eliminated, thus impeding their recognition. That is, increasing the accuracy of some classes can lower the accuracy of others.

One possible solution to the situation described above is to combine ensembles of classifiers and multiobjective evolutionary algorithms (MOEAs). In contrast to traditional machine learning methodologies that construct a single hypothesis (model) of the training dataset, ensemble learning techniques induce a set of hypotheses and combine them through some consensus method or operator. An essential characteristic of ensembles of classifiers is their generalization power, which is greater than that of the classifiers composing the ensemble (base classifiers), as formally presented in (TUMER; GHOSH, 1996). The MOEAs can adequately manage conflicting objectives in the learning process by simultaneously evolving a set of solutions into two or more objectives without needing to impose preferences on the objectives.

Using this context, in this paper, we propose a new ensemble-based method for multi-class imbalanced learning, which we call the evolutionary inversion of class distribution for imbalanced learning (EVINCI). Using a MOEA, EVINCI optimizes a set of samples taken from the training dataset so that they present a higher concentration of instances of the minority classes in the overlapping areas, thereby making these classes more evident. We developed an extension of the N1 complexity measure, called N1byClass, for EVINCI to estimate the overlap

percentage of each pair of classes. With the support of the N1byClass measure and the improved accuracy of the model induced by the samples, EVINCI selectively reduces the concentration of less representative instances of the majority classes in the overlapping areas while also selecting the samples that produce more accurate models.

To increase its generalization power and reduce the information losses resulting from the selection process, the EVINCI classification system consists of an ensemble of classifiers, in which each base classifier is induced by a different optimized sample. Moreover, the processing of the proposed method incorporates two rules to promote the diversity of the classifiers generated by its evolutionary process, based on the necessary condition for building an effective ensemble of classifiers (DIETTERICH, 1997). First, it eliminates samples generated with a high degree of similarity. Then, the measure of classifier diversity, pairwise failure crediting (PFC) (CHANDRA; YAO, 2006), resolves any tie issues in the selection process.

The remainder of this paper is organized as follows. In section II, we discuss the problem of classifying a multi-class unbalanced dataset and the classical approaches that have been proposed in the literature to address this problem. In section III, we present the use of an ensemble of classifiers to solve the classification problem in unbalanced datasets. In section IV, we propose our solution whereby we change the complexity measure N1 for use in the multi-class datasets domain. In the next section (V), we present a more detailed description of the proposed method. We performed experiments on 22 datasets with different imbalance ratios and numbers of classes ranging from 2 to 18. We describe the experiments and discuss the results we obtained in section VI. Finally, in section VII, we present our conclusions.

4.2 Unbalanced Datasets Methods and Issues

The generation of classification models from unbalanced datasets has been extensively addressed in the machine learning literature over at least the last 20 years. However, most of the proposed techniques were designed and tested for binary dataset scenarios, i.e., datasets with two classes. In this case, the researchers focus on the correct classification of the class with fewer examples (minority class), since the classifier usually tends to choose the majority class by default. Unfortunately, when a multi-class dataset is presented, the solutions proposed in the literature are not directly applicable or achieve below-expected performance (FERNÁNDEZ et al., 2013).

Class decomposition is a commonly applied solution to multi-class problems, whereby a multi-class problem is transformed into a set of sub-problems, each with two classes. The most common application of this technique is as follows: given a dataset with more than two classes, one class is chosen as the positive class, and all the other classes are labeled as contrary to the positive, i.e., negative. This new dataset labeling is used to induce a binary classifier. The process is repeated so that in each round a different class is chosen as positive (RIFKIN; KLAUTAU,

2004). For example, for a dataset with five classes, five binary classifiers will be generated. This technique is known as one-against-all or one-vs-others. However, in their study, (WANG; YAO, 2012) examined many issues related to the classification of multi-class unbalanced datasets. Regarding the decomposition class, their study results indicated that this methodology provided no advantages for multi-class imbalance learning, and even made the generated sub-problems more unbalanced.

When considering unbalanced binary datasets, the solutions proposed in the literature can be divided into two groups: the data level and algorithm level. The first group, which is most popular in the literature, preprocesses the dataset to be presented to the classification algorithm. The goal of these methods is to rebalance the classes by resampling the data space. This is mainly achieved by undersampling the majority classes, oversampling the minority classes, or some combination of both. Thus, the expected result is the containment of any system bias for the majority class that is due to the different class distribution. The main advantage of these techniques is that they are independent of the classification algorithms used in the next phase.

Typically, undersampling only eliminates examples of the majority classes. When undersampling is performed randomly (random undersampling (RUS)), there may be a loss of relevant information from the classes that have been reduced. Directed or informative undersampling attempts to work around this problem by detecting and eliminating a less significant fraction of the data. This is the strategy used in the one-sided selection (OSS) technique (KUBAT; MATWIN, 1997), which attempts to remove redundant and/or noisy instances from the majority class lying close to the boundary. Border instances are detected by applying Tomek links and instances far from the decision boundary (redundant instances) are discovered using the condensed nearest neighbor (CNN) rule (HART, 1968). The elimination of examples from the majority class lying close to the separation boundary is also performed by the majority undersampling technique (MUTE) (BUNKHUMPORNPAT; SINAPIROMSARAN; LURSINSAP, 2011), which defines security levels for each instance from the majority class and uses these to determine the need for undersampling.

In oversampling methods, elements of the minority class is replicated or generated synthetically until the size of the minority class is close or equal to that of the other class. In random oversampling (ROS), minority class instances are randomly selected and replicated. The synthetic minority over-sampling technique (SMOTE) (CHAWLA et al., 2002), another example of an algorithm that uses oversampling, uses data interpolation to synthetically generate new instances of the minority classes. To generate a new example, SMOTE considers the feature space and then selects an example from the minority class and finds its k -nearest neighbors that share the same class. Then, it synthetically generates new instances along line segments that join any/all of its k -nearest neighbors.

Depending on how the examples are generated, oversampling techniques can increase the overlap between classes. Some methods have been proposed to minimize this drawback,

including the modified synthetic minority oversampling technique (MSMOTE) (HU et al., 2009) and adaptive synthetic sampling (ADASYN) (HE et al., 2008). In (SÁEZ et al., 2015), the authors propose another extension to the SMOTE method, the SMOTE-IPF. In this method, the ensemble-based filter IPF (KHOSHGOFTAAR; REBOURS, 2007) is applied after the generation of the new cases by SMOTE, so that, through an iterative process, select and remove examples considered as noise. Another aspect to be addressed is the tendency of instance replication to increase the computational cost of the learning process (SUN; WONG; KAMEL, 2009) and generation data that may not make sense in the investigated problem.

The second group of solutions proposed to address unbalanced binary datasets are those in the algorithm group. These solutions are based on the adaptation of some existing classification algorithm to alleviate the bias of these algorithms toward the majority class. There are three main categories in this group — those based on recognition, cost-sensitive methods, and ensemble-based methods.

Recognition-based methods take the form of one-class learners. This is an extreme case in which only examples from one class are used to induce the classification model, usually instances from the minority class. The one-class SVM method (SCHÖLKOPF et al., 2001) is recognition-based in that it considers only the minority class during the learning process to determine the class of interest. It infers the properties of minority class cases and from those can predict which examples differ from the class of interest. However, as indicated by the authors in (ALI; SHAMSUDDIN; RALESCU, 2015), some classification algorithms such as decision trees, naive-Bayes, and others do not work with examples from only one class, which makes these methods unpopular and restricts them to certain learning algorithms.

Many existing classification algorithms are designed to assign equal costs to the errors made in different classes, and the modification of this criterion is the central proposal of cost-sensitive methods. These approaches include assigning different costs to incorrect predictions or developing training criteria that are more sensitive to a skewed distribution. The latter is the case in dynamic sampling (DyS) (LIN; TANG; YAO, 2013), which proposes to train a multilayer perceptron (MLP). At each epoch of the MLP training process, DyS uses a decision heuristic based on the current status of the MLP to decide which examples will be used to update the MLP weights. As noted by the authors in (SUN; WONG; KAMEL, 2009), solutions at the algorithm level are usually specific to a particular algorithm and/or problem. Consequently, they are only useful in certain contexts and generally require experience in the field of application and in the classification algorithms used.

In recent years, there has been growing use of ensembles of classifiers as a possible solution for imbalanced learning (SUN; SONG; ZHU, 2012; BHOWAN et al., 2013; YIN et al., 2014; QIAN et al., 2014). The proposed solutions are based on a combination of ensemble learning techniques and some resampling method or cost-sensitive method, or an adaptation of some existing classification algorithm. We discuss the use of ensembles of classifiers for

unbalanced datasets in the next section.

4.3 Imbalanced Ensemble Learning

Ensemble methods leverage the classification power of base classifiers by combining them to form a new classifier that outperforms each of them. (DIETTERICH, 1997) discusses and provides an overview of why ensemble methods usually outperform single classifier methods. In the work conducted by (HANSEN; SALAMON, 1990), the authors proved that under specific constraints the expected error rate of an instance goes to zero as the number of base classifiers goes to infinity. To do so, the base classifiers must have an accuracy rate higher than 50% and be as diverse as possible. Two classifiers are considered to be diverse if their misclassifications are made at different instances in the same test set. Considering the techniques used to construct an ensemble of classifiers, the algorithms most often used are those based on bagging (BREIMAN, 1996) and boosting (FREUND; SCHAPIRE, 1997) methods.

In the bagging method, different samples bootstrapped from the training dataset induce the set of base classifiers. Sampling is performed by replacement, and each sample has the same size and class distribution as the original training dataset. When an unknown case appears in the system, each base classifier makes a prediction, and the class with the most votes is assigned to the new instance.

Bagging-based methods that have been proposed to deal with unbalanced datasets differ mainly in the way they collect the samples that induce the base classifiers. These methods construct balanced samples from the training dataset. That is, the problem of unbalanced datasets is addressed by pre-processing the samples before inducing each classifier. As such, different sampling strategies lead to different bagging-based methods. Random oversampling bagging (OverBagging) and SMOTEBagging (WANG; YAO, 2009) methods perform oversampling upon each iteration of the bagging method. The OverBagging method conducts a random oversampling of the minority classes and SMOTEBagging synthetically generates new instances using the SMOTE algorithm. The UnderBagging (BARANDELA; VALDOVINOS; SÁNCHEZ, 2003) method performs a random undersampling of the majority classes before inducing the base classifiers.

The AdaBoost method (FREUND; SCHAPIRE, 1997) is the most typical algorithm in the boosting family. The complete training dataset is used to sequentially generate the base classifiers. During each iteration, examples incorrectly classified in the previous iteration are emphasized when a new classifier is being induced. In the AdaBoost method, the base classifiers produce weighted votes based on their overall accuracy. So, when a new instance is presented, each base classifier gives its vote, and the class that receives more votes is assigned to the new instance.

As mentioned in (GALAR et al., 2012), boosting algorithms are usually fused with

cost-sensitive learning or resampling techniques, which then respectively generate cost-sensitive boosting or boosting-based ensembles. The AdaBoost algorithm is oriented toward overall accuracy and, as noted above, when the dataset is unbalanced this kind of algorithm tends to promote new instances as being in the majority class. For this reason, cost-sensitive boosting algorithms propose to change how the weights of instances are updated, prioritizing the minority class. This is the case in the AdaCost (FAN et al., 1999) method, which increases the weights of misclassified examples of minority class more aggressively but decreases the weights of correctly classified samples more conservatively. As in bagging-based methods, boosting-based methods use some re-sampling technique to deal with unbalanced datasets. This is the case in RUSBoost (SEIFFERT et al., 2010), which applies random undersampling in the original training dataset at each iteration and adjusts the distribution of the weights of instances according to the new size of the dataset.

Several methods were proposed aiming to improve the diversity and accuracy of the base classifiers simultaneously. These methods typically use some evolutionary algorithms as mechanisms for evolving a group of solutions or classifiers. This is the case in multiobjective genetic sampling (MOGASamp) (FERNANDES; CARVALHO; COELHO, 2015), which builds an ensemble of classifiers by manipulating samples taken from the training dataset. In this method, each sample represents an individual in the MOEA proposed by the authors, which is being evaluated based on the classification model that it induces. The authors defined the objectives of the MOEA as the selection of samples that induce classifiers with higher accuracy and that present more significant divergence from other classifiers. In (BHOWAN et al., 2013), the authors proposed a multiobjective genetic programming (MOGP) method that uses the accuracies of the minority and majority classes as opposing objectives in the learning process. The MOGP approach was adapted to evolve diverse solutions into an ensemble, thereby improving the general classification performance. Both the MOGASamp and MOGP methods were proposed to deal with binary datasets.

4.4 N1byClass

As discussed earlier, Ho and Basu proposed the N1 measure of complexity, which estimates the separability of classes in binary datasets. N1 is based on the percentage of points (instances) of the dataset that are connected in a minimum spanning tree (MST) and belong to different classes. The MST has direct application in a range of fields, including network design, image processing, and clustering algorithms (GRAHAM; HELL, 1985).

Given a dataset, a spanning tree for this dataset is a graph that contains all the instances of the dataset that are connected by a weighted and non-directed edge and that have no cycle. Thus, for a given dataset, it is possible to construct several spanning trees with different associated costs (sum of weighted edges). An MST is a spanning tree whose cost is the lowest of all the

possible spanning trees (GRAHAM; HELL, 1985). Considering the weights associated with each edge as a Euclidean distance, then two points connected in an MST and belonging to different classes either lie at the boundary of separation of the two classes or one of these points represents noise.

Based on this concept, our proposal of the N1byClass is to generate an MST for the dataset to connect the points by their Euclidean distance. After this step, each class of the dataset is checked for the percentage of its instances that are connected to the other classes and to itself, which generates a matrix of values. The value corresponding to N1byClass for the class i , considering the class j , is given by Equation 4.1, as follows:

$$N1byClass_{\{i,j\}} = \frac{1}{n_i} \sum a_{i,j} \quad (4.1)$$

where n_i is the number of elements of class i and a_{ij} represents the existence of a connection between an instance of class i and an instance of class j . It must be observed that when there is a connection between instances of the same class, this connection is counted twice. However, this situation is disregarded by EVINCI, since it looks for information on areas of overlap between the minority classes and the majority classes. Section 4.5.1 explains how EVINCI makes use of the N1byClass matrix.

Figure 8 shows an MST for a dataset with three classes and 15 instances. For the MST shown in Figure 8, Table 7 shows the resulting N1byClass.

Figure 8 – Minimum Spanning Tree

Class	Square	Heptagon	Circle
Square	0.4	1.0	0.2
Heptagon	1.0	1.2	0.4
Circle	0.2	0.4	0.8

Table 7 – N1byClass based on the MST shown in Figure 8

4.5 Proposed Method

The primary objective of EVINCI is to build an ensemble of classifiers with high accuracy and generalization power for multi-class imbalance classification. These base classifiers are induced by optimized and diverse samples from unbalanced datasets. To do so, the proposed method uses a MOEA to evolve a combination of samples, guided by the class distribution in regions of overlap between majority and minority classes and by the accuracy of the classifiers induced by the samples. EVINCI resolves tie situations in the selection process by the PFC measure of classifier diversity. PFC has been shown to produce good estimates of the diversity of classifiers when applied to imbalanced learning problems (CHANDRA; YAO, 2006; BHOWAN et al., 2013; FERNANDES; CARVALHO; COELHO, 2015). The use of PFC and a mechanism to eliminate similar solutions after the crossover process promotes the creation of diverse solutions in the evolutionary process. Figure 9 shows the workflow of the proposed method.

Figure 9 – EVINCI's Workflow

EVINCI uses a customized well-known MOEA, based on NSGA II (DEB et al., 2002), that uses the non-dominance rank of its objectives to select the most suitable solutions. Non-dominance rank is a common Pareto-based dominance measure that calculates the number of other solutions in a population that dominate a given solution. In Pareto dominance, a solution x_1 dominates another solution x_2 if it is no worse than x_2 in any objective, and x_1 is undoubtedly better than x_2 in at least one of them. This technique allows individuals to be ranked according to

their performance on all the objectives for all individuals in the population. So, a non-dominated solution will have the best fitness of 0, whereas high fitness values indicate poor-performing solutions, i.e., solutions dominated by many other solutions.

To systematically decide which classes are majority and minority in a multi-class dataset, EVINCI uses a limit based on the class distribution of the dataset. Thus, all classes having fewer elements than this value are considered to be minority. Equation 4.2 shows how this limit is calculated:

$$lim = Abs \left(mean(Sc_1, Sc_2, \dots, Sc_n) - \frac{SD(Sc_1, Sc_2, \dots, Sc_n)}{2} \right) \quad (4.2)$$

where Sc_n represents the number of instances of the class n in the training dataset, *mean* is the arithmetic mean, *SD* is the standard deviation, and *Abs* indicates that the absolute value of the result must be extracted.

Equation 4.2, which we developed during the experimental tests of our work, fits very well with the datasets used in our experiments, but needs more validation with other datasets. We emphasize that we used Equation 4.2 as a systematic form of decision-making, but the proposed method can accept the manual insertion of the definition of classes by a dataset specialist. Table 8 shows in bold the dataset classes used in the experiments that were identified as minority classes.

Next, we highlight the main contributions of EVINCI and how EVINCE differs from the other methods also proposed to deal with imbalanced learning. First, as far as the authors know, EVINCE is the first method based on the concept that models induced from samples from the original unbalanced dataset with a higher concentration of instances from the minority classes in the overlap regions can build ensembles of classifiers with higher generalization ability and accuracy. EVINCE disregards the overall imbalance rate of the dataset, focusing on the overlap areas. For such, EVINCI uses a complexity measure specifically designed to extract information regarding overlap regions in datasets, the N1byClass. This is another important contribution from this paper. It must be observed that during the evolutionary process there is no restriction to the imbalance growth of the samples.

EVINCI employs NSGA-II for sampling optimization. However, the process of the evolutionary algorithm was changed in several points to fit the desired search. First, a process was introduced to eliminate similar solutions and the crowding distance was replaced by a measure of diversity of classifiers, since it was seen to be better for the desired search, as explained in Section 4.5.3. In addition, EVINCI employs a strategy to maintain the best ensemble found during sample evolution. This strategy differs from most elitism strategies used in evolutionary algorithms. In the proposed strategy, the fittest individuals go from one generation to another, but the new version of elitism proposed for EVINCI maintains the best result of the interaction of the individuals throughout the generations, that is, the population that produced the most effective ensemble of classifiers. Section 4.5.3 explains how the process that results in the "Saved Ensemble" occurs.

4.5.1 Initial Population and Fitness

To generate a diverse initial population, random sampling is performed with different imbalance ratios up to the 1:3 limit. According to the authors of the study performed in (PRATI; BATISTA; SILVA, 2014), this level of imbalance causes no significant loss of accuracy in most classifiers. The individuals (samples) are represented by binary vectors with the same size as the training dataset, for which each cell represents an instance of the training dataset. The vector value 1 indicates that the instance corresponding to that position in the training dataset is present in the sample and the value 0 represents its absence. Figure 10a shows two examples of vector representation of individuals for a dataset with 15 instances and three classes.

After generating the initial population and identifying which classes are majority and which are minority (either manually or by Equation 4.2), an N1byClass matrix is generated for each individual. The first objective of the adapted MOEA is computed as the sum of the values corresponding to the percentage of instances of the majority classes that are at the border of separation with the minority classes. This means that in the N1byClass matrix, the cells are identified by rows relating to majority classes and columns relating to minority classes. The genetic algorithm looks for solutions that have lower values of this objective, that is, samples in which the majority classes present fewer connections in the MST with minority classes.

The second objective of the adapted MOEA is the accuracy of the classification model generated by the sample. Each individual induces a classification model and its accuracy is calculated by verifying its effectiveness in the complete training dataset. We used the G-mean (YIN et al., 2014) as the accuracy measure in our study. Since this measure determines the geometric mean of the correctness of a given classifier, low accuracy in at least one class leads to a low G-mean value, which makes it less useful in practice. Equation 4.3 shows how the G-mean is calculated:

$$G - mean = \left(\prod_{i=1}^m \frac{tr_i}{n_i} \right)^{\frac{1}{m}} \quad (4.3)$$

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 .

4.5.2 Selection and Reproduction

The values for the above two objectives are used to generate the non-dominance rank, which represents the fitness level of each individual in the population and is used to select individuals who will participate in the breeding process. This selection is performed based on a tournament between three individuals and tie situations are decided randomly. The tournament process returns a number of pairs of individuals equal to the size of the population.

For each pair of parents, two new individuals are generated using one-point crossover (POLI; LANGDON, 1998) per class. It selects a single crossover point on both sub-vectors representing the instances of a given class, and all data beyond that point are exchanged between the two parents. After applying the process to all classes in the dataset, the sub-vectors are consolidated to form the resulting offspring. Figure 10 presents two individuals representing samples of a dataset with 15 instances and 3 classes (Figure 10a) and the crossover process for the class represented by black squares (Figure 10b). Mutation occurs in a percentage of the generated offspring by inverting their bits in a random part of the vector representing an individual.

(a) Vectors representing random samples of a dataset with 15 instances and 3 classes.

(b) One-point crossover per class

Figure 10 – Crossover process

After the reproduction process, the proposed method applies a mechanism to eliminate samples with a high level of similarity. During our experimental tests, we observed that similar individuals with high fitness are more likely to be selected for the breeding process to produce future generations. This increases the number of similar samples and reduces the expected diversity of the solutions. This is why the elimination mechanism is invoked to discard individuals with similarity levels greater than 85%. After this process, if the number of individuals is less than the initial population size, new reproduction and mutation processes are performed.

4.5.3 New Generation, Saved Ensemble and Stop Criteria

All individuals in the offspring population are evaluated according to the defined objectives and put with the current population to form an intermediate population. Then, the non-dominance rank is rebuilt for this intermediate population and, based on the updated rank, individuals are selected to comprise the new generation. First, individuals with better levels of non-dominance are selected (i.e. non-dominance rank equal to 0), then only those who are not

dominated by the first individuals, and so on until the default population size is reached. Tie situations are resolved using the measure of classifier diversity PFC.

As noted earlier, the proposed method uses the PFC measure of classifier diversity, due to its good results in imbalanced classification and because it shows more compliance with the performed search than does the crowding distance measure 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, 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 measured for each individual based on its pair-wise comparison with all individuals in the intermediary population.

The composition of the ensemble is meant to relieve the loss of information inherent in the sampling process. Thus, different classifiers should have different views of the dataset. This is supported by the diversity mechanisms we incorporated into the proposed method. Despite the individuals selected for the next generation having fitness values better than or equal to the individuals of the current generation, this does not guarantee that the resulting ensemble of these individuals will have better accuracy than the ensemble of the current generation. 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 the G-mean accuracy measure is extracted from this evaluation. First, the models of the initial population and its G-mean are saved as *"Saved Ensemble"*. After each generation, the G-mean of the current ensemble population is compared with that of the *"Saved Ensemble"*. If the current ensemble of classifiers shows improvement in the G-mean, the models of current population replaces the *"Saved Ensemble"*.

This process stops after a fixed number of generations, after 10 generations without any replacement of the *"Saved Ensemble"*, or when the G-mean of the ensemble reaches its maximum value, i.e., $\text{max. G-mean} = 1.0$. The classification models of all individuals in the final *"Saved Ensemble"* comprise the ensemble of classifiers returned by EVINCI. When a new example is presented to the system, its class is determined by the majority vote based on the output of all the classifiers.

4.6 Experiments

In this section, we present an empirical analysis of EVINCI, including a comparison of its performance with those of other ensemble-based methods that have been proposed for imbalanced learning. In addition, we analyze the EVINCI results when using only one objective function. That is, we determine how the proposed method behaves when its only objective is to

decrease the density of the majority classes in the overlapping areas and how it behaves when using only model accuracy as a selection factor. Our goal in these experiments was to verify whether the proposed method actually offers some advantage in terms of overall performance and to examine its influence on the learning process. The comparisons also allowed us to determine the individual strengths and weaknesses of the proposed method compared to other state-of-art approaches.

4.6.1 Experimental Setup

In the experiments, we included twenty-two datasets with different imbalance ratios and with the number of classes ranging from 2 to 18. We obtained these datasets from the UCI (BACHE; LICHMAN, 2013) and Keel (ALCALÁ-FDEZ et al., 2011) repositories, except for the *Dnormal* dataset, which is an artificial dataset. All these datasets are summarized in Table 8, including the number of classes (#C), number of features (#F), multi-class imbalance ratio (Imb Ratio) (TANWANI; FAROOQ, 2010), and class distribution.

Data Set	#Class	#F	Imb Ratio	Class Distribution
Abalone	18	8	1.0572	15 : 57 : 115 : 259: 391: 568: 689: 634: 487: 267: 203: 126: 103 : 67 : 58 : 42 : 32 : 26
Balance-scale	3	4	1.0457	49 : 288: 288
Car	4	6	1.1029	384: 69 : 1210: 65
Chess	18	6	1.0604	2796: 1433: 2854: 2166: 471 : 198 : 4553: 1712: 78 : 683 : 592 : 390 : 1985: 4194: 81 : 3597: 246 : 27
Contraceptive	3	9	1.1499	629: 333 : 511
Dermatology	6	34	1.0334	112: 61: 72: 49: 52: 20
Dnormal	3	2	1.1962	324: 342: 84
Ecoli	5	7	1.1674	143: 77: 35 : 20 : 52
Ecoli2	2	7	2.8223	284: 52
Glass	4	9	1.1280	70: 76: 17 : 29
New-thyroid	3	5	1.7762	150: 35 : 30
Nursery	4	8	2.0267	4320: 4266: 4044: 328
Oilspill	2	49	10.9497	896: 41
Page-blocks	5	10	7.1041	4913: 329: 28 : 88: 115
Penbased	10	16	1.0002	115: 114: 114: 106 : 114: 106 : 105 : 115: 105 : 106
Poker	2	10	16.3789	2050: 25
Satellite	6	36	1.0532	1533: 703 : 1358: 626 : 707 : 1508
Shuttle	3	9	2.6304	1706: 338: 123
Thyroid	3	20	8.2745	17 : 37 : 666
Winequality	2	11	41.0061	880: 20
Yeast	9	8	1.1651	463: 35 : 44 : 51 : 163: 244: 429: 20 : 30
Yeast5	2	8	22.0114	1440: 44

Table 8 – Basic Dataset Characteristics (#C: Number of Classes, #F: Number of Features, Imbalance Ratio, Class Distribution. Minority classes indicated by Equation 4.2 are in bold.)

Here, we report the results of 30 trials of stratified 5-fold cross-validation. In this procedure, we divided the original dataset into five non-intersecting subsets, each of which maintains the original class imbalance ratio. For each fold, we trained each algorithm used in

the experiments using the examples of the remaining fold, and considered the predictive model performance to be the prediction accuracy rate of the induced model tested on the current fold.

Then, we compared the performance of the proposed method with five state-of-art ensemble-based methods that have been proposed in the literature for imbalanced learning. We used the same base classifier and the same size of the resulting ensemble to ensure the fairness of the comparison. It was used the implementation of C4.5 (QUINLAN, 1993) available in the RWeka package (HORNİK; BUCHTA; ZEILEIS, 2009). All the methods used in these experiments return an ensemble composed of ten base classifiers. The ensemble-based methods we used in the experiments included: OverBagging (WANG; YAO, 2009), UnderBagging (BARANDELA; VALDOVINOS; SÁNCHEZ, 2003), SmoteBagging (WANG; YAO, 2009), AdaboostM1 (FREUND; SCHAPIRE, 1997), and RUSBoost (SEIFFERT et al., 2010). We used 20 generations as the stopping criterion in EVINCI, or when the *Saved Ensemble* had reached its maximum G-mean value, i.e., max. G-mean = 1.0, or after producing 10 generations without observing any improvement in the G-mean from that of the *Saved Ensemble*.

Data Set	EVINCI	SmoteBagging	RUSBagging	ROSBagging	AdaboostM1	Rusboost
Abalone	0.0041 (1)	0.0000 (2)	0.0000 (2)	0.0000 (2)	0.0000 (2)	0.0000 (2)
Balance-scale	0.5435 (3)	0.1466 (6)	0.6130 (2)	0.4914 (4)	0.3145 (5)	0.6204 (1)
Car	0.8274 (4)	0.8318 (3)	0.8099 (5)	0.8795 (1)	0.8551 (2)	0.7591 (6)
Chess	0.5956 (4)	0.6144 (3)	0.3218 (5)	0.6731 (2)	0.6759 (1)	0.2866 (6)
Contraceptive	0.5155 (1)	0.5065 (4)	0.5106 (2)	0.5019 (5)	0.4765 (6)	0.5083 (3)
Dermatology	0.9643 (2)	0.9741 (1)	0.9524 (5)	0.9561 (4)	0.9607 (3)	0.9324 (6)
Dnormal	0.8755 (1)	0.8709 (3)	0.8748 (2)	0.8692 (4)	0.8483 (6)	0.8673 (5)
Ecoli	0.7939 (2)	0.7953 (1)	0.7838 (3)	0.7665 (4)	0.7266 (6)	0.7622 (5)
Ecoli2	0.8620 (3)	0.8574 (5)	0.8659 (2)	0.8581 (4)	0.8439 (6)	0.8708 (1)
Glass	0.6621 (1)	0.5564 (5)	0.5709 (4)	0.6352 (3)	0.6574 (2)	0.4511 (6)
New-thyroid	0.9045 (2)	0.8908 (6)	0.8998 (3)	0.8984 (4)	0.8937 (5)	0.9128 (1)
Nursery	0.9330 (4)	0.9602 (3)	0.9062 (5)	0.9700 (2)	0.9830 (1)	0.8988 (6)
Oilspill	0.7678 (2)	0.6604 (5)	0.7980 (1)	0.7026 (4)	0.6174 (6)	0.7519 (3)
Page-blocks	0.9424 (1)	0.8841 (4)	0.9205 (2)	0.8665 (5)	0.8308 (6)	0.9101 (3)
Penbased	0.9156 (5)	0.9290 (2)	0.9261 (4)	0.9278 (3)	0.9562 (1)	0.8899 (6)
Poker	0.4877 (1)	0.0179 (6)	0.4218 (2)	0.2328 (3)	0.0894 (4)	0.0751 (5)
Satellite	0.8695 (2)	0.8647 (5)	0.8678 (4)	0.8714 (1)	0.8684 (3)	0.8605 (6)
Shuttle	0.9980 (1)	0.9970 (4)	0.9960 (6)	0.9980 (3)	0.9980 (2)	0.9965 (5)
Thyroid	0.9766 (1)	0.9675 (3)	0.9378 (5)	0.9737 (2)	0.8836 (6)	0.9638 (4)
Winequality	0.6611 (2)	0.0872 (6)	0.5929 (3)	0.2132 (5)	0.2994 (4)	0.6701 (1)
Yeast	0.2780 (2)	0.1346 (5)	0.1507 (3)	0.1444 (4)	0.0000 (6)	0.3064 (1)
Yeast5	0.9586 (1)	0.8978 (4)	0.9555 (3)	0.8725 (5)	0.8241 (6)	0.9565 (2)
G-Mean Average	0.7426	0.6566	0.7126	0.6956	0.6638	0.6932
Ranking Count	9-7-2-3-1-0	2-2-5-4-5-4	1-7-5-3-5-1	2-4-4-8-4-0	3-4-2-2-2-9	5-2-3-1-4-7
Ranking Average	2.09	3.91	3.32	3.36	4.05	3.82

Table 9 – G-mean Values Achieved by Different Methods in the Experiments over 30 Runs with their Ranks by Dataset (between parentheses), G-mean Average for Each Method, Ranking Count for Each Method, and Ranking Average

4.6.2 Experimental Results - Compared Methods

In this section, we present the experimental results obtained for each dataset listed in Table 8. For each dataset, the six methods executed 5-fold cross-validation ten times. Table 9 shows the average G-mean values obtained by each method and their ranking position (in parentheses) for each dataset in the experiment. The last three rows summarize the results comparing the six methods in all the datasets considered. The first row is the *G-Mean Average* obtained by the compared methods. The *Ranking Count* row shows the number of datasets in which each technique obtained the best G-Mean value, the second best, and so on. For example, the six numbers 3-4-2-2-2-9 in the fifth column represent the ranking of the AdaboostM1 method. It obtained the highest G-Mean value on just three of the 22 datasets, the second highest values on four datasets, and so on. The *Average Ranking* row shows the average of the ranking position obtained by each method on all datasets.

In Table 9, we can see that EVINCI obtained the best G-mean average (0.7426) of all the compared methods. It also received the highest number of best rankings, with nine wins, as well as the lowest average rank of 2.09. The proposed method obtained the first or second best position a total of 16 times, which is twice as many as the method with the second highest G-mean average, i.e., RUSBag. RUSBag obtained an average G-mean score of 0.7126, but obtained the first or second best position result only eight times, with just one at best position and an average rank of 3.32.

The above results indicate that the proposed method achieved the best overall performance. The ranking provided by the Friedman test supports this assumption, showing EVINCI to be the best-ranked method. The Friedman test also rejects the null-hypothesis, i.e., it says that there is a statistically significant difference between the algorithms ($p\text{-value} = 2.190271 \times e^{-3}$). Hence, we executed the Nemenyi post-hoc test for pairwise comparison. In our experiments, the proposed method outperformed SmoteBag, AdaboostM1, and Rusboost with a statistical significance at the 95% confidence level.

4.6.3 Further Analysis

To analyze whether the aggregation of the two objectives positively influences the proposed method, we conducted experiments on the 22 datasets by running EVINCI with only one objective at a time. Table 10 shows the G-mean average obtained in a 30-time 5-fold cross-validation of these analyses. The proposed method used the two objectives in the first column, only the G-mean as the sampling process objective in the G-mean column, and only the density in the overlapping areas in the N1byClass column.

In Table 10, we can see that EVINCI, when using both goals, achieved the best predictive results in most of the studied datasets, 16 out of 22. It also obtained the highest average G-mean of 0.7426 versus 0.7275 and 0.7237 when using only the G-mean and the density in the overlapping

areas as targets, respectively. The Friedman test supports this result, indicating that the proposed method, when using both objectives, obtained the best classification. This means that there is a statistically significant difference between the algorithms ($p\text{-value} = 2.476668 \times e^{-6}$). The Nemenyi post-hoc test result indicates that the original EVINCI outperforms the other versions with a statistical significance at the 95% confidence level.

The second position in the Friedman ranking was obtained when using only the density of the majority classes in the regions of overlap as the genetic algorithm's objective. In fact, this version obtained the second position in half of the experimental datasets and obtained the best G-mean value in four of them. As stated previously, in this version the G-mean of the sample-induced model is not used as the objective of the genetic algorithm. However, classifiers are induced to analyze the ensembles of classifiers resulting from the generation. Despite this, the fact that this version obtained the second-best position is a good indication that a set of samples having a higher density of minority classes in the overlap areas can generate an ensemble of classifiers with high predictive performance. In fact, this version of EVINCI obtained a better G-mean average than the second-best ensemble-based method used in the experiments, i.e., RUSBag, which obtained 0.7237 against 0.7126.

Data Set	Two Objectives	One Objective	
	EVINCI	G-Mean	N1byClass
Abalone	0.0041	0.0000	0.0000
Balance-scale	0.5435	0.5022	0.5273
Car	0.8274	0.8214	0.8140
Chess	0.5956	0.5108	0.3891
Contraceptive	0.5155	0.4965	0.5114
Dermatology	0.9643	0.9527	0.9640
Dnormal	0.8755	0.8626	0.8687
Ecoli	0.7939	0.7588	0.7870
Ecoli2	0.8620	0.8395	0.8528
Glass	0.6621	0.6593	0.6676
New-thyroid	0.9045	0.8861	0.8973
Nursery	0.9330	0.9311	0.9220
Oilspill	0.7678	0.7529	0.7776
Page-blocks	0.9424	0.9412	0.9340
Penbased	0.9156	0.9061	0.9069
Poker	0.4877	0.5836	0.3983
Satellite	0.8695	0.8665	0.8712
Shuttle	0.9980	0.9983	0.9979
Thyroid	0.9766	0.9610	0.9689
Winequality	0.6611	0.6273	0.6617
Yeast	0.2780	0.2326	0.2464
Yeast5	0.9586	0.9147	0.9578
G-Mean Average	0.7426	0.7275	0.7237
Ranking Count	16-6-0	2-5-15	4-11-7
Ranking Average	1.27	2.59	2.13

Table 10 – G-mean Achieved by Different Versions in the Experiments over 30 Runs, G-mean Average for Each Version, Ranking Count for Each Version, and Ranking Average

As an example of the processing of the proposed method and the evolution of the samples, consider Figure 11. Figure 11.A shows a sample (individual) randomly taken from the initial population and Figure 11.B shows a random individual from the fifth generation. These are samples from an artificial dataset (*Dnormal* in Table 8), in which the green color indicates the minority class. As we can see, the sample shown in Figure 11.A has a low overall imbalance ratio, but has separation boundaries without any particular class dominance. In Figure 11.B, we note that the majority classes had an increased number of instances, but their borders of separation with the minority classes contained more instances of the minority class. A C4.5 classifier induced by sample A obtained accuracy values in the red, black, and green classes, respectively, of 0.9343, 0.8649, and 0.8656 and that by sample B obtained accuracy values of 0.9536, 0.8649, and 0.9402, respectively. This represents a big increase in the recognition rate of the minority class. This improvement of accuracy with respect to the minority classes was observed frequently in the other datasets used in the experiments.

Figure 11 – Figure A Represents a Sample taken from Initial Population and Figure B a Sample From Fifth Generation

4.7 Conclusion

In this paper, we presented a new evolutionary ensemble-based method for multi-class imbalanced learning, which we named the evolutionary inversion of class distribution for imbalanced learning (EVINCI). Using a MOEA, EVINCI evolves a set of samples taken from an unbalanced dataset to induce an ensemble of classifiers with high predictive accuracy. The evolutionary guidance of the proposed method is based on studies that indicate that the main difficulty experienced by classification algorithms for unbalanced datasets is related to overlapping areas. To address this issue, we developed the data complexity measure N1byClass for use by EVINCI, which produces a matrix of values that estimates the percentage of overlap in each class with other classes.

With the help provided by N1byClass and the accuracy of the model induced by the samples, EVINCI selectively reduces the concentration of less representative instances of the majority classes in the overlapping areas while selecting samples that produce more accurate models. To increase its generalization power and reduce information loss associated with the selection process, the EVINCI classification system comprises an ensemble of classifiers, in which each optimized sample induces a different base classifier.

We performed experiments on 22 datasets with different imbalance ratios and numbers of classes ranging from 2 to 18, and the results showed that EVINCI outperforms other relevant methods in most cases. In fact, the proposed method obtained the best G-mean average (0.7426), the highest number of wins (9), and the lowest average rank (2.09). Another interesting point is that, by summing the number of wins and second places obtained by EVINCI in its rank by dataset, the total is twice that obtained by the second-best method, RUSBag.

Further investigation revealed EVINCI's efficiency in the aggregation of the two objective functions, namely, a lower density of majority classes in the overlapping areas and accuracy in the sample-generated model. We observed that the combination of the two objectives enhanced the performance of the proposed method, which obtained the best results in most datasets (16) with a statistical significance of 95%, as determined in a Nemenyi posthoc test.

The version that uses only the density of the overlapping regions obtained the second best result, with a G-mean average higher than that of RUSBag. This is a good indication that a set of samples having a higher density of minority classes in the overlapping areas can generate an ensemble of classifiers with high predictive performance. In future work, we may consider investigating the possibility of generating such samples without the need to induce classification models during the evolutionary process.

Acknowledgment

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

4.8 Bibliography

ALCALÁ-FDEZ, J. et al. Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic and Soft Computing*, v. 17, n. 2-3, p. 255–287, 2011. Citation on page 103.

ALI, A.; SHAMSUDDIN, S. M. H.; RALESCU, A. L. Classification with class imbalance problem: A review. In: . [S.l.: s.n.], 2015. Citations on pages 30 and 94.

BACHE, K.; LICHMAN, M. *UCI Machine Learning Repository*. 2013. Available: <<http://archive.ics.uci.edu/ml>>. Citations on pages 54, 64, 72, 73, and 103.

BARANDELA, R.; VALDOVINOS, R.; SÁNCHEZ, J. New applications of ensembles of classifiers. *Pattern Analysis & Applications*, v. 6, n. 3, p. 245–256, Dec 2003. Available: <<https://doi.org/10.1007/s10044-003-0192-z>>. Citations on pages 95 and 104.

BHOWAN, U. et al. Evolving Diverse Ensembles using Genetic Programming for Classification with Unbalanced Data. *IEEE Transactions on Evolutionary Computation*, v. 17, n. 3, p. 368–386, 2013. Citations on pages 31, 35, 36, 66, 67, 70, 94, 96, 98, 114, 116, and 117.

BREIMAN, L. Bagging predictors. *Machine Learning*, v. 24, n. 2, p. 123–140, 1996. ISSN 1573-0565. Available: <<http://dx.doi.org/10.1023/A:1018054314350>>. Citations on pages 32, 50, 66, and 95.

BUNKHUMPORNPAT, C.; SINAPIROMSARAN, K.; LURSINSAP, C. MUTE: Majority under-sampling technique. In: *2011 8th International Conference on Information, Communications & Signal Processing*. [S.l.]: IEEE, 2011. p. 1–4. Citations on pages 65 and 93.

CHANDRA, A.; YAO, X. Ensemble learning using multi-objective evolutionary algorithms. *J. Math. Model. Algorithms*, v. 5, n. 4, p. 417–445, 2006. Available: <<http://dx.doi.org/10.1007/s10852-005-9020-3>>. Citations on pages 36, 48, 51, 63, 92, 98, and 117.

CHAWLA, N. V. et al. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, v. 16, p. 321–357, 2002. Citations on pages 29, 64, 65, 93, and 115.

DAS, S.; DATTA, S.; CHAUDHURI, B. B. Handling data irregularities in classification: Foundations, trends, and future challenges. *Pattern Recognition*, v. 81, p. 674 – 693, 2018. ISSN 0031-3203. Available: <<http://www.sciencedirect.com/science/article/pii/S0031320318300931>>. Citation on page 90.

DEB, K. et al. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Trans. Evol. Comp*, IEEE Press, Piscataway, NJ, USA, v. 6, n. 2, p. 182–197, Apr. 2002. ISSN 1089-778X. Available: <<http://dx.doi.org/10.1109/4235.996017>>. Citations on pages 53, 67, 98, and 117.

DIETTERICH, T. G. Machine-learning research – four current directions. *AI MAGAZINE*, v. 18, p. 97–136, 1997. Citations on pages 31, 48, 63, 67, 92, and 95.

FAN, W. et al. Adacost: Misclassification cost-sensitive boosting. In: *Proceedings of the Sixteenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999. (ICML '99), p. 97–105. ISBN 1-55860-612-2. Available: <<http://dl.acm.org/citation.cfm?id=645528.657651>>. Citations on pages 33 and 96.

FERNANDES, E. R. Q.; CARVALHO, A. C. P. L. F. de; COELHO, A. L. V. An evolutionary sampling approach for classification with imbalanced data. In: *IEEE. Neural Networks (IJCNN), 2015 International Joint Conference on*. [S.l.], 2015. p. 1–7. Citations on pages 67, 96, 98, 116, and 117.

FERNÁNDEZ, A. et al. Analysing the classification of imbalanced data-sets with multiple classes: Binarization techniques and ad-hoc approaches. *Knowledge-Based Systems*, v. 42, p. 97 – 110, 2013. ISSN 0950-7051. Available: <<http://www.sciencedirect.com/science/article/pii/S0950705113000300>>. Citations on pages 28, 62, and 92.

FREUND, Y.; SCHAPIRE, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.*, Academic Press, Inc., Orlando, FL, USA, v. 55, n. 1, p. 119–139, Aug. 1997. ISSN 0022-0000. Available: <<http://dx.doi.org/10.1006/jcss.1997.1504>>. Citations on pages 32, 50, 66, 79, 95, and 104.

FRIEDMAN, J. H.; RAFSKY, L. C. Multivariate generalizations of the wald-wolfowitz and smirnov two-sample tests. *Ann. Statist.*, The Institute of Mathematical Statistics, v. 7, n. 4, p. 697–717, 07 1979. Available: <<https://doi.org/10.1214/aos/1176344722>>. Citation on page 91.

GALAR, M. et al. A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, v. 42, n. 4, p. 463–484, 2012. Citations on pages 32, 33, 63, 95, and 114.

GARCÍA, V.; MOLLINEDA, R. A.; SÁNCHEZ, J. S. On the k-nn performance in a challenging scenario of imbalance and overlapping. *Pattern Analysis and Applications*, v. 11, n. 3, p. 269–280, Sep 2008. ISSN 1433-755X. Available: <<https://doi.org/10.1007/s10044-007-0087-5>>. Citation on page 90.

GRAHAM, R. L.; HELL, P. On the history of the minimum spanning tree problem. *Annals of the History of Computing*, v. 7, n. 1, p. 43–57, Jan 1985. ISSN 0164-1239. Citations on pages 96 and 97.

HANSEN, L. K.; SALAMON, P. Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.*, IEEE Computer Society, Washington, DC, USA, v. 12, n. 10, p. 993–1001, Oct. 1990. ISSN 0162-8828. Available: <<http://dx.doi.org/10.1109/34.58871>>. Citations on pages 31, 95, and 117.

HART, P. E. The condensed nearest neighbor rule (corresp.). *IEEE Transactions on Information Theory*, v. 14, n. 3, p. 515–516, 1968. Citations on pages 49, 65, and 93.

HE, H. et al. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In: *IEEE Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*. [S.l.], 2008. p. 1322–1328. Citations on pages 65 and 94.

HO, T. K.; BASU, M. Complexity measures of supervised classification problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 24, n. 3, p. 289–300, Mar 2002. ISSN 0162-8828. Citation on page 90.

HORNIK, K.; BUCHTA, C.; ZEILEIS, A. Open-source machine learning: R meets Weka. *Computational Statistics*, v. 24, n. 2, p. 225–232, 2009. Citations on pages 104 and 131.

HU, S. et al. Msmote: Improving classification performance when training data is imbalanced. In: *Computer Science and Engineering, 2009. WCSE '09. Second International Workshop on*. [S.l.: s.n.], 2009. v. 2, p. 13–17. Citations on pages 65 and 94.

KHOSHGOFTAAR, T. M.; REBOURS, P. Improving software quality prediction by noise filtering techniques. *Journal of Computer Science and Technology*, v. 22, n. 3, p. 387–396, May 2007. ISSN 1860-4749. Available: <<https://doi.org/10.1007/s11390-007-9054-2>>. Citation on page 94.

KUBAT, M.; MATWIN, S. Addressing the curse of imbalanced training sets: One-sided selection. In: *In Proceedings of the Fourteenth International Conference on Machine Learning*. [S.l.]: Morgan Kaufmann, 1997. p. 179–186. Citations on pages 30, 49, 65, 93, and 116.

LIN, M.; TANG, K.; YAO, X. Dynamic sampling approach to training neural networks for multiclass imbalance classification. *IEEE Trans. Neural Netw. Learning Syst.*, v. 24, n. 4, p. 647–660, 2013. Available: <<http://dblp.uni-trier.de/db/journals/tnn/tnn24.html#LinTY13>>. Citations on pages 66, 72, 74, 79, and 94.

LÓPEZ, V. et al. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences*, v. 250, n. Supplement C, p. 113 – 141, 2013. ISSN 0020-0255. Available: <<http://www.sciencedirect.com/science/article/pii/S0020025513005124>>. Citation on page 90.

POLI, R.; LANGDON, W. B. Genetic programming with one-point crossover. In: _____. *Soft Computing in Engineering Design and Manufacturing*. London: Springer London, 1998. p. 180–189. ISBN 978-1-4471-0427-8. Citations on pages 70 and 101.

PRATI, R. C.; BATISTA, G. E. A. P. A.; SILVA, D. F. Class imbalance revisited: a new experimental setup to assess the performance of treatment methods. *Knowledge and Information Systems*, Springer London, p. 1–24, 2014. Available: <<http://dx.doi.org/10.1007/s10115-014-0794-3>>. Citations on pages 62, 90, and 100.

QIAN, Y. et al. A resampling ensemble algorithm for classification of imbalance problems. *Neurocomputing*, Elsevier, v. 143, p. 57–67, Nov. 2014. Citations on pages 31, 66, 94, and 114.

QUINLAN, R. *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann Publishers, 1993. Citation on page 104.

RIFKIN, R.; KLAUTAU, A. In defense of one-vs-all classification. *J. Mach. Learn. Res.*, JMLR.org, v. 5, p. 101–141, Dec. 2004. ISSN 1532-4435. Available: <<http://dl.acm.org/citation.cfm?id=1005332.1005336>>. Citations on pages 29 and 93.

SÁEZ, J. A. et al. Smote–ipf: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering. *Information Sciences*, v. 291, p. 184–203, 2015. ISSN 0020-0255. Available: <<http://www.sciencedirect.com/science/article/pii/S0020025514008561>>. Citation on page 94.

SCHÖLKOPF, B. et al. Estimating the support of a high-dimensional distribution. *Neural Computation*, v. 13, n. 7, p. 1443–1471, 2001. Citations on pages 30, 65, and 94.

SEIFFERT, C. et al. Rusboost: A hybrid approach to alleviating class imbalance. *IEEE Trans. Systems, Man, and Cybernetics, Part A*, v. 40, n. 1, p. 185–197, 2010. Available: <<http://dblp.uni-trier.de/db/journals/tsmc/tsmca40.html/#SeiffertKHN10>>. Citations on pages 33, 96, and 104.

SUN, Y.; WONG, A. K. C.; KAMEL, M. S. Classification of imbalanced data: a review. *IJPRAI*, v. 23, n. 4, p. 687–719, 2009. Available: <<http://dx.doi.org/10.1142/S0218001409007326>>. Citations on pages 64, 65, 66, 94, and 115.

SUN, Z.; SONG, Q.; ZHU, X. Using Coding-Based Ensemble Learning to Improve Software Defect Prediction. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, v. 42, n. 6, p. 1806–1817, Nov. 2012. Citations on pages 66 and 94.

TANWANI, A. K.; FAROOQ, M. Classification potential vs. classification accuracy: A comprehensive study of evolutionary algorithms with biomedical datasets. In: _____. *Learning Classifier Systems: 11th International Workshop, IWLCS 2008, Atlanta, GA, USA, July 13, 2008, and 12th International Workshop, IWLCS 2009, Montreal, QC, Canada, July 9, 2009, Revised Selected Papers*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. p. 127–144. Available: <https://doi.org/10.1007/978-3-642-17508-4_9>. Citation on page 103.

TUMER, K.; GHOSH, J. Analysis of decision boundaries in linearly combined neural classifiers. *Pattern Recognition*, v. 29, p. 341–348, 1996. Citations on pages 48, 63, 66, 91, and 115.

WANG, S.; YAO, X. Diversity analysis on imbalanced data sets by using ensemble models. *2009 IEEE Symposium on Computational Intelligence and Data Mining*, p. 324–331, 2009. Citations on pages 33, 95, and 104.

WANG, S.; YAO, X. Multiclass imbalance problems: Analysis and potential solutions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, v. 42, n. 4, p. 1119–1130, Aug 2012. ISSN 1083-4419. Citations on pages 29, 67, 93, and 116.

YIN, Q.-Y. et al. A Novel Selective Ensemble Algorithm for Imbalanced Data Classification Based on Exploratory Undersampling. *Mathematical Problems in Engineering*, Hindawi Publishing Corporation, v. 2014, p. 1–14, 2014. Citations on pages 31, 33, 66, 71, 73, 75, 94, 100, 114, and 117.

AN ENSEMBLE OF CONVOLUTIONAL NEURAL NETWORKS FOR UNBALANCED DATASETS: A CASE STUDY WITH WAGON COMPONENT INSPECTION

Authors:

Everlandio R. Q. Fernandes (everlandio@usp.br)

Rafael L. Rocha (rafael.rocha@pq.itv.org)

Bruno Ferreira (bruno.ferreira@senaipa.org.br)

Eduardo Carvalho (eduardo.isi@sesipa.org.br)

Ana Carolina Siravenha (anacarolina.isi@sesipa.org.br)

Ana Claudia S. Gomes (claudia.isi@senaipa.org.br)

Schubert Carvalho (schubert.carvalho@itv.org)

Cleidson R. B. de Souza (cleidson.desouza@acm.org)

Abstract

Railway component inspection is a technique widely used for maintenance because defective components pose safety issues. Nevertheless, finding defective components is a hard task because they are normally hidden by dusty, which poses hard problems for the image segmentation algorithms. To approach this problem, manual inspection by humans is normally used, but it is time consuming, expensive and sometimes dangerous. Meanwhile, automatic approaches that uses machine learning algorithms are also difficult because the datasets are strongly unbalanced. Such datasets usually induce biased classification models that identify new instances as members of the class with the greatest abundance of examples in the training data. In this paper, we propose a new method that combines the use of Convolutional Neural Networks (CNN) with imbalanced

learning to address the challenge of using machine learning to identify defective components. Our method was tested with real-world data from images used for wagon component inspection. Moreover, we compared our method with an ensemble of MLP networks using features extraction, such as the LeNet, and a CNN network without ensemble learning. Results indicate that our proposed method produced the higher overall accuracy compared to the other methods.

5.1 INTRODUCTION

Railway component inspection is an important issue because train derailments usually occur when there are failures in wheels or axles, objects on the railways or damage tracks (MACUCCI et al., 2016). Moreover, accidents in railways can lead to fatalities and casualties, as well as damage tracks and trains. In addition to financial costs, there might be environmental and social costs associated with repairs where the train derailment takes place.

Overall, inspecting railway components that can potentially cause derailment is an important task for the railways maintenance. Machine vision technology has been widely used for inspecting railways components to increase efficiency, effectiveness, and objectivity (PERNG; LIU; CHANG, 2011). Previous inspections systems that use machine vision include wheel profile and safety appliances (RESENDIZ; HART; AHUJA, 2013). Despite its wide usage, automatic inspection from image understanding is not without challenging: the identification procedure might deal with several stages including image acquisition, preprocessing, feature extraction, and finally classification. The feature extraction stage specially is one that leverages on human knowledge and experience about the problem, thereby a human expert might discover (extract) the best feature set for training a classifier. To address this dependency on human expertise, Convolutional Neural Networks (CNN) have been more and more adopted due to their efficiency in the feature extraction and learning patterns from image data (LECUN et al., 1998). In other words, CNNs have become one of the most used deep learning approaches to classify images (LECUN et al., 1998) (GOODFELLOW; BENGIO; COURVILLE, 2016).

One challenge in machine learning, including CNNs, is to learn a model from unbalanced datasets. Such datasets often arise during the acquisition of images in real world scenarios. As an example, the shear pad (from now referred simply as pad) is one of the most important wagon components that need to be inspected. However, large organizations have a large wagon fleet with a wide variety of wagon models. Furthermore, not all wagons have a pad, and only a very small percentage of them have defects. In short, an automatic wagon inspection system needs to deal with a multi-class dataset with a very high imbalance rate.

Meanwhile, recently published studies (BHOWAN et al., 2013) (YIN et al., 2014) (WANG et al., 2013) (QIAN et al., 2014) have reported the successful use of ensembles of classifiers for classification with unbalanced datasets, where each classifier is induced by a different sample from the original dataset (GALAR et al., 2012). Ensembles are designed to

Figure 12 – Proposed method Workflow. Balanced samples are generated from the unbalanced dataset, each sample will be applied to a CNN, with the results obtained (accuracy and diversity) passed by a non-dominant rank followed by the application of the pruning technique to obtain the ensemble result.

increase the accuracy of their base classifiers by combining them through a consensus function or operator (ZHOU, 2009). Moreover, ensembles have a greater capacity for generalization than their base classifiers; this was formally presented in the study carried out in (TUMER; GHOSH, 1996).

In this context, this work seeks the use of CNN together with ensembles to correctly classify, through images, the condition in which a wagon component is found. More specifically, we used three types of classifiers to evaluate the pads: (i) the first one is CNN (LeNet, CNN architecture described in (ROCHA et al., 2018); (ii) Ensemble of Multi Layer Perceptron (MLP); (iii) Ensemble of Convolutional Neural Networks. The latter being our main proposal for solving the problem described. It is called from now on by Imbalanced Learning with Ensemble of Convolutional Neural Network - ILEC.

The rest of this paper is organized as follows. The next section presents the related works associated with our method. Section III describes the method that we propose. The following section briefly describes the real-world problem where we tested our method. Section V describes the methodology we used to test our method. Section VI focuses on our experimental results and, finally, the last section concludes the paper presenting our plans for future work.

5.2 Related Work

When dealing with imbalanced data sets, the propose solutions can be categorized in two levels: In (i) data and (ii) algorithm level. For the first, the dataset is altered to achieve balance in the class distribution (CHAWLA et al., 2002; ZHOU; LIU, 2006; SUN et al., 2007). In the second, the algorithm is modified to give higher importance to the minority classes (QUINLAN, 1991; ZADROZNY; ELKAN, 2001).

Works studying resample techniques for changing the class distribution can be found in the literature (SUN; WONG; KAMEL, 2009; CHAWLA et al., 2002) and, empirically, show that

a preprocessing of the classes distribution is frequently useful. The resampling techniques can be *oversampling*, *undersampling* or a combination of both. In the oversampling the dataset is expanded to make minority class match the size of the majority class. In the undersampling the majority class is reduced until the size of the minority class.

One of the undersampling strategies to reduce the majority is called Random Undersampling (RUS), however this simple technique may remove useful data. To alleviate such an effect, a directed undersampling may detect and remove less representative instances. The One-sided Selection (OSS) technique (KUBAT; MATWIN, 1997) use this strategy.

As stated previously, recently published studies have reported the successful use of ensembles of classifiers for classification with unbalanced datasets. Several methods that take into account diversity and accuracy of base classifiers have been proposed. Multiobjective Genetic Sampling (MOGASamp) (FERNANDES; CARVALHO; COELHO, 2015) constructs an ensemble of classifiers induced from balanced samples in the training data set. 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 data set and induce classifiers with high accuracy and diversity.

Bhowan and colleagues (BHOWAN et al., 2013) 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.

Wang and colleagues (WANG; YAO, 2012) investigate two types of multi-class 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 ensemble approaches using class decomposition (the one-against-all strategy) and approaches not using class decomposition. According to their experimental results, the use of class decomposition did not provide any advantages in multi-class imbalance learning.

5.3 Imbalanced Learning with Ensemble of Convolutional Neural Network (ILEC)

The primary objective of the proposed method is to build an ensemble of convolution neural network to deal with imbalanced image datasets. For this, the proposed method uses an imbalanced learning technique to construct a series of classifiers that aims to not harm the minority classes. To make the ensemble more accurate and with greater generalization power, a

pruning technique based on a ranking of non-dominance between diversity and accuracy of the classifier is applied to the process. Figure 12 outlines the proposed method, which is detailed below.

Hansen and Salamon (HANSEN; SALAMON, 1990) have demonstrated that under specific conditions an ensemble's expected error rate for a new instance tends to zero as the number of base classifiers tends to infinity. To do so, the base classifiers must have individual accuracy above 50% and be as diverse as possible. Two classifiers are considered diverse if they make mistakes in different instances of the same test dataset. Therefore, diversity and accuracy should be the main objectives when choosing the base classifiers that make up the ensemble.

For this reason, the first action of our method is to repeatedly apply random undersampling to the training dataset to obtain a series of samples with balanced subsets of images. In order to achieve a diverse set of samples that also takes into account the minority class, ILEC selects only 80% of the most minority class in each sample. For example, if the class with the least number of instances has ten images and the dataset has three classes, each sample will have 24 images ($0.8 \times 10 \times 3$), with eight images in each class.

After the sampling process, each subset of images induces a CNN model that is validated using the entire training dataset. That is, the accuracy of the CNN model is calculated verifying its effectiveness in the complete training dataset. The accuracy measure used in this research was the G-mean (YIN et al., 2014). The geometric mean of the classifier correction, G-mean calculation procedure, produces low values if the classifier presents low precision in at least one of the classes. That is, a low G-mean value indicates that the classifier has significant faults in at least one class, which makes this classifier less useful in practice.

To verify the heterogeneity of the models previously generated, the measure of diversity of classifiers Pairwise Failure Crediting (PFC) (CHANDRA; YAO, 2006) is also calculated from the CNN models. This measure is calculated for each model using a pair-wise comparison with all models. It indicates how much a model's response differs from the whole group. The proposed method uses the PFC diversity measure due to the good results in imbalanced learning presented in (BHOWAN et al., 2013) and (FERNANDES; CARVALHO; COELHO, 2015).

Accuracy and diversity are often conflicting objectives, since two classifiers with high accuracy usually have low dissimilarity. For this reason, ILEC proposes the generation of a non-dominance rank (DEB et al., 2002) to select which classifiers will compose their CNN ensemble through a pruning process. The non-dominance ranking is a well-known measure of Pareto-based dominance that computes the number of solutions in a population that dominate a particular solution considering two or more objectives. A non-dominated solution will have the lowest value in the ranking, i.e., 0, while high ranking values indicate low-performance solutions, that is, solutions dominated by many others.

The pruning process used by ILEC sequentially removes the classifiers that have the

Figure 13 – Pad absent

Figure 14 – Undamaged pad

Figure 15 – Damaged pad

highest non-dominance values. This indicates that the first classifiers to come out of the ensemble are those with lower values of accuracy and diversity in the population. Tie situations are decided randomly. Initially and after each pruning iteration, i.e., after a classifier is removed, the accuracy of the ensemble is calculated by majority vote. The pruning process is only confirmed if the resulting ensemble's accuracy is no worse than that of the ensemble of the previous iteration. The process repeats until the elimination of a classifier causes a decrease in the accuracy of the resulting ensemble, or when only the classifiers that are non dominated by any other remain, that is, classifiers with the ranking value equal to 0. When a new example is presented to the ensemble, its class is determined by the majority vote¹ considering the output of each base classifier.

5.4 Statement of the problem

In many railway companies, the wagon maintenance is performed by an employee, who has to evaluate, in a given period of time, several items including the compression bar, triangle, an adapter of the bearing box, and a plate of support of the coupling. As wagons are often the most representative asset in railway operations, this maintenance should be more accurate and optimized. The visual inspection by humans place employees in risk situation since they need to be in dangerous places (for instance, very close to the trains) in order to inspect those items (HART et al., 2008).

To be more specific, we are interested in a wagon component called *pad*. The pad is part of the railway truck, which is a structure underneath the wagon, or locomotive, to which other important components are attached, such as wheels, hence, etc. Typically two trucks are fitted to the wagon and have 4 wheels to support the cart. The pad is composed of metal and rubber and has a role similar to a damper. It is positioned between each of the side frame pedestals and the corresponding roller bearing adapter (IWNICKI, 2006).

This work is part of a large project aiming the construction of an automatic inspection system for wagon components for the Vale S.A. Vale is the second largest mining company in the world. In Brazil the company operates approximately 2,000 kilometers of railroad tracks, and this type of transportation plays a fundamental role to its operations because the company has

¹ Note that other voting strategies could be used without large impact in our method.

one of the biggest trains in the world, made up of four locomotives and 330 wagons. Furthermore, the company transports iron ore through its railways 24 hours per day, 7 days per week.

Due to a large quantity of wagons, models, and the aggressive operating environment, the pad may suffer all kinds of damages. A camera system capture images that are processed by image processing and computer vision algorithms. Specifically, in this paper, we test our method in a task of pad classification based on the collected images. There might be 3 classes: broken pad, absent pad and pad without problems. The details of how we conducted our tests are described in the following section.

5.5 Experiments

This session describes the methodology used in this paper, as well the materials used. The subsection 5.5.1 presents the description of the dataset used to train and test the various classification methods. The subsection 5.5.2 describes the methodology of texture feature extraction, by Discrete Wavelet Transform and Gray-level Co-Occurrence Matrices, and the post-processing steps of feature vector composition and data normalization. And lastly, the subsection 5.5.3 explains about the deep learning methods used in the tests.

5.5.1 Database

Real images captured to compose the database were taken from the entire truck and reduced to the region of interest (pad). To standardizing the final resolution of the images was set to $(128 \times 256 \times 1)$ (rows $\times$ columns $\times$ channels). Due camera resolution, the images were captured in gray scale, it explains the third dimension in the final resolution.

The database has a total of 334 images, divided into 3 distinct classes or labels, which are: class 1 (pad absent - Fig. 13), class 2 (undamaged pad - Fig. 14) and class 3 (damaged pad - Fig. 15), with 53, 241 and 40 images, respectively. It is important to notice that these images were collected from real trucks, and pads, through our industrial partner. This shows an unbalanced dataset where a large number of images is concentrated in undamaged pad class. The dataset composed of 334 images were divided 80% for training and 20% for testing.

5.5.2 Texture analysis

To prepare the input of the ANN, Discrete Wavelet Transform (DWT) and Gray-Level Co-Occurrence Matrices (GLCM) were used to generate the feature vector with 128 coefficients. The measures of four energy, four moments and deviation of DWT directional decompositions (vertical, horizontal and diagonal) in 4 levels generated 108 coefficients of a feature vector. Five attributes were extracted from GLCM, they are the angular second moment (ASM), inverse different moment (IDM), entropy, contrast and correlation at four distinct distances, which

in degrees are $\alpha = \{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$, obtaining 20 coefficients. Then the feature vector is normalized with min-max normalization technique, used for instance by Han et. al (HAN; KAMBER; PEI, 2012), and Siravenha and Carvalho (SIRAVENHA; CARVALHO, 2016).

In experiments with this technique, repeated random undersampling in the training dataset generate ten balanced samples. Therefore, each of these samples induces an MLP classifier, producing an ensemble of MLPs. A simple but efficient consensus function, majority vote, indicates the final response of the classification system. The experiments were repeated using different amounts of epochs in MLP training — 50, 100 and 500.

Table 11 – Comparison of approaches. The MPL, LeNet, CNN, and ILEC approaches were compared at different epochs according to the G-mean, Standard Deviation (SD) and accuracy of each of the classes.

		G-mean	SD	class 1	class 2	class 3
Ensemble MLP - Feature Extraction	50 ep.	0.8137	0.0353	0.7364	0.7653	0.9625
	100 ep.	0.8728	0.0304	0.8000	0.8347	1.0000
	500 ep.	0.9402	0.0261	0.9091	0.9163	1.0000
LeNet	20 ep.	0.8413	0.3004	0.9636	0.9694	0.8000
	50 ep.	0.9622	0.0220	0.9909	1.0000	0.9000
CNN	20 ep.	0.8751	0.1551	0.9091	0.9735	0.8125
	50 ep.	0.8548	0.3006	0.9727	1.0000	0.7875
ILEC	10 ep.	0.9358	0.0247	0.9988	0.9079	0.9234
	20 ep.	0.9711	0.0166	0.9943	0.9468	0.9922
	30 ep.	0.9668	0.0220	0.9955	0.9452	0.9875
	40 ep.	0.9664	0.0225	1.0000	0.9398	0.9875
	50 ep.	0.9435	0.0230	0.9864	0.9398	0.9656

5.5.3 Deep Learning Approach

The feature extraction is a costly and complex process, which requires knowledge from the domain. The adequate selection of the features to be extracted greatly affect the performance of the system. The Convolutional Neural Networks (CNN) (LECUN, 1989) is at the state of the art for the pattern recognition task. Its differential is the integration of the feature extraction with the classification. The network can be divided into two parts, the first is the feature extractor, that combines convolutional and pooling layers, the second part is usually a fully connected layer responsible for the classification.

5.5.3.1 LeNet

One of the first CNN model for training images (LECUN et al., 1998), firstly used for handwriting recognition, the LeNet is one of the principal models to compare others methods and architectures. The network is composed of seven layers, with two convolutional (*conv1* and *conv2*), two pooling (*pool1* and *pool2*) and three fully connected (*fully1*, *fully2* and *fully3*).

Among many points are important in LeNet, such as max pooling, different activation functions (sigmoid and tanh nonlinearities), fully connected layers for classification.

The fundamental difference in this work is related to the use of the number of fully connected layers, which in this work we use 2 (*fully1* and *fully2*), and the number of neurons used in *fully1*, which in this case is 500.

Experiments with this technique test its efficiency in the problem addressed by this study using the complete training dataset as input to the network. It is also being evaluated its efficacy with different amount of epochs to train the network — 20 and 50.

5.5.3.2 CNN Architecture

The CNN architecture that composes the ILEC proposal was previously published in (ROCHA et al., 2017) and (ROCHA et al., 2018), and consist of two convolutional layers (*conv1* and *conv2*), two fully connected layers (*fully1* and *fully2*) and one MaxPooling layer (*pool1*). The sequence that the layers has been implemented is *conv1*, *conv2*, *pool1*, *fully1* and *fully2*.

Conv1 is the input layers that receives a gray scale image with resolution of 128x256 as input, has a kernel size of 3 and strides of 2 with ReLU activation function and 32 output filters. The next layer (*conv2*) has the same kernel size, stride and activation function of *conv1*, but with 64 output filters. Both *conv1* and *conv2* is no padding.

The third layer is *pool1* and has pool size and stride of 2 with no padding. The next layer is *fully1* with 128 outputs neurons with ReLU activation function. The last layer is a fully connected with 3 outputs by softmax activation function corresponding to the probability distributions of three classes with cross-entropy loss function.

This proposal of Convolutional Neural Network is the base classifier used by ILEC. Thus, experiments with this CNN are divided into two groups. The first one without any treatment for the unbalanced dataset. In this way, experiments use the complete training dataset with different amounts of epochs to train the CNN — 20 and 50. The second is as part of ILEC, described in Section 5.3. ILEC initially produces ten balanced samples, using in each sample only 80% of the examples of the class with the least amount of instances (minority class), as described in that section. ILEC also is tested with different amount of epochs — 10, 20, 30, 40 and 50.

5.6 Results

Table 11 summarizes the results of the experiments we described in the previous section. As previously stated, we trained our approach with different sets of epochs, namely: 10, 20, 30, 40, and 50. We can notice that after 20 epochs, the convolutional neural networks begin to show signs of overfitting, therefore reducing the overall accuracy of our method. Therefore, in the rest of this section we will focus specifically on the ILEC trained with 20 epochs.

According to Table 11, the proposed method has the best overall accuracy (G-mean: 0.9711) when compared to other methods. The closest approach to ILEC is the LeNet with 50 epochs (G-mean: 0.9622). However, the class of most interest in this study, i.e., damaged pad (class 3), presents a much lower result (0.9000) when compared to the proposed method (0.9922). It is also important to notice that the proposed method has the smaller standard deviation among all the tested methods.

When we compare the two tested ensembles, it is possible to note that both MLPs (100, and 500 epochs) present a 100% accuracy for class 3 compared to ILEC (99.22%). However, the approach using MLP networks are heavily dependent on the feature extraction method used, as in (PUN; LEE, 2004). Meanwhile, the proposed method is based on convolutional networks where such extraction is not needed (LECUN et al., 1998).

Rocha and colleagues (ROCHA et al., 2018) have developed a network architecture specific for pad classification. This is the CNN method described on section 5.5.3 and mentioned on Table 11. Given the focus of the CNN, we decided to adopt this as the base classifier in the ILEC. Another option would be to use the Lenet, but this is a classic classifier (LECUN et al., 1998). Again, it is worth noting that ILEC's results are better than CNN's. This suggests that ILEC might be able to improve the results of any convolutional network that is designed for a specific goal, since the base classifier is only one component of the proposed method. This, however, needs to be further explored.

Finally, it is important to mention that in this paper we have not taken into account the time necessary to train the different methods. We do recognize that this is an important aspect to be addressed. Since we are using an ensemble of convolutional networks, we believe our training time will be reduced because these networks will be trained in parallel. We plan to validate this in our future work.

5.7 Conclusions and Future Work

In this paper, we proposed a new method that uses an ensemble of convolutional neural networks for unbalanced datasets. We tested our method with 3 other methods: an ensemble of MLP networks, the LeNet, and the CNN networks. We used images from the pad, a wagon component. Such images are used to identify defective components in a railway inspection process. The results suggest that the proposed method has the best overall accuracy, the smaller standard deviation among all the tested methods, while at the same time has a good accuracy for all classes, especially on the class of interest.

We plan to test our method with additional industrial datasets to further validate ILEC. In addition, an interesting question is whether the proposed method is able to improve the results of other convolutional networks, since these networks will be only one component of the proposed method. As previously mentioned, we will assess the training time of our methods to find out

its potential impact in real-world scenarios. Finally, we will augment our dataset with Salt and Pepper, Gaussian and Poison noises besides rotation, translation, and reduction of pixels in the image (GONZALEZ; WOODS, 2008).

5.8 Bibliography

BHOWAN, U. et al. Evolving Diverse Ensembles using Genetic Programming for Classification with Unbalanced Data. *IEEE Transactions on Evolutionary Computation*, v. 17, n. 3, p. 368–386, 2013. Citations on pages 31, 35, 36, 66, 67, 70, 94, 96, 98, 114, 116, and 117.

CHANDRA, A.; YAO, X. Ensemble learning using multi-objective evolutionary algorithms. *J. Math. Model. Algorithms*, v. 5, n. 4, p. 417–445, 2006. Available: <<http://dx.doi.org/10.1007/s10852-005-9020-3>>. Citations on pages 36, 48, 51, 63, 92, 98, and 117.

CHAWLA, N. V. et al. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, v. 16, p. 321–357, 2002. Citations on pages 29, 64, 65, 93, and 115.

DEB, K. et al. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Trans. Evol. Comp.*, IEEE Press, Piscataway, NJ, USA, v. 6, n. 2, p. 182–197, Apr. 2002. ISSN 1089-778X. Available: <<http://dx.doi.org/10.1109/4235.996017>>. Citations on pages 53, 67, 98, and 117.

FERNANDES, E. R. Q.; CARVALHO, A. C. P. L. F. de; COELHO, A. L. V. An evolutionary sampling approach for classification with imbalanced data. In: IEEE. *Neural Networks (IJCNN), 2015 International Joint Conference on*. [S.l.], 2015. p. 1–7. Citations on pages 67, 96, 98, 116, and 117.

GALAR, M. et al. A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, v. 42, n. 4, p. 463–484, 2012. Citations on pages 32, 33, 63, 95, and 114.

GONZALEZ, R. C.; WOODS, R. E. *Digital image processing*. Upper Saddle River, N.J.: Prentice Hall, 2008. Citation on page 123.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. <<http://www.deeplearningbook.org>>. Citation on page 114.

HAN, J.; KAMBER, M.; PEI, J. *Data mining: concepts and techniques*. [S.l.]: Morgan Kaufmann - Elsevier, 2012. 113-114 p. Citation on page 120.

HANSEN, L. K.; SALAMON, P. Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.*, IEEE Computer Society, Washington, DC, USA, v. 12, n. 10, p. 993–1001, Oct. 1990. ISSN 0162-8828. Available: <<http://dx.doi.org/10.1109/34.58871>>. Citations on pages 31, 95, and 117.

HART, J. et al. Machine vision using multi-spectral imaging for undercarriage inspection of railroad equipment. In: *Proceedings of the 8th World Congress on Railway Research, Seoul, Korea*. [S.l.: s.n.], 2008. Citation on page 118.

IWNICKI, S. Handbook of railway vehicle dynamics. In: _____. [S.l.]: CRC Press, 2006. p. 548. ISBN 9780849333217. Citation on page 118.

KUBAT, M.; MATWIN, S. Addressing the curse of imbalanced training sets: One-sided selection. In: *In Proceedings of the Fourteenth International Conference on Machine Learning*. [S.l.]: Morgan Kaufmann, 1997. p. 179–186. Citations on pages 30, 49, 65, 93, and 116.

LECUN, Y. Generalization and network design strategies. *Connectionism in perspective*, Zurich, Switzerland: Elsevier, p. 143–155, 1989. Citation on page 120.

LECUN, Y. et al. Gradient-based learning applied to document recognition. In: *Proceedings of the IEEE*. [s.n.], 1998. v. 86, n. 11, p. 2278–2324. Available: <<http://ieeexplore.ieee.org/document/726791/>>. Citations on pages 114, 120, and 122.

MACUCCI, M. et al. Derailment detection and data collection in freight trains, based on a wireless sensor network. *IEEE Transactions on Instrumentation and Measurement*, IEEE, v. 65, n. 9, p. 1977–1987, 2016. Citation on page 114.

PERNG, D.-B.; LIU, H.-W.; CHANG, C.-C. Automated smd led inspection using machine vision. *The International Journal of Advanced Manufacturing Technology*, v. 57, n. 9, p. 1065–1077, Dec 2011. ISSN 1433-3015. Available: <<http://dx.doi.org/10.1007/s00170-011-3338-y>>. Citation on page 114.

PUN, C.-M.; LEE, M.-C. Extraction of shift invariant wavelet features for classification of images with different sizes. *IEEE Transactions Pattern Analysis and Machine Intelligence*, v. 26, n. 9, p. 1228–1233, 2004. Citation on page 122.

QIAN, Y. et al. A resampling ensemble algorithm for classification of imbalance problems. *Neurocomputing*, Elsevier, v. 143, p. 57–67, Nov. 2014. Citations on pages 31, 66, 94, and 114.

QUINLAN, J. R. Improved estimates for the accuracy of small disjuncts. *Machine Learning*, Springer, v. 6, n. 1, p. 93–98, 1991. Citations on pages 64 and 115.

RESENDIZ, E.; HART, J. M.; AHUJA, N. Automated visual inspection of railroad tracks. *IEEE Transactions on Intelligent Transportation Systems*, v. 14, n. 2, p. 751–760, June 2013. ISSN 1524-9050. Citation on page 114.

ROCHA, R. L. et al. Avaliação de técnicas de deep learning aplicadas à identificação de peças defeituosas em vagões de trem. In: CLUA, E.; PÁDUA, F. L. C. (Ed.). *Workshop of Industry Applications (WIA) in the 30th Conference on Graphics, Patterns and Images (SIBGRAP'17)*.

Niterói, RJ, Brazil: [s.n.], 2017. Available: <<http://sibgrapi2017.ic.uff.br/>>. Citation on page 121.

ROCHA, R. L. et al. A deep-learning-based approach for automated wagon component inspection (in press). In: *In proceedings of SAC 2018: Symposium on Applied Computing, Pau, France, April 9-13, 2018, 8 pages*. [S.l.: s.n.], 2018. Citations on pages 115, 121, and 122.

SIRAVENHA, A. C.; CARVALHO, S. R. Plant classification from leaf textures. In: *2016 International Conference on Digital Image Computing: Techniques and Applications, DICTA 2016, Gold Coast, Australia, November 30 - December 2, 2016*. [s.n.], 2016. p. 1–8. Available: <<https://doi.org/10.1109/DICTA.2016.7797073>>. Citation on page 120.

SUN, Y. et al. Cost-sensitive boosting for classification of imbalanced data. *Pattern Recogn.*, Elsevier Science Inc., New York, NY, USA, v. 40, n. 12, p. 3358–3378, Dec. 2007. ISSN 0031-3203. Available: <<http://dx.doi.org/10.1016/j.patcog.2007.04.009>>. Citations on pages 27, 33, 62, 64, and 115.

SUN, Y.; WONG, A. K. C.; KAMEL, M. S. Classification of imbalanced data: a review. *IJPRAI*, v. 23, n. 4, p. 687–719, 2009. Available: <<http://dx.doi.org/10.1142/S0218001409007326>>. Citations on pages 64, 65, 66, 94, and 115.

TUMER, K.; GHOSH, J. Analysis of decision boundaries in linearly combined neural classifiers. *Pattern Recognition*, v. 29, p. 341–348, 1996. Citations on pages 48, 63, 66, 91, and 115.

WANG, J. et al. Ensemble of Cost-Sensitive Hypernetworks for Class-Imbalance Learning. In: *2013 IEEE International Conference on Systems, Man, and Cybernetics*. [S.l.]: IEEE, 2013. p. 1883–1888. Citations on pages 31, 66, 75, and 114.

WANG, S.; YAO, X. Multiclass imbalance problems: Analysis and potential solutions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, v. 42, n. 4, p. 1119–1130, Aug 2012. ISSN 1083-4419. Citations on pages 29, 67, 93, and 116.

YIN, Q.-Y. et al. A Novel Selective Ensemble Algorithm for Imbalanced Data Classification Based on Exploratory Undersampling. *Mathematical Problems in Engineering*, Hindawi Publishing Corporation, v. 2014, p. 1–14, 2014. Citations on pages 31, 33, 66, 71, 73, 75, 94, 100, 114, and 117.

ZADROZNY, B.; ELKAN, C. Learning and making decisions when costs and probabilities are both unknown. In: *ACM. Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.], 2001. p. 204–213. Citations on pages 64 and 115.

ZHOU, Z.-H. Ensemble learning. In: LI, S. Z.; JAIN, A. K. (Ed.). *Encyclopedia of Biometrics*. [S.l.]: Springer US, 2009. p. 270–273. Citations on pages 48, 63, 66, and 115.

ZHOU, Z.-H.; LIU, X.-Y. Training cost-sensitive neural networks with methods addressing the class imbalance problem. *Knowledge and Data Engineering, IEEE Transactions on*, IEEE, v. 18, n. 1, p. 63–77, Jan. 2006. ISSN 1041-4347. Available: <<http://dx.doi.org/10.1109/tkde.2006.17>>. Citations on pages 62, 64, and 115.

Conclusion

Imbalanced datasets are present in many real world applications, such as rare disease studies, bank fraud detection, defective pieces identification on production lines, to mention some. The main problem resulting from imbalanced datasets is that when classic machine learning algorithms are applied to these datasets, the data imbalance can cause a bias in the generated classification model, which would tend to classify new instances as belonging to the majority classes. Solutions able to deal with this problem are investigated in an area of machine learning known as imbalanced learning.

In this thesis, the candidate investigated solutions for the construction of classification systems generated from imbalanced datasets. More specifically, the candidate proposed three solutions based on evolutionary ensemble learning and a fourth solution for imbalanced datasets of images, using a ensemble of CNNs and an ensemble pruning method based on non-dominance ranking.

The first proposal - MOGASamp, presented in Chapter 2, deals with imbalanced binary datasets. This proposal involves the customization of a multiobjective evolutionary algorithm to optimize the processing of sampling subsets from the original dataset, which are then used to train individual classifiers that are later combined in an ensemble of classifiers.

The candidate used evolutionary algorithms because they have several positive aspects that can benefit the ensemble building process. First, they allow the use, as one of the objectives of the MOEA, that the classifier induced by a sample (an individual in the evolutionary algorithm) has high predictive accuracy in both classes. Thus, the proposed method can look for samples that contain the most significant instances of the majority class and keep the correct classification of instances from the minority class. Afterwards, evolutionary algorithms optimize a set of solutions, which are used to construct the idealized ensemble of classifiers. To increase the chances of the method returning an effective ensemble of classifiers with a high generalization ability, the candidate reinforced the search for diversified solutions using the PFC diversity measure as the second objective of the MOEA and developed a mechanism that eliminates similar solutions after the reproduction procedure. The candidate experimentally compared the predictive performance of MOGASamp with six state-of-the-art methods for imbalanced learning. In these comparisons, the proposed method presented an overall superior predictive performance, in terms of accuracy measures, for both classes.

As previously mentioned, most of the proposed techniques for imbalanced learning were designed and tested only for binary dataset scenarios. Unfortunately, when working with a multiclass dataset, the solutions proposed in the literature are not directly applicable or achieve below-expected performance. In Chapter 3, the candidate presented the first proposal for imbalanced multiclass datasets - E-MOSAIC.

E-MOSAIC can be considered an extension of MOGASamp. However, the candidate

made significant changes to be able to handle multiclass datasets. Multiclass datasets may have more than one class of interest, i.e., multiple classes that need to have a high degree of predictive accuracy. For this reason, the proposed method uses as the conflicting objectives of the customized MOEA the accuracy of the classifier induced by the sample (individual) for each class of the dataset, trying to increase the predictive accuracy for all classes. E-MOSAIC also presents an additional step aiming at improving the predictive performance of the ensemble of classifiers returned from the evolutionary process. The ensemble delivered at the end of the evolutionary process is not necessarily the ensemble that was produced in the last generation, but rather, the ensemble that presented the best accuracy during the evolutionary process, considering G-mean and mAUC accuracy measures. The experiments compared the predictive performance of E-MOSAIC with state-of-the-art imbalanced classification methods, including methods based on resampling, active learning, cost-sensitive and boosting. According to the experimental results, the proposed method obtained the best predictive performance regarding the accuracy measures for the multiclass datasets used in the experiments.

Based on several studies that indicated that the main reason for the low predictive performance of the classical classification algorithms in imbalanced datasets is related to the degree of overlap between the classes and the imbalance ratio in the overlapping areas, the candidate proposed and described, in Chapter 4, the EVINCI method. EVINCI is also a evolutionary-based ensemble method designed to deal with binary and multiclass imbalanced datasets. The main inspiration is to use MOEA to selectively reduce the concentration of less representative instances of the majority classes in the overlapping areas, while selecting samples that produce more accurate models. To guide this evolutionary process, EVINCI considers the accuracy of the model induced by the sample and a specially developed measure of dataset complexity, named N1byClass. This measure estimates the percentage of overlap for each pair of classes. EVINCI inherits from E-MOSAIC the procedures that promote the diversity of solutions and deliver the best ensemble produced during the evolutionary process. The experimental results indicated that the proposed method outperforms other state-of-the-art ensemble-based methods for imbalanced learning. EVINCI obtained the best G-mean average, the best rank average for the 22 datasets analyzed, and the highest number of wins.

To analyze whether the aggregation of the two objectives positively influences EVINCI, the candidate conducted experiments on the 22 datasets by running the proposed method with only one objective at time. The candidate observed that the combination of the two objectives enhanced the performance of the proposed method, which obtained the best results in most datasets with a statistical significance of 95%, as determined after the application of the Nemenyi posthoc test.

In Chapter 5, the candidate presented a case study for the identification of defective pieces in the wagon inspection task, for which the candidate proposed the ILEC method. ILEC initially generates a pool of balanced and diversified samples, from which a set of CNNs are

induced. In this study, due to the high computational cost associated with the induction of CNNs, the candidate chose do not to perform optimization of the initial samples. Instead, the candidate proposed an ensemble pruning method based on non-dominance ranking, considering the two main requirements that produce efficient ensembles, i.e., accuracy and diversity of their base classifiers. The experimental results suggest that the proposed method has the best overall accuracy and the smaller standard deviation among all the tested methods, while, at the same time, has a high predictive accuracy for all classes, specifically for the class of interest.

5.9 Future Work

Several studies have shown that the overall imbalance rate of the dataset is not the main cause of the low predictive accuracy of classifiers in the minority classes, but intrinsic characteristics of the dataset that are aggravated by the class imbalance. Thus, some future work prospects are the further study of these characteristics and propose metrics to identify such deficiencies in the dataset. These metrics can be applied to optimize samplings in the dataset, as well as improve the methods proposed in this thesis.

The pruning method, developed for ILEC (Chapter 5), produced significant improvement in the predictive accuracy of the original CNN ensemble and has a relatively low computational cost. Thus, the analysis of adding this mechanism to the other methods proposed in this thesis could provide important research insights. A possible line of research in this direction would be to use dynamic-sized ensembles. That is, after each iteration, the pruning mechanism would be applied to the individuals selected to compose the ensemble. Although this solution increases the computational cost of the methods, it is possible to counterbalance this inconvenience by making the pruning process in parallel.

A major challenge in imbalanced learning research and application is when the dataset is unstable, the case in which the change of the label of instances can result in the appearance of new classes and disappearance of existing classes. The traditional methods of concept drift detection have difficulties in this type of problem, for example, the absence for a long period of new instances of the minority classes can be interpreted as a change of concept and trigger an unnecessary update in the classification system. In addition, this research area has other open points that need further research that can result in the proposal of modified or new approaches. For example, credit risk analysis models of financial institutions are generated from imbalanced dataset. These models may suffer deterioration in their accuracy due to external situations, such as changes in the economy may lead to "good" customers do not fulfill their obligations to the financial institution.

The diversity of base classifiers is essential for the construction of efficient ensembles. In the methods proposed in this thesis, multiobjective evolutionary algorithms are used to evolve samples from the dataset to induce base classifiers with high predictive accuracy in all classes of

the data set and high diversity among them. A new approach to be incorporated into the proposed methods would be to generate base classifiers specialized in a particular set of classes. Ensembles of specialized classifiers have the advantage of being able to use different sources or datasets for its induction, they still present some weaknesses that requiring more efficient solutions, such as an effective way to reach the consensus of the predictions (LI; LIN, 2017). Due to the scarcity of data sources that generate efficient rules for the minority classes, the use of methods from specialized classifier ensembles can bring significant advantages to imbalanced learning.

Deep Learning represents a major trend in the area of Machine Learning due to its wide variety of successful applications. However, neural networks are usually largely affected dataset imbalance, due to the scarcity of instances of the minority classes necessary to update the weights of the network. The use of evolutionary algorithms to optimize image sampling, as performed in the proposals presented in chapters 2, 3 and 4, presents a high computational cost. Thus, it is necessary to study the representativeness of instances in deep learning algorithms to reduce this cost.

5.10 Bibliography

LI, Z.; LIN, D. Integrating specialized classifiers based on continuous time markov chain. *CoRR*, abs/1709.02123, 2017. Available: <<http://arxiv.org/abs/1709.02123>>. Citation on page 130.

COMPARISON OF THE PROPOSED METHODS

In this appendix, we present a comparison between the methods we propose to deal with the problem of imbalanced learning. Namely, MOGASamp (Chapter 2), EMOSAIC (Chapter 3) and EVINCI (Chapter 4). The ILEC method, presented in Chapter 5, deals with a dataset of images that presents an imbalance in the distribution of images by class. For this reason, the ILEC method is not part of the experiments presented below.

A.1 Experimental Setup

All methods used in these experiments were configured to use the J48 classification algorithm as base classifiers. We used the J48 algorithm available in the RWeka package (HORNÍK; BUCHTA; ZEILEIS, 2009). We also set the mutation rate for the three methods as 0.1.

In the experiments, we utilized the same twenty-two datasets used in the article which presented the method EVINCI (Chapter 4). The datasets present different imbalance ratios and with the number of classes ranging from 2 to 18. All these datasets are summarized in Table 8 of the Chapter 4, including the number of classes (#C), number of features (#F), imbalance ratio (Imb Ratio), and class distribution. We report the results of 30 trials of stratified 5-fold cross-validation. We used the G-Mean and MAUC performance metrics to compare the methods. These metrics are described in Section 3.4.2 of Chapter 3.

The initial proposals of the MOGASamp and EMOSAIC methods suggest that the population size of the evolutionary algorithm should be 30 individuals and that the evolutionary process has a maximum limit of 30 generations. On the other hand, the proposal of EVINCI suggests that the method should be executed with a population of 10 individuals and with a maximum of 20 generations. For this reason, we perform experiments with these two configurations.

MOGASamp was initially projected to deal only with unbalanced binary datasets, which is why the AUC metric is one of the objectives of the evolutionary process of that method. To perform the experiments of the MOGASamp method on a multiclass dataset the AUC metric was replaced by its extension that estimates the performance of classifiers in multiclass datasets, i.e., MAUC, as described in Section 3.4.2 of Chapter 3. The MAUC metric calculates the unweighted mean of the area under the roc curve for each pair of classes, so in binary datasets, the MAUC metric will have the same value as the AUC.

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 (DEMŠAR, 2006). 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.

Datasets	MOGASamp	EMOSAIC	EVINCI
Abalone	0.0000	0.0060	0.0071
Balance-scale	0.4576	0.6622	0.4487
Car	0.8433	0.8498	0.8040
Chess	0.5447	0.5568	0.5284
Contraceptive	0.5176	0.5210	0.5114
Dermatology	0.9642	0.9506	0.9705
Dnormal	0.8797	0.8763	0.8685
Ecoli	0.7943	0.8052	0.7757
Ecoli2*	0.8623	0.8616	0.8517
Glass	0.6869	0.6577	0.6715
New-thyroid	0.9126	0.9239	0.8874
Nursery	0.9436	0.9075	0.9452
Oilspill*	0.7195	0.7952	0.7433
Page-blocks	0.9196	0.9276	0.9486
Penbased	0.9371	0.9369	0.9296
Poker*	0.5450	0.6170	0.4743
Satellite	0.8805	0.8814	0.8700
Shuttle	0.9979	0.9972	0.9971
Thyroid	0.9421	0.9436	0.9302
Winequality*	0.5379	0.7472	0.6133
Yeast	0.1170	0.2362	0.3095
Yeast5*	0.8875	0.9501	0.9432
G-Mean Average	0.7223	0.7550	0.7286
Ranking Count	5-11-6	12-7-3	5-4-13
Ranking Average	2.05	1.59	2.36

Table 12 – G-mean Achieved by the Proposed Methods in the Experiments over 30 Runs with Population Size Equals to 30 and the Maximum Limit of 30 Generations. Also presents G-mean Average for Each Method, Ranking Count, and Ranking Average

A.2 Experimental Results

The Tables 12 and 13 show the G-mean and MAUC averages values, respectively, obtained by each method for each dataset in the experiments with the population size setting of 30 and with a maximum number of 30 generations. The last three rows in each Table summarize the results comparing the three methods in all the datasets considered. The first row is the metric average (G-Mean or MAUC) obtained by the compared methods. The *Ranking Count* row shows the number of datasets in which each technique obtained the best performance metric value, the second best, and so on. For example, the three numbers 5-11-6 in the first column of Table 12 represent the ranking of the MOGASamp method. It obtained the highest G-Mean value on five of the 22 datasets, the second highest values on 11 datasets, and the third highest values on six datasets. The *Average Ranking* row shows the average of the ranking position obtained by each method on all datasets.

Datasets	MOGASamp	EMOSAIC	EVINCI
Abalone	0.7704	0.7721	0.7756
Balance-scale	0.8321	0.8397	0.8256
Car	0.9717	0.9734	0.9565
Chess	0.9483	0.9479	0.9219
Contraceptive	0.7079	0.7125	0.7129
Dermatology	0.9957	0.9974	0.9964
Dnormal	0.9692	0.9699	0.9632
Ecoli	0.9569	0.9595	0.9550
Ecoli2	0.9264	0.9312	0.9206
Glass	0.8956	0.8844	0.8832
New-Thyroid	0.9927	0.9911	0.9903
Nursery	0.9923	0.9876	0.9910
Oilspill	0.9205	0.9095	0.8914
Page-blocks	0.9887	0.9875	0.9879
Penbased	0.9945	0.9944	0.9944
Poker	0.8075	0.7214	0.7386
Satellite	0.9871	0.9870	0.9862
Shuttle	1.0000	1.0000	0.9996
Thyroid	0.9935	0.9932	0.9930
Winequality	0.8092	0.8329	0.8313
Yeast	0.8520	0.8707	0.8522
Yeast5	0.9729	0.9829	0.9782
MAUC Average	0.9220	0.9203	0.9157
Ranking Count	11-5-6	10-9-3	2-8-12
Ranking Average	1.77	1.68	2.45

Table 13 – MAuc Achieved by the Proposed Methods in the Experiments over 30 Runs with Population Size Equals to 30 and Maximum Limit of 30 Generations. Also presents MAuc Average for Each Method, Ranking Count, and Ranking Average

Using this configuration (population size = 30 and maximum generations = 30) and the metric G-Mean, the Table 12 presents the EMOSAIC method as the one that obtained the best G-Mean mean, the largest number of wins and the lowest ranking average, which suggests that EMOSAIC achieved the best overall performance. The ranking provided by the Friedman test

Datasets	MOGASamp	EMOSAIC	EVINCI
Abalone	0.0000	0.0000	0.0000
Balance-scale	0.5205	0.6372	0.4517
Car	0.8183	0.8430	0.7950
Chess	0.4989	0.4916	0.5325
Contraceptive	0.5138	0.5176	0.5111
Dermatology	0.9664	0.9533	0.9643
Dnormal	0.8721	0.8755	0.8630
Ecoli	0.7751	0.7963	0.7694
Ecoli2	0.8555	0.8713	0.8522
Glass	0.6625	0.6459	0.6594
New-thyroid	0.9128	0.9168	0.8890
Nursery	0.9313	0.9116	0.9416
Oilspill	0.7453	0.7834	0.7454
Page-blocks	0.9076	0.9203	0.9446
Penbased	0.9246	0.9260	0.9261
Poker	0.5275	0.5861	0.5251
Satellite	0.8701	0.8719	0.8623
Shuttle	0.9980	0.9977	0.9975
Thyroid	0.9449	0.9452	0.9339
Winequality	0.5717	0.6461	0.6266
Yeast	0.1120	0.2914	0.4933
Yeast5	0.9150	0.9406	0.9420
G-Mean Average	0.7202	0.7440	0.7375
Ranking Count	3-13-6	12-6-4	6-5-11
Ranking Average	2.14	1.64	2.23

Table 14 – G-mean Achieved by the Proposed Methods in the Experiments over 30 Runs with Population Size Equals to 10 and Maximum Limit of 20 Generations. Also presents G-mean Average for Each Method, Ranking Count, and Ranking Average

supports this assumption, showing EMOSAIC to be the best-ranked method. The Friedman test also rejects the null-hypothesis, i.e., it says that there is a statistically significant difference between the algorithms ($p\text{-value} = 0.0322$). Hence, we executed the Nemenyi post-hoc test for pairwise comparison. The results of this test show that EMOSAIC outperforms EVINCI with a statistical significance at the 95% confidence level. However, there are no significant statistical differences between EMOSAIC and MOGASamp neither between MOGASamp and EVINCI.

A similar but not identical situation occurs when we analyze from the perspective of the MAUC metric, that is, Table 13. The EMOSAIC method obtained the lowest average ranking among the analyzed methods, the second-best average MAUC and the second highest number of victories among the methods. However, it is important to note that although the MOGASamp method obtained the highest number of victories (11), EMOSAIC obtained almost twice as many second places as MOGASamp (9) and half of the third places (3). This suggests that EMOSAIC performs well (first or second place) in most of the data set used in the experiments. The EVINCI method presented the worst results, ranking third in most datasets. The Friedman test says that there is a statistically significant difference between the algorithms ($p\text{-value} = 0.01899$) and shows EMOSAIC as the best-ranked method by MAUC metric. The Nemenyi post-hoc test for pairwise comparison shows that EVINCI is outperformed by MOGASamp with statistical

Datasets	MOGASamp	EMOSAIC	EVINCI
Abalone	0.7434	0.7431	0.7467
Balance-scale	0.8203	0.8292	0.8167
Car	0.9623	0.9682	0.9532
Chess	0.9283	0.9297	0.9220
Contraceptive	0.7036	0.7083	0.7076
Dermatology	0.9948	0.9963	0.9949
Dnormal	0.9655	0.9667	0.9587
Ecoli	0.9531	0.9533	0.9506
Ecoli2	0.9148	0.9327	0.9083
Glass	0.8738	0.8711	0.8712
New-thyroid	0.9913	0.9851	0.9854
Nursery	0.9900	0.9861	0.9904
Oilspill	0.9037	0.8855	0.8716
Page-blocks	0.9872	0.9857	0.9860
Penbased	0.9917	0.9916	0.9928
Poker	0.7613	0.6957	0.7172
Satellite	0.9832	0.9836	0.9827
Shuttle	0.9999	0.9999	0.9996
Thyroid	0.9908	0.9891	0.9897
Winequality	0.7905	0.8010	0.8124
Yeast	0.8461	0.8547	0.8444
Yeast5	0.9737	0.9760	0.9749
MAUC Average	0.9122	0.9106	0.9080
Ranking Count	7-11-4	12-2-8	4-8-10
Ranking Average	1.86	1.82	2.27

Table 15 – MAuc Achieved by the Proposed Methods in the Experiments over 30 Runs with Population Size Equals to 10 and Maximum Limit of 20 Generations. Also presents MAuc Average for Each Method, Ranking Count, and Ranking Average

significance at the 90% confidence level and by the EMOSAIC at the 95% confidence level.

The Tables 14 and 15 show the G-mean and MAUC averages values, respectively, for the configuration using population size equal to 10 and the maximum number of generations equal to 20. Like the previous tables, these also present in the last three lines, the average of the metric used, the count of the rankings and the ranking average.

As in the Tables that present the results of experiments with the configuration that evolves a population of 30 individuals, Tables 14 and 15 also suggest that EMOSAIC is the method that obtained better performance even when it evolves a population of only ten individuals. On the G-Mean metric perspective, EMOSAIC obtained the highest G-Mean average, the highest number of wins and the lowest average ranking. A similar situation can also be seen in Table 15, with a small difference in the MAUC average, where EMOSAIC loses to MOGASamp by a little difference (0.0017). However, the Friedman test indicates that there is no statistical difference between the methods.

A.3 Bibliography

DEMsAR, J. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.*, JMLR.org, v. 7, p. 1–30, Dec. 2006. ISSN 1532-4435. Available: <http://dl.acm.org/citation.cfm?id=1248547.1248548>. Citations on pages [54](#), [76](#), [77](#), and [132](#).

HORNIK, K.; BUCHTA, C.; ZEILEIS, A. Open-source machine learning: R meets Weka. *Computational Statistics*, v. 24, n. 2, p. 225–232, 2009. Citations on pages [104](#) and [131](#).