A Survey of Predictive Modeling on Imbalanced Domains

PAULA BRANCO and LUÍS TORGO and RITA P. RIBEIRO, LIAAD - INESC TEC, DCC - Faculty of Sciences, University of Porto, Porto, Portugal

Many real world data mining applications involve obtaining predictive models using data sets with strongly imbalanced distributions of the target variable. Frequently, the least common values of this target variable are associated with events that are highly relevant for end users (e.g. fraud detection, unusual returns on stock markets, anticipation of catastrophes, etc.). Moreover, the events may have different costs and benefits, which when associated with the rarity of some of them on the available training data creates serious problems to predictive modeling techniques. This paper presents a survey of existing techniques for handling these important applications of predictive analytics. Although most of the existing work addresses classification tasks (nominal target variables), we also describe methods designed to handle similar problems within regression tasks (numeric target variables). In this survey we discuss the main challenges raised by imbalanced domains, propose a definition of the problem, describe the main approaches to these tasks, propose a taxonomy of the methods, summarize the conclusions of existing comparative studies as well as some theoretical analyses of some methods and refer to some related problems within predictive modeling.

CCS Concepts: •**Computing methodologies** → **Cost-sensitive learning;** *Supervised learning;*

Additional Key Words and Phrases: Imbalanced Domains, Rare Cases, Classification, Regression, Performance Metrics

ACM Reference Format:
Paula Branco, Luís Torgo and Rita P. Ribeiro, 2016. A Survey of Predictive Modeling on Imbalanced Domains. *ACM Comput. Surv.* 1, 1, Article 1 (January 1), 56 pages.
DOI: 0000001.0000001

1. INTRODUCTION

Predictive modeling is a data analysis task whose goal is to build a model of an unknown function $Y = f(X_1, X_2, \cdots, X_p)$, based on a training sample $\{\langle \mathbf{x}_i, y_i \rangle\}_{i=1}^{n}$ with examples of this function. Depending on the type of the variable Y, we face either a classification task (nominal Y) or a regression task (numeric Y). Models are obtained through a search process guided by the optimization of some criterion. The most frequent criteria are the error rate for classification and the mean squared error for regression. For some real world applications it is of key importance that the obtained

This work is financed by the ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme within project ≪POCI-01-0145-FEDER-006961≫ and by the North Portugal Regional Operational Programme (ON.2 – O Novo Norte), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF), and by national funds, through the Portuguese funding agency, Fundação para a Ciência e a Tecnologia (FCT) within ≪Project NORTE-07-0124-FEDER-000059≫. Paula Branco was supported by a scholarship from the Fundação para a Ciência e Tecnologia (FCT), Portugal (scholarship number PD/BD/105788/2014). Part of the work of Luís Torgo was supported by a sabbatical scholarship (SFRH/BSAB/113896/2015) from the Fundação para a Ciência e Tecnologia (FCT).
Author's addresses: P. Branco, L. Torgo and R. Ribeiro, LIAAD – INESC TEC, Campus da FEUP, Rua Dr. Roberto Frias, 4200 - 465 Porto, Portugal; DCC – Faculty of Sciences, University of Porto, Rua do Campo Alegre, s/n, 4169 - 007 Porto, Portugal; email: paula.branco@dcc.fc.up.pt, ltorgo@dcc.fc.up.pt and rpribeiro@dcc.fc.up.pt.

DOI: 0000001.0000001

models are particularly accurate at some sub-range of the domain of the target variable. Examples include diagnosis of rare diseases, forecasting rare extreme returns in financial markets, among many others. Frequently, these specific sub-ranges of the target variable are poorly represented in the available training sample. In these cases, we face what is usually known as a problem of imbalanced domains, or imbalanced data sets. Informally, in these domains the cases that are more important for the user are rare and few exist on the available training set. The combination of the specific preferences of the user with the poor representation of these situations creates problems at several levels. Namely, we typically need (i) special purpose evaluation metrics that are biased towards the performance of the models on these rare cases, and moreover, we need means for (ii) making the learning algorithms focus on these rare events. Without addressing these two questions, models will tend to be biased to the most frequent (and uninteresting for the user) cases, and the results of the "standard" evaluation metrics will not capture the competence of the models on these rare cases.

The main contributions of this work are: i) provide a general definition of the problem of imbalanced domains suitable for classification and regression tasks; ii) review the main performance assessment measures for classification and regression tasks under imbalanced domains; iii) propose a taxonomy of existing approaches to tackle the problem of imbalanced domains both for classification and regression tasks; iv) describe the most important techniques to address this problem; (v) summarize the conclusions of some existing experimental comparisons; and (vi) review some theoretical analyses of specific methods. Existing surveys address only the problem of imbalanced domains for classification tasks (e.g. Kotsiantis et al. [2006]; He and Garcia [2009]; Sun et al. [2009]). Therefore, the coverage of performance assessment measures and approaches to tackle both classification and regression tasks is an innovative aspect of our paper. Another key feature of our work is the proposal of a broader taxonomy of methods for handling imbalanced domains. Our proposal extends previous taxonomies by including post-processing strategies. Finally, the paper also includes a summary of the main conclusions of existing experimental comparisons of approaches to these tasks as well as references to some theoretical analyses of specific techniques.

The paper is organized as follows. Section 2 defines the problem of imbalanced domains and the type of existing approaches to address this problem. Section 3 describes several evaluation metrics that are biased towards performance assessment on the relevant cases in these domains. Section 4 provides a taxonomy of the approaches to imbalanced domains, describing some of the most important techniques in each category. In Section 5 we present some general conclusions of existing experimental comparisons of different methods. Section 6 describes the main theoretical contributions for understanding the problem of imbalanced domains. Finally, Section 7 explores some problems related with imbalanced domains and Section 8 concludes the paper also including a summary of recent trends and open research questions.

2. PROBLEM DEFINITION

As we have mentioned before the problem of imbalanced domains occurs in the context of predictive tasks where the goal is to obtain a good approximation of the unknown function $Y = f(X_1, X_2, \cdots, X_p)$ that maps the values of a set of p predictor variables into the values of a target variable. This approximation, $h(X_1, X_2, \cdots, X_p)$, is obtained using a training data set $D = \{\langle \mathbf{x}_i, y_i \rangle\}_{i=1}^{n}$.

The problem of imbalanced domains can be informally described by the following two assertions:

(1) the user assigns more importance to the predictive performance of the obtained approximation $h(X_1, X_2, \cdots, X_p)$ on a subset of the target variable domain;

(2) the cases that are more relevant for the user are poorly represented in the training set, up to the point of leading to bad estimates of their conditional density by the models.

The non-uniform importance mentioned in assertion (1) can occur in different forms, namely: (i) by assigning different benefits to accurate predictions of the values of the target variable; (ii) by having different costs associated with different types of prediction errors; (iii) or by a mixture of both situations. This means that there is a strong relationship between imbalanced problems and cost-sensitive learning (e.g. Elkan [2001]). Both result from these non-uniform preference biases of the user. However, a cost sensitive problem may not be imbalanced if the cases that are more relevant are sufficiently represented in the training data, i.e. if assertion (2) is not true. This means that an imbalanced problem always involves non-uniform costs/benefits, but the opposite is not always true.

The quality of the information we have concerning the user domain preferences (item (1) in the above list) is also of key importance as it can have an impact on: (i) the way we evaluate and/or compare alternative models; and (ii) the process used to influence the learning process in order to obtain models that are "optimal" according to these user preferences. This was termed by Weiss [2013] as the "problem-definition issue". In one extreme the user may be able to provide information of the full utility function, $u(\hat{y}, y)$, that determines the value for the user of predicting $\hat{y}$ for a true value of y. According to Elkan [2001] this should be a positive value for accurate predictions (a benefit) and a negative value for prediction errors (a cost). Having the full specification of this function is the ideal setting. Unfortunately, this information is frequently difficult to obtain in real world applications, particularly for regression tasks where the target variable has an infinite domain. A slightly less challenging task for the user is to provide a simpler function that assigns a relevance score to each value of the target variable domain. We will call this the relevance function, $\phi()$, which is a function that maps the values of the target variable into a range of importance, where 1 corresponds to maximal importance and 0 to minimum relevance,

$$\phi(Y) : \mathcal{Y} \rightarrow [0, 1] \tag{1}$$

where $\mathcal{Y}$ is the domain of the target variable Y. This is an easier function to be defined by the user because, among other aspects, it only depends on one variable (y), while the utility function depends on two variables ($\hat{y}$ and y). Moreover, the definition of a utility function requires that a non-negligible amount of domain information is available whereas for the relevance function less information is needed. In effect, the utility of predicting a value $\hat{y}$ for a true value of y depends on both the relevance of each of these values but also on the associated loss [Torgo and Ribeiro 2007; Ribeiro 2011], i.e.

$$u(\hat{y}, y) = g(\phi(\hat{y}), \phi(y), L(\hat{y}, y)) \tag{2}$$

where $L(\hat{y}, y)$ is typically the 0/1 loss for classification tasks or the squared error for regression.

Finally, there are also applications where the available information is very informal, e.g. "the class c is the more relevant for me". This type of problem definition creates serious limitations both in terms of procedures to evaluate the models, but also in terms how to proceed to learn a model that takes this into consideration.

Let us assume the user has defined the function $\phi()$ that represents the importance assigned to the target variable domain and has also defined a threshold t_R which sets the boundary above which the target variable values are relevant. It is important to highlight that this threshold is not used for declaring a class or range of values irrele-

vant. It is used for understanding which target values the user considers normal and which are the most relevant ones. Using this threshold we can split the domain of the target variable in two complementary subsets, $\mathcal{Y}_R \subset \mathcal{Y} = \{y \in \mathcal{Y} : \phi(y) > t_R\}$ and $\mathcal{Y}_N = \mathcal{Y} \setminus \mathcal{Y}_R$. In this context, D_R is the subset of the training samples D where $y \in \mathcal{Y}_R$ and D_N is the subset of the training sample with the normal (or less important) cases, i.e $D_N = D \setminus D_R$.

Using the above notation we can provide a more formal definition of required conditions for a predictive task to be considered an imbalanced problem:

(1) The non-uniform importance of the predictive performance of the models across the domain of the target variable can result from:
   (a) $L(y, y) = L(x, x) \implies u(y, y) = u(x, x)$, i.e. accurate predictions may have different benefits;
   (b) $L(y_1, y_2) = L(x_1, x_2) \implies u(y_1, y_2) = u(x_1, x_2)$, i.e. the cost of similar errors is not uniform;
   (c) a mixture of both situations

(2) $|D_R| << |D_N|$, i.e. relevant values are poorly represented in the training set.

As we have mentioned, the problem of imbalanced domains is associated with a mismatch between the importance assigned by the user to some predictions (1) and the representativeness of the values involved in these predictions on the available training sample (2). Still, it is important to stress that among the possible mismatches between these two factors, only one type really leads to the so-called problem of imbalanced domains. In effect, only when the more important cases are poorly represented in the available data we have a problem. It is this lack of representativeness that causes: (i) the "failure" of standard evaluation metrics as they are biased towards average performance and will not correctly assess the performance of the models on these rare events; (ii) the learning techniques to disregard these rare events due to their small impact on the standard evaluation metrics that usually guide their learning process or due to their lack of statistical significance. Other types of mismatch do not have these consequences. If the user has a non-uniform preference bias but the data distribution is balanced, then the second consequence does not occur as the important cases are sufficiently represented in the data, while the first consequence is not so serious because the important cases are not rare and thus will have an impact on the standard performance metrics[1]. Moreover, if the user has a uniform preference over the different types of predictions, then even if the data distribution is imbalanced this is not a problem given the indifference of the user to where the errors occur.

Regarding the failure of traditional evaluation metrics several solutions have been proposed to address this problem and overcome existing difficulties, mainly for classification tasks. We will review these proposals in Section 3.

With respect to the inadequacy of the obtained models a large number of solutions has also appeared in the literature. We propose a categorization of these approaches that considers four types of strategies: (i) modifications on the learning algorithms, (ii) changes on the data before the learning process takes place, (iii) transformations applied to the predictions of the learned models and finally (iv) hybrid strategies that combine different types of strategies. These solutions will be reviewed in Section 4.

We will now illustrate the problem of imbalanced domains with two concrete examples: one in classification and another in regression.

For imbalanced classification we use the *Glass* data set from the UCI ML repository. This data set contains 213 examples, and the target variable (TYPE) includes 6 different classes (1,2,3,5,6,7). Figure 1 displays the bar chart with the frequencies of

[1]Though potentially not as exacerbated as one could wish.

the classes. We have chosen this particular data set to highlight that the problem of imbalanced domains is very relevant and challenging in the multiclass case. For illustration purposes, let us assume that the lowest the class frequency, the highest the relevance for the users of this application. The figure also shows the relevance scores (ϕ) of the classes, which were computed from the frequency of each class. Suppose the user informs us that any class value with a relevance higher than 0.5 is important. This would mean that examples of classes 3, 5 and 6 are important for the user, and the examples from the remaining classes are not so relevant. The number of relevant cases ($|D_R|$) would be 39, while the number of irrelevant cases ($|D_N|$) would be the remaining 174 cases. This means that the more relevant cases are not very well represented in the training sample D. Applying a standard classification algorithm to such data set would lead to models that would have unreliable estimates of the conditional probability of the classes 3, 5 and 6, as they are very poorly represented in the available data. This would not be a problem if those were not the classes that are more important to the user. Moreover, using a standard evaluation metric (e.g. error rate) to compare alternative models for this data set, could eventually lead the user to select a model that is not the best performing model on the classes that are more relevant.

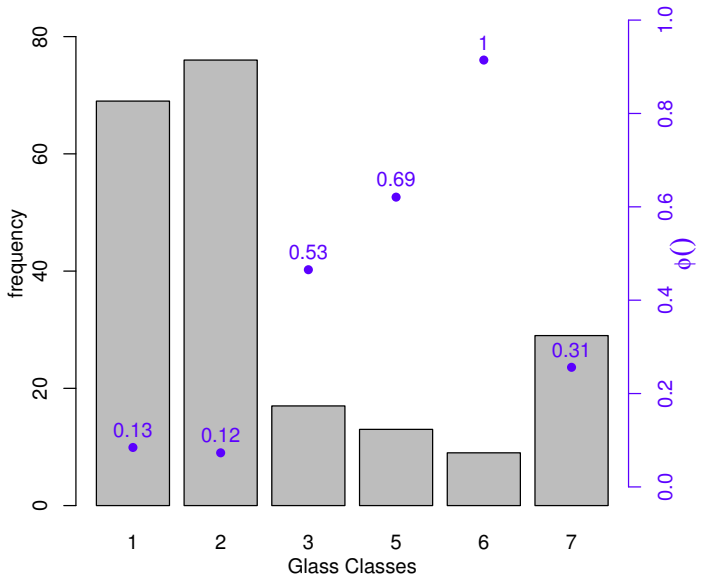

Fig. 1. Distribution of classes in *glass* data set (bars) and relevance of each class (blue) inversely proportional to the classes frequencies.

As an example of a regression task, we selected the *Forest Fires* data set[2]. This data set includes 2831 examples. Figure 2 shows the distribution of the data set target vari-

[2]Available in the UBA R package http://www.dcc.fc.up.pt/~rpribeiro/uba/.

able[3], the relevance function $\phi()$ automatically determined (using a method proposed in Ribeiro [2011] for cases where high relevance is associated with low frequency) and a boxplot of the examples target variable distribution. If we use again a relevance threshold of 0.5 we would have $|D_R| = 489$ and $|D_N| = 2342$. Once again, standard regression algorithm would have difficulties in performing well on the rare extreme high values of the target, because of their rarity in the training set. Again, this would be a problem given the established preference bias for this application, i.e. be accurate at the prediction of the biggest forest fires.

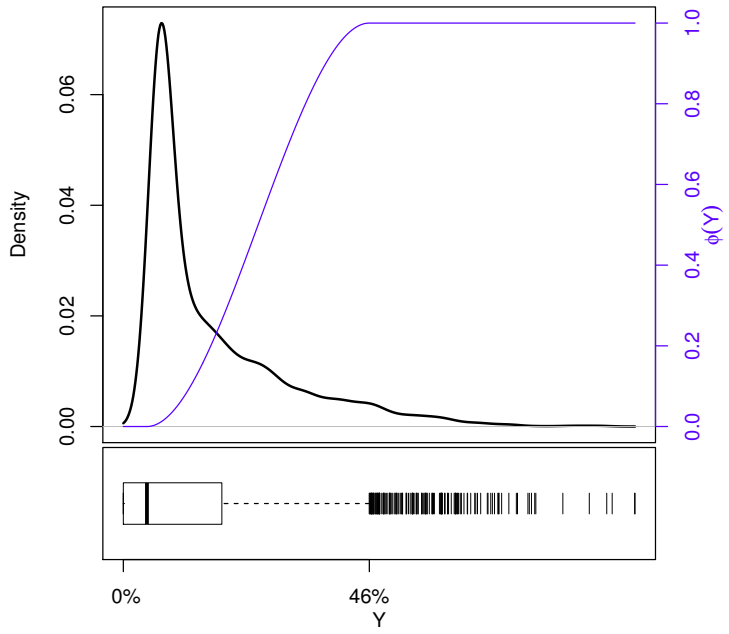

Fig. 2. Distribution of the burnt area in *forest fires* data set (black), relevance function automatically estimated (blue) and boxplot of the examples distribution.

3. PERFORMANCE METRICS FOR IMBALANCED DOMAINS

This section describes existing approaches for performance assessment on imbalanced problems. This is the most studied aspect of predictive modeling for these tasks. Nevertheless, issues such as the error estimation procedure and the statistical tests used on imbalanced domains are also extremely important and have been, so far, largely neglected. These issues present challenges when considering imbalanced domains and much research is still needed [Japkowicz 2013].

Obtaining a model from data can be seen as a search problem guided by an evaluation criterion that establishes a preference ordering among different alternatives.

[3]Approximated through a kernel density estimator.

The main problem with imbalanced domains is the user preference towards the performance on cases that are poorly represented in the available data sample. Standard evaluation criteria tend to focus the evaluation of the models on the most frequent cases, which is against the user preferences on these tasks. In fact, the use of traditional metrics in imbalanced domains can lead to sub-optimal classification models [He and Garcia 2009; Weiss 2004; Kubat and Matwin 1997] and may produce misleading conclusions since these measures are insensitive to skewed domains [Ranawana and Palade 2006; Daskalaki et al. 2006]. As such, selecting proper evaluation metrics plays a key role in the task of correctly handling data imbalance. Adequate metrics should not only provide means to compare the models according to the user preferences, but can also be used to drive the learning of these models.

As we have mentioned, there are several ways of expressing the user preference biases. In case we have the highest quality information, in the form of an utility function $u(\hat{y}, y)$, the best way to evaluate the learned models would be by the total utility of its predictions, given by

$$U = \sum_{i=1}^{n_{test}} u(\hat{y}_i, y_i) \tag{3}$$

When the full information on the operating context is not available we have to resort to other evaluation metrics. In this section, we provide an exhaustive description of most of the metrics that have been used in the context of imbalanced domains problems.

We have organized the performance assessment measures into scalar (numeric) and graphical-based (graphical or scalar based in graphical information) metrics. Scalar metrics present the results in a more succinct way (a single number reflects the performance of the learner) but also have drawbacks. If the user knows the deployment setting of the learned model, then scalar metrics may be adequate. However, if the deployment context is not known in advance, then the graphical-based metrics may be more useful [Japkowicz 2013]. Graphical-based measures allow the visualization or synthesis of the performance of an algorithm across all operating conditions. We must also emphasize that using different evaluation metrics may lead to different conclusions (e.g. Van Hulse et al. [2007]) which is problematic and reinforces the need for finding suitable metrics that are capable of assessing correctly the user goals.

Table I summarizes the main references concerning performance assessment proposals for imbalanced domains in classification and regression.

Table I. Metrics for classification and regression, corresponding sections and main bibliographic references

Task type (Section)	Main References
Classification (3.1)	Bradley [1997]; Kubat et al. [1998]; Provost et al. [1998]; Drummond and Holte [2000]; Estabrooks and Japkowicz [2001]; Ferri et al. [2005]; Davis and Goadrich [2006]; Ranawana and Palade [2006]; Cohen et al. [2006]; Wu et al. [2007]; Weng and Poon [2008]; García et al. [2008]; Batuwita and Palade [2009]; García et al. [2009, 2010]; Hand [2009]; Ferri et al. [2009]; Sokolova and Lapalme [2009]; Thai-Nghe et al. [2011]; Ferri et al. [2011a]; Batuwita and Palade [2012]
Regression (3.2)	Zellner [1986]; Cain and Janssen [1995]; Christoffersen and Diebold [1997]; Bi and Bennett [2003]; Crone et al. [2005]; Torgo [2005]; Torgo and Ribeiro [2007]; Lee [2008]; Torgo and Ribeiro [2009]; Ribeiro [2011]; Hernández-Orallo [2013]; Branco [2014]

3.1. Metrics for Classification Tasks

Let us start with some notation. Consider a test set with n examples each belonging to one of $c \in \mathbb{C}$ different classes. For each test case, $\mathbf{x}_i$, with a true target variable value $y_i = f(\mathbf{x}_i)$, a classifier outputs a predicted class, $\hat{y}_i = h(\mathbf{x}_i)$. This predicted class is typically the class with highest estimated conditional probability, $\hat{y}_i = \text{argmax}_y \hat{P}(Y = y \,|\, X = \mathbf{x}_i)$, but other decision thresholds (or decision rules, mostly for multiclass tasks) can be used[4]. Let $I()$ be an indicator function that returns 1 if its argument is true and 0 otherwise. Let $n_c = \sum_{i=1}^{n} I(y_i = c)$ represent the total number of examples that belongs to class c. The prior probability of class c can be estimated as $p(Y = c) = \frac{n_c}{n}$. The estimated conditional probability of example $\mathbf{x}_i$ belonging to class c is given by $\hat{P}(Y = c \,|\, X = \mathbf{x}_i)$, or in a simplified way $\hat{P}(c \,|\, \mathbf{x}_i)$.

3.1.1. Scalar Metrics.

Two-Class Problems.

Consider a binary classification task with a negative ($Y = -$) and a positive class ($Y = +$). The confusion matrix for a two-class problem presents the results obtained by a given classifier (cf. Table II). This table provides for each class the instances that were correctly classified, i.e. the number of True Positives (TP) and True Negatives (TN), and the instances that were wrongly classified, i.e. the number of False Positives (FP) and False Negatives (FN).

Table II. Confusion matrix for a two-class problem.

		Predicted		**Total**
		Positive ($Y = +$)	Negative ($Y = -$)	
True	Positive ($Y = +$)	TP= $\sum_{i=1}^{n} I(y_i = +)I(\hat{y}_i = +)$	FN = $n_+ - TP$	$n_+ = \sum_{i=1}^{n} I(y_i = +)$
	Negative ($Y = -$)	FP = $n_- - TN$	TN= $\sum_{i=1}^{n} I(y_i = -)I(\hat{y}_i = -)$	$n_- = \sum_{i=1}^{n} I(y_i = -)$
Total		$\sum_{i=1}^{n} I(\hat{y}_i = +)$	$\sum_{i=1}^{n} I(\hat{y}_i = -)$	n

Accuracy (cf. Equation 4) and its complement *error rate* are the most frequently used metrics for estimating the performance of learning systems in classification problems. For two-class problems, *accuracy* can be defined as follows,

$$accuracy = \frac{TP + TN}{TP + FN + TN + FP} \tag{4}$$

Considering a user preference bias towards the minority (positive) class examples, *accuracy* is not suitable because the impact of the least represented, but more important, examples is reduced when compared to that of the majority class. For instance, if we consider a problem where only 1% of the examples belong to the minority class, a high *accuracy* of 99% is achievable by predicting the majority class for all examples. Yet, all minority class examples, the rare and more interesting cases for the user, are misclassified. This is worthless when the goal is the identification of the rare cases.

[4]For crisp classifiers we can assume that the probability is 1 for the predicted class and 0 for the remaining classes.

The metrics used in imbalanced domains must consider the user preferences and, thus, should take into account the data distribution. To fulfill this goal several performance measures were proposed. From Table II the following measures (cf. Equations 5-10) can be obtained,

$$true\ positive\ rate\ (recall\ or\ sensitivity):\ TP_{rate} = \frac{TP}{TP+FN} \tag{5}$$

$$true\ negative\ rate\ (specificity\ ):\ TN_{rate} = \frac{TN}{TN+FP} \tag{6}$$

$$false\ positive\ rate:\ FP_{rate} = \frac{FP}{TN+FP} \tag{7}$$

$$false\ negative\ rate:\ FN_{rate} = \frac{FN}{TP+FN} \tag{8}$$

$$positive\ predictive\ value\ (precision\ ):\ PP_{value} = \frac{TP}{TP+FP} \tag{9}$$

$$negative\ predictive\ value:\ NP_{value} = \frac{TN}{TN+FN} \tag{10}$$

However, as some of these measures exhibit a trade-off and it is impractical to simultaneously monitor several measures, new metrics have been developed, such as the F_β [Rijsbergen 1979], the *geometric mean* [Kubat et al. 1998] or the *receiver operating characteristic* (*ROC*) curve [Egan 1975].

The F_β is defined as a combination of both *precision* and *recall*, as follows:

$$F_\beta = \frac{(1+\beta)^2 \cdot recall \cdot precision}{\beta^2 \cdot precision + recall} \tag{11}$$

where β is a coefficient set by the user to adjust the relative importance of *recall* with respect to *precision* (if $\beta = 1$ *precision* and *recall* have the same weight, large values of β will increase the weight of *recall* whilst values less than 1 will give more importance to *precision*). The majority of the papers that use F_β for performance evaluation under imbalanced domains adopt $\beta = 1$, which corresponds to giving the same importance to *precision* and *recall*.

The F_β is commonly used and is more informative than accuracy about the effectiveness of a classifier on predicting correctly the cases that matter to the user (e.g. Estabrooks and Japkowicz [2001]). This metric value is high when both the *recall* (a measure of completeness) and the *precision* (a measure of exactness) are high.

An also frequently used metric when dealing with imbalanced data sets is the *geometric mean* (*G-Mean*) which is defined as:

$$\textbf{\textit{G-Mean}} = \sqrt{\frac{TP}{TP+FN} \times \frac{TN}{TN+FP}} = \sqrt{sensitivity \times specificity} \tag{12}$$

G-Mean is an interesting measure because it computes the *geometric mean* of the accuracies of the two classes, attempting to maximize them while obtaining good balance. This measure was developed specifically for assessing the performance under imbalanced domains. However, with this formulation equal importance is given to both classes. In order to focus the metric only on the positive class, a new version of *G-Mean* was proposed. In this new formulation, *specificity* is replaced by *precision*.

Several other measures were proposed for dealing with some particular disadvantages of the previously mentioned metrics. For instance, a metric called *dominance* [García et al. 2008] (cf. Equation 13) was proposed to deal with the inability of *G-Mean* to explain how each class contributes to the overall performance.

$$dominance = TP_{rate} - TN_{rate} \tag{13}$$

This measure ranges from -1 to $+1$. A value of $+1$ represents situations where perfect *accuracy* is achieved on the minority (positive) class, but all cases of the majority class are missed. A value of -1 corresponds to the opposite situation.

Another example is the *index of balanced accuracy* (*IBA*) [García et al. 2009, 2010] (cf. Equation 14) which quantifies a trade-off between an index of how balanced both class accuracies are and a chosen unbiased measure of overall *accuracy*.

$$IBA_{\alpha}(M) = (1 + \alpha \cdot dominance)M \tag{14}$$

where $(1 + \alpha \cdot dominance)$ is the weighting factor and M represents any performance metric. $IBA_{\alpha}(M)$ depends on two user-defined parameters: M and α. The first one, M, is an assessment measure previously selected by the user, and the second one, α, will give more or less importance to $dominance$.

Another interesting metric, named mean class-weighted accuracy (*CWA*), was proposed by Cohen et al. [2006]. This metric tries to overcome the limitation of F_{β} of not taking into account the performance on the negative class. At the same time, it also tries to deal with the drawback of *G-Mean* which does not allow to give more importance to the minority class. *CWA* metric (cf. Equation 15) tries to deal with both problems by providing a mechanism for the user to define the weights to be used.

$$CWA = w \cdot sensitivity + (1 - w) \cdot specificity \tag{15}$$

with $0 \leq w \leq 1$ as the user-defined weight of the positive class.

Other metrics created with similar objectives include *optimized precision* [Ranawana and Palade 2006], *adjusted geometric mean* [Batuwita and Palade 2009, 2012] or *B42* [Thai-Nghe et al. 2011].

Multi-class Problems.

Although most metrics were proposed for handling two-class imbalanced tasks, some proposals also exist for the multi-class case.

Accuracy is among the metrics that were extended for multi-class problems. Equation 16 presents the definition of *accuracy* for multi-class tasks as an average of the accuracy of each class. However, for the reasons that we have already mentioned, this is not an appropriate choice for imbalanced domains.

$$accuracy = \frac{\sum_{i=1}^{n} I(y_i = \hat{y}_i)}{n} \tag{16}$$

The extension to multi-class of the *precision* and *recall* concepts is not an easy task. Several ways of accomplishing this were proposed in the literature. If we focus on a single class c, Equations 17 and 18 provide the *recall* and *precision* for that class, respectively. Equation 19 represents the corresponding F_{β} score.

$$recall(c) = \sum_{i=1}^{n} \frac{I(y_i = c)I(\hat{y}_i = c)}{n_c} \tag{17}$$

$$precision(c) = \frac{\sum_{i=1}^{n} I(y_i = c) I(\hat{y}_i = c)}{\sum_{i=1}^{n} I(\hat{y}_i = c)} \tag{18}$$

$$F_\beta(c) = \frac{(1+\beta)^2 \cdot recall(c) \cdot precision(c)}{\beta^2 \cdot precision(c) + recall(c)} \tag{19}$$

However, using $recall(c)$ and $precision(c)$ in multi-class problems is not a practical solution. If we consider a problem with 5 classes we would obtain 10 different scores (a *precision* and a *recall* value for each class). In this case, it is not easy to compare the performance of different classifiers. In order to obtain a single aggregated value for *precision* or *recall* in a certain test set, two main strategies can be used: micro or macro averaging which we will represent through the use of indexes μ and M, respectively. Equations 20 to 22 provide the definitions of *precision* and *recall* considering both micro (μ) and macro (M) averaging strategies.

$$Rec_\mu = Prec_\mu = \frac{\sum_{i=1}^{n} I(y_i = \hat{y}_i)}{n} \tag{20}$$

$$Rec_M = \frac{\sum_{c \in \mathbb{C}} recall(c)}{|\mathbb{C}|} \tag{21}$$

$$Prec_M = \frac{\sum_{c \in \mathbb{C}} precision(c)}{|\mathbb{C}|} \tag{22}$$

We must highlight that macro averaging measures assign an equal weight to all existing classes, while for micro averaging based metrics more importance is assigned to classes with higher frequencies. Therefore, micro averaging measures are usually considered unsuitable for imbalanced domains because of the mismatch between the examples distribution and the relevance $\phi()$ assigned by the user.

Regarding the F_β measure, several different proposals were made to provide an extension for multi-class problems. Equation 23, proposed by Ferri et al. [2009], averages the F_β values obtained for each class.

$$MF_\beta = \frac{\sum_{c \in \mathbb{C}} F_\beta(c)}{|\mathbb{C}|} \tag{23}$$

Two other proposals regarding an extension of F_β to multi-class tasks exist: one using the micro averaged values of *recall* and *precision* and a similar one that uses the macro averaged values [Sokolova and Lapalme 2009]. Equations 24 and 25 show these definitions.

$$MF_\beta\mu = \frac{(1+\beta^2) \cdot Prec_\mu \cdot Rec_\mu}{\beta^2 \cdot Prec_\mu + Rec_\mu} \tag{24}$$

$$MF_{\beta M} = \frac{(1+\beta^2) \cdot Prec_M \cdot Rec_M}{\beta^2 \cdot Prec_M + Rec_M} \tag{25}$$

The macro-averaged accuracy (*MAvA*), presented by Ferri et al. [2009], is obtained with an arithmetic average over the *recall* of each class as follows:

$$MAvA = \frac{\sum_{c \in \mathbb{C}} recall(c)}{|\mathbb{C}|} \tag{26}$$

The *MAvA* measure assigns equal weights to the existing classes. Sun et al. [2006] presented the $MAvG$ metric, a generalization of the *G-Mean* for more than two classes (cf. Equation 27). The $MAvG$ is the geometric average of the *recall* score in each class.

$$MAvG = \sqrt[|\mathbb{C}|]{\prod_{c \in \mathbb{C}} recall(c)} \tag{27}$$

Finally, we highlight that the CWA measure (cf. Equation 15) presented for two-class problems, was generalized for multi-class [Cohen et al. 2006] as follows:

$$CWA = \sum_{c \in \mathbb{C}} w_c \cdot recall(c) \tag{28}$$

where $0 \leq w_c \leq 1$ and $\sum_{c \in \mathbb{C}} w_c = 1$. In this case it is the user responsibility to specify the weights w_c assigned to each class.

Although some effort has been made regarding scalar metrics for multi-class evaluation there is still a big gap regarding assessment measures for multi-class imbalanced domains. This is still an open problem, with only few solutions proposed and presenting more challenges than binary classification.

3.1.2. Graphical-based Metrics .

Two-Class Problems.

Two popular tools used in imbalanced domains are the *receiver operating characteristics* (*ROC*) curve (cf. Figure 3) and the corresponding area under the *ROC* curve (*AUC*) [Metz 1978]. Provost et al. [1998] proposed *ROC* and *AUC* as alternatives to *accuracy*. The ROC curve allows the visualization of the relative trade-off between benefits (TP_{rate}) and costs (FP_{rate}). The performance of a classifier for a certain distribution is represented by a single point in the *ROC* space. A *ROC* curve consists of several points each one corresponding to a different value of a decision/threshold parameter used for classifying an example as belonging to the positive class.

However, comparing several models through *ROC* curves is not an easy task unless one of the curves dominates all the others [Provost and Fawcett 1997]. Moreover, *ROC* curves do not provide a single-value performance score which motivates the use of *AUC*. The *AUC* allows the evaluation of the best model on average. Still, it is not biased towards the minority class. The area under the *ROC* curve (*AUC*) is given by a definite integral. Several ways exist to evaluate the *AUC*, being the trapezoidal method the most widely used. This method obtains the value of *AUC* through the use of trapezoids built with linear interpolation of the *ROC* curve points.

Another interesting property of the *AUC* regards the equivalence between the *AUC* and the probability that, given two randomly chosen examples, one from each class, the classifier will rank the positive example higher than the negative[Fawcett 2006]. This is also known as the Wilcoxon test of ranks. Using this property, the *AUC* can be determined by the following Equation:

$$AUC(c, c') = \frac{\sum_{i=1}^{n} I(y_i = c) \sum_{t=1}^{n} I(y_t = c') L(\hat{P}(c \,|\, x_i), \hat{P}(c \,|\, x_t))}{n_c \cdot n_{c'}} \tag{29}$$

where c and c' are the two classes of the problem and L is a function defined as follows:

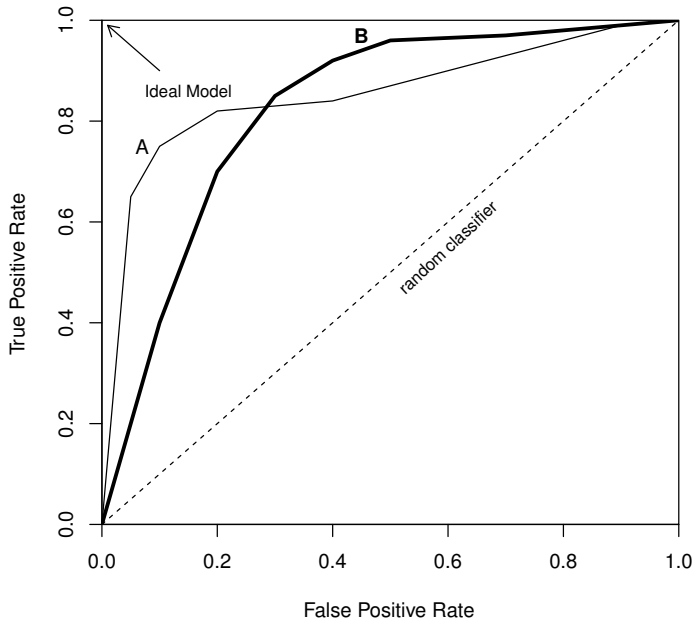

Fig. 3. *ROC* curve of three classifiers: A, B and random.

$$L(x,y) = \begin{cases} 1 & if \;\; x > y \\ 0.5 & if \;\; x = y \\ 0 & if \;\; x < y \end{cases} \tag{30}$$

AUC has become a very popular metric in the context of imbalanced domains. However, one of the problems that affects *AUC* concerns the crossing of *ROC* curves, which may produce misleading estimates. This issue results from using a single metric for summarizing a ROC curve. Another important problem of *AUC*, highlighted by Hand [2009], regards the existence of variations in the evaluation of *AUC* depending on the classifier used. This is a more serious problem because this means that the *AUC* evaluates different classifiers through the use of different measures. Hand [2009] showed that the evaluation provided by *AUC* can be misleading but has also proposed an alternative for allowing fairer comparisons: the *H-measure*. The *H-measure* is a standardized measure of the expected minimum loss obtained for a given cost distribution defined between the two classes of the problem. Hand [2009] proposes the use of a $beta(x;2,2)$ distribution for representing the cost. The advantages pointed for using this distribution are two-fold: it allows a general comparison of the results obtained by different researchers, and it gives less weight to the more extreme values of cost. Although the coherence of *AUC* was questioned by Hand, a possible coherent interpretation for this measure was also presented by Ferri et al. [2011b]. Despite surrounded with some controversy, the *AUC* is still one of the most used measures under imbalanced domains. To provide a better adaptation of this metric to these domains, several *AUC* variants were proposed for two-class problems.

A version of the *AUC* which incorporates probabilities is *Prob AUC* [Ferri et al. 2005] defined in Equation 31. The *Prob AUC* tries to overcome the problem of *AUC* measure which only considers the ranking of the examples disregarding the probabilities associated with them.

$$Prob\ AUC(c,c') = \frac{\sum_{i=1}^{n} \frac{I(y_i=c)\hat{P}(c\,|\,x_i)}{n_c} - \sum_{i=1}^{n} \frac{I(y_i=c')\hat{P}(c'\,|\,x_i)}{n_{c'}} + 1}{2} \tag{31}$$

The *Scored AUC*, presented by Wu et al. [2007], is a measure similar to *Prob AUC* that also includes probabilities in its definition (cf. Equation 32). This variant has also the goal of obtaining a score more robust to variations in the rankings that occur because of small changes in the probabilities.

$$Scored\ AUC(c,c') = \frac{\sum_{i=1}^{n} I(y_i=c) \sum_{t=1}^{n} I(y_t=c') L(\hat{P}(c\,|\,x_i)\hat{P}(c\,|\,x_t)) \cdot (\hat{P}(c\,|\,x_i) - \hat{P}(c'\,|\,x_t))}{n_c \cdot n_{c'}} \tag{32}$$

A weighted version of the *AUC*, *WAUC*, was proposed by Weng and Poon [2008] for dealing with imbalanced data sets. This new measure assumes that the area near the top of the graph is more relevant. Therefore, instead of summing the areas to obtain the *AUC* giving the same importance to all, *WAUC* progressively assigns more weight to the areas closer to the top of the ROC curve.

Precision-recall curves (*PR curves*) are recommended for highly skewed domains where *ROC* curves may provide an excessively optimistic view of the performance [Davis and Goadrich 2006]. *PR curves* have the *recall* and *precision* rates represented on the axes. A strong relation between *PR* and *ROC* curves was found by Davis and Goadrich [2006]. Figure 4 shows both curves for the imbalanced hepatitis data set [5]. The results displayed were obtained with an SVM model considering the minority class as the relevant one.

Another relevant tool for two-class problems are cost curves (Figure 5) that were introduced by Drummond and Holte [2000]. In these curves the performance (i.e. the expected cost normalized to $[0, 1]$) is represented in the y-axis. The x-axis (also normalized to $[0, 1]$) displays the probability cost function which is defined as follows:

$$PCF(+) = \frac{p(+)C(-|+)}{p(+)C(-|+) + p(+)C(+|-)} \tag{33}$$

where $p(c_1)$ represents the probability of a given class c_1 and $C(c_1|c_2)$ represents the cost of misclassifying an example of a class c_2 as being of class c_1. There is a relation of duality between ROC and cost curves. In fact, a point in the ROC space is represented by a line in the cost space and a line on ROC space is represented by a point in cost space.

Brier Curves [Ferri et al. 2011a] are a graphical representation that can be used with probabilistic binary classifiers that try to overcome an optimistic view of performance provided by cost curves. Brier curves and cost curves are complementary in the sense that these two curves used together are able to condense most of the information relative to a classifier performance.

Multi-class Problems.

Dealing with multi-class problems using graphical-based metrics is a much more complex task. A possible way for obtaining ROC curves with c different classes is to use

[5] This data set is available in UCI repository (https://archive.ics.uci.edu/ml/datasets/Hepatitis).

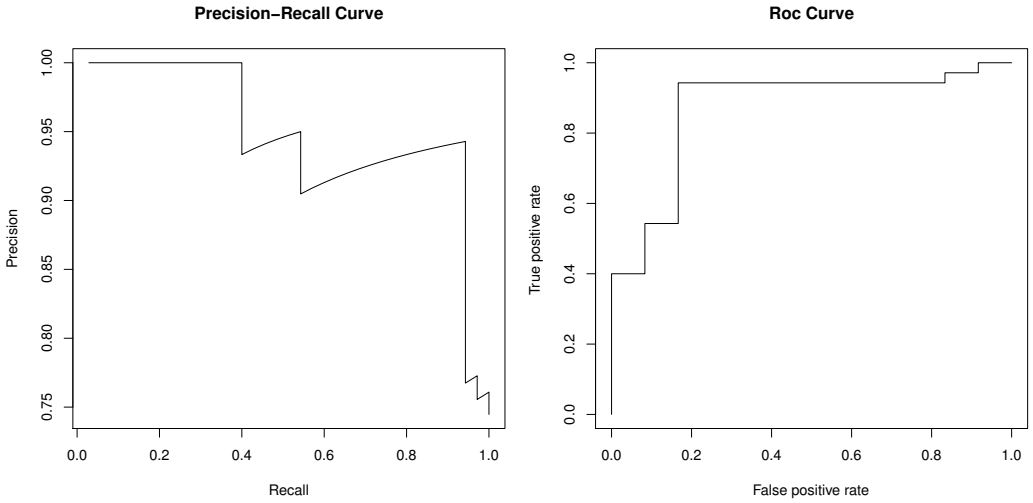

Fig. 4. Precision-recall curve and ROC curve for the hepatitis data set.

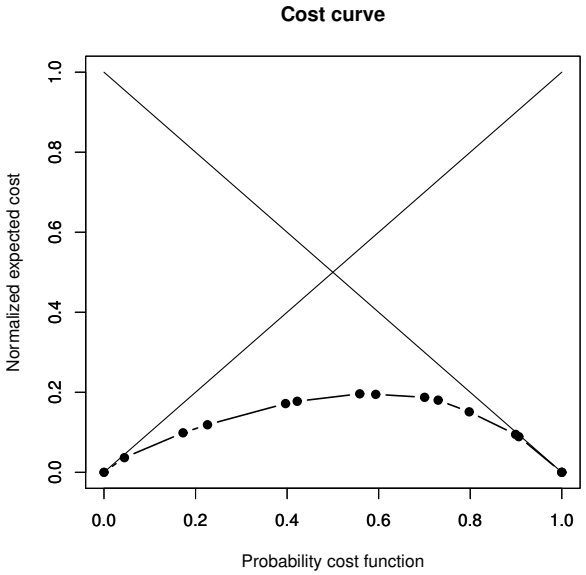

Fig. 5. Example of a Cost Curve.

the one-vs-all strategy. In this method, each class is considered as the positive class at a time and all the other classes are joined as the negative class. However, as the number of classes increases, the complexity of constructing the ROC curve grows exponentially. For the simpler case of three classes a ROC surface was proposed [Mossman 1999].

The *AUC* was also adapted to multi-class problems (e.g. Ferri et al. [2009]). Several proposals exist to accomplish this adaptation (cf. Equations 34 to 39) each one making different assumptions. *AUNU* and *AUNP* use the approach one vs all to compute the *AUC* of a $|\mathbb{C}|$-class problem transforming it into $|\mathbb{C}|$ two-class problems. Each one of the classes is considered the positive class and all the others are aggregated into one negative class. In *AUNU* classes are assumed to be uniformly distributed and in *AUNP* the prior probability of each class is taken into account. *AU1U* and *AU1P* compute the AUC of all pairs of classes, which corresponds to $|\mathbb{C}|(|\mathbb{C}|-1)$ two-class problems. The first measure considers that the classes are uniformly distributed and the latter incorporates the prior probability of the classes. Finally, *Scored AUC* and *Prob AUC* were also extended to a multi-class setting with *SAUC* (cf. Equation 38) and *PAUC* (cf. Equation 39), respectively. These two variants also consider all the combinations of pairs of classes ($|\mathbb{C}|(|\mathbb{C}|-1)$).

$$AUNU = \frac{\sum_{c \in \mathbb{C}} AUC(c, rest_c)}{|\mathbb{C}|} \tag{34}$$

where $rest_c$ is the aggregation of all the problem classes with the exception of class c.

$$AUNP = \sum_{c \in \mathbb{C}} p(c) \cdot AUC(c, rest_c) \tag{35}$$

$$AU1U = \frac{\sum_{c \in \mathbb{C}} \sum_{c' \in \mathbb{C} \setminus \{c\}} AUC(c, c')}{|\mathbb{C}|(|\mathbb{C}|-1)} \tag{36}$$

$$AU1P = \frac{\sum_{c \in \mathbb{C}} \sum_{c' \in \mathbb{C} \setminus \{c\}} p(c) \cdot AUC(c, c')}{|\mathbb{C}|(|\mathbb{C}|-1)} \tag{37}$$

$$SAUC = \frac{\sum_{c \in \mathbb{C}} \sum_{c' \in \mathbb{C} \setminus \{c\}} Scored\ AUC(c, c')}{|\mathbb{C}|(|\mathbb{C}|-1)} \tag{38}$$

$$PAUC = \frac{\sum_{c \in \mathbb{C}} \sum_{c' \in \mathbb{C} \setminus \{c\}} Prob\ AUC(c, c')}{|\mathbb{C}|(|\mathbb{C}|-1)} \tag{39}$$

Comparative studies involving some of the metrics proposed for the multi-class imbalanced context (e.g. Alejo et al. [2013]; Sánchez-Crisostomo et al. [2014]) concluded that these metrics do not always reflect correctly the performance in the minority/majority classes. This means that these metrics may not be reliable when assessing the performance in multi-class problems.

3.2. Metrics for Regression Tasks

3.2.1. Scalar Metrics.

Very few efforts have been made regarding evaluation metrics for regression tasks in imbalanced domains. Performance measures commonly used in regression, such as *Mean Squared Error* (MSE) and *Mean Absolute Error* (MAE)[6] (cf. Equations 40 and

[6]Also known as *Mean Absolute Deviation* (MAD).

41) are not adequate to these specific problems. These measures assume a uniform relevance of the target variable domain and evaluate only the magnitude of the error.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{40}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{41}$$

Although the magnitude of the numeric error is important, for tasks with imbalanced domains of the target variable, the metrics should also be sensitive to the errors location within the target variable domain, because as in classification tasks, users of these domains are frequently biased to the performance on poorly represented values of the target. This means that the error magnitude must have a differentiated impact depending on the values of the target domain where the error occurs.

In the area of finance several attempts have been made for considering differentiated prediction costs through the proposal of asymmetric loss functions [Zellner 1986; Cain and Janssen 1995; Christoffersen and Diebold 1996, 1997; Crone et al. 2005; Granger 1999; Lee 2008]. However, the proposed solutions, such as *LIN-LIN* or *QUAD-EXP* error metrics, all suffer from the same problem: they can only distinguish between over- and under-predictions. Therefore, they are still unsuitable for addressing the problem of imbalanced domains with a user preference bias towards some specific ranges of values.

Another alternative is the concept of utility-based regression [Ribeiro 2011; Torgo and Ribeiro 2007]. This concept is based on the assumption that the user assigns a non-uniform relevance to the values of the target variable domain. In this context, the usefulness of a prediction depends on both the numeric error of the prediction (which is provided by a certain loss function $L(\hat{y}, y)$) and the relevance (importance) of the predicted $\hat{y}$ and true y values. As within classification tasks, we have a problem of imbalanced domains if the user assigns more importance to predictions involving values of the target variable that are rare (i.e. poorly represented in the training sample). The proposed framework for utility-based regression provides means for easy specification of an utility function, $u(\hat{y}, y)$, for regression tasks. This means that we can use this framework to evaluate and/or compare models using the total utility of their predictions as indicated in Equation 3.

This utility-based framework was also used by Torgo and Ribeiro [2009] and Ribeiro [2011] to derive the notions of *precision* and *recall* for regression in tasks with non-uniform relevance of the target values. Based on this previous work, Branco [2014] proposed the following measures of *precision* and *recall* for regression,

$$precision = \frac{\sum_{\phi(\hat{y}_i)>t_R}(1 + u(\hat{y}_i, y_i))}{\sum_{\phi(\hat{y}_i)>t_R}(1 + \phi(\hat{y}_i))} \tag{42}$$

$$recall = \frac{\sum_{\phi(y_i)>t_R}(1 + u(\hat{y}_i, y_i))}{\sum_{\phi(y_i)>t_R}(1 + \phi(y_i))} \tag{43}$$

where $\phi(y_i)$ is the relevance associated with the true value y_i, $\phi(\hat{y}_i)$ is the relevance of the predicted value $\hat{y}_i$, t_R is a user-defined threshold signalling the cases that are relevant for the user, and $u(\hat{y}_i, y_i)$ is the utility of making the prediction $\hat{y}_i$ for the true value y_i, normalized to $[-1, 1]$.

3.2.2. Graphical-based Metrics.

Following the efforts made within classification, some attempts were made to adapt the existing notion of *ROC* curves to regression tasks. One of these attempts is the *ROC space for regression* (*RROC* space) [Hernández-Orallo 2013] which is motivated by the asymmetric loss often present on regression applications where both over-estimations and under-estimations entail different costs. *RROC* space is defined by plotting the total over-estimation and under-estimation on the x-axis and y-axis, respectively (cf. Figure 6). *RROC* curves are obtained when the notion of shift is used, which allows adjusting the model to an asymmetric operating condition by adding or subtracting a constant to the predictions. The notion of dominance can also be assessed by plotting the curves of different regression models, similarly to *ROC* curves in classification problems. Other evaluation metrics were explored, such as the *Area Over the RROC*

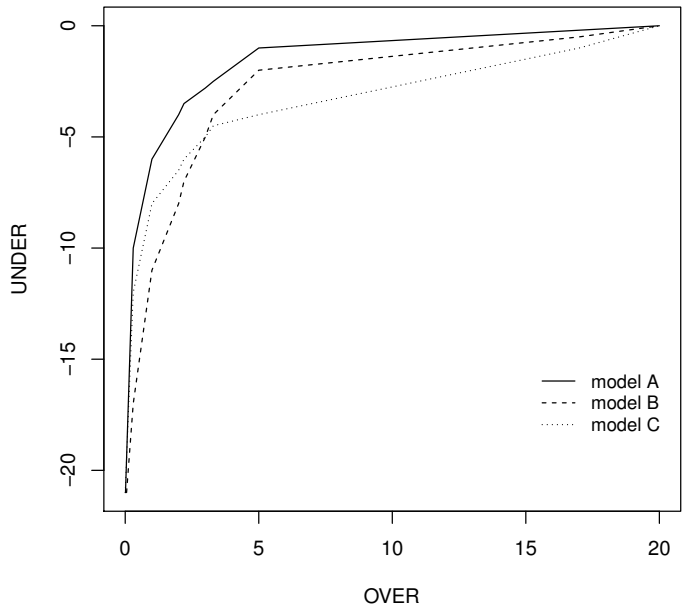

Fig. 6. *RROC* curve of three models: A, B and C.

curve (*AOC*) which was shown to be equivalent to the error variance. In spite of the relevance of this approach, it only distinguishes over from under predictions.

Another relevant effort towards the adaptation of the concept of ROC curves to regression tasks was made by Bi and Bennett [2003] with the proposal of *Regression Error Characteristic* (*REC*) curves that provide a graphical representation of the cumulative distribution function (cdf) of the error of a model. These curves plot the error tolerance and the accuracy of a regression function which is defined as the percentage of points predicted within a given tolerance ϵ. *REC* curves illustrate the predictive performance of a model across the range of possible errors (cf. Figure 7). The *Area Over the Curve* (*AOC*) can also be evaluated and is a biased estimate of the expected error of a

model [Bi and Bennett 2003]. *REC* curves, although interesting, are still not sensitive to the error location across the target variable domain.

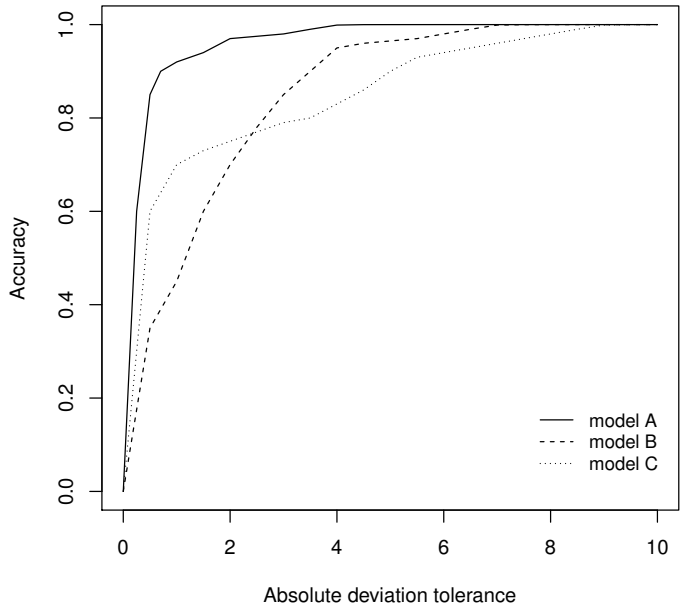

Fig. 7. *REC* curve of three models: A, B and C.

To address this problem *Regression Error Characteristic Surfaces* (*RECS*) [Torgo 2005] were proposed. These surfaces incorporate an additional dimension into *REC* curves representing the cumulative distribution of the target variable. *RECS* show how the errors corresponding to a certain point of the REC curve are distributed across the range of the target variable (cf. Figure 8). This tool allows the study of the behavior of alternative models for certain specific values of the target variable. By zooming on specific regions of REC surfaces we can carry out two types of analysis that are highly relevant for some application domains. The first involves checking how certain values of prediction error are distributed across the domain of the target variable, which tells us where errors are more frequent. The second type of analysis involves inspecting the type of errors a model has on a certain range of the target variable that is of particular interest to us, which is very relevant for imbalanced domains.

4. STRATEGIES FOR HANDLING IMBALANCED DOMAINS

Imbalanced domains raise significant challenges when building predictive models. The scarce representation of the most important cases leads to models that tend to be more focused on the normal examples, neglecting the rare events. Several strategies have been developed to address this problem, mainly in a classification setting. Even when considering solely the existing solutions for classification tasks, these are mostly biased towards binary classification. Proposals exist specifically for the multiclass case but in

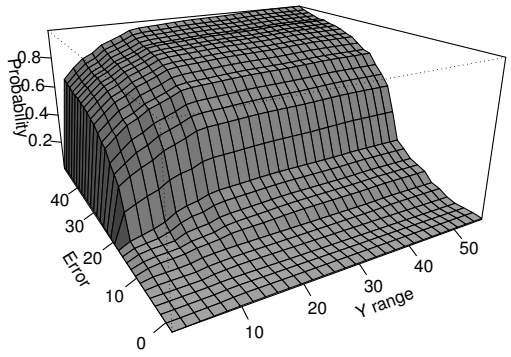

Fig. 8. An example of the *REC* surface.

a much lower number. The effectiveness and applicability of these strategies depends on the information the user is able to provide on his preference biases - the so-called "problem-definition issue" [Weiss 2013] mentioned in Section 2. We propose to group the existing approaches to learn under imbalanced domains into the following four main categories:

— Data Pre-processing;
— Special-purpose Learning Methods;
— Prediction Post-processing;
— Hybrid Methods.

Data pre-processing approaches include solutions that pre-process the given imbalanced data set, changing the data distribution to make standard algorithms focus on the cases that are more relevant for the user. These methods have the following advantages: (i) can be applied with any existing learning tool; and (ii) the chosen models are biased to the goals of the user (because the data distribution was previously changed to match these goals), and thus it is expected that the models are more interpretable in terms of these goals. The main inconvenient of this strategy is that it may be difficult to relate the modifications in the data distribution with the information provided by the user concerning the preference biases. This means that mapping the given data distribution into an optimal new distribution according to the user goals is typically not easy.

Special-purpose learning methods comprise solutions that change the existing algorithms to be able to learn from imbalanced data. The following are important advantages: (i) the user goals are incorporated directly into the models; and (ii) it is expected that the models obtained this way are more comprehensible to the user. The main disadvantages of these approaches are: (i) the user is restricted to the learning algorithms that have been modified to be able to optimize his goals, or has to develop

new algorithms for the task; (ii) if the target loss function changes, the model must be relearned, and moreover, it may be necessary to introduce further modifications in the algorithm which may not be straightforward; (iii) it requires a deep knowledge of the learning algorithms implementations; and (iv) it may not be easy to translate the user preferences into a suitable loss function that can be incorporated into the learning process.

Prediction post-processing approaches use the original data set and a standard learning algorithm, only manipulating the predictions of the models according to the user preferences and the imbalance of the data. As advantages, we can enumerate that: (i) it is not necessary to be aware of the user preference biases at learning time; (ii) the obtained model can, in the future, be applied to different deployment scenarios (i.e. different loss functions), without the need of re-learning the models or even keeping the training data available; and (iii) any standard learning tool can be used. However, these methods also have some drawbacks: (i) the models do not reflect the user preferences; (ii) the models interpretability may be jeopardized as they were obtained optimizing a loss function that is not in accordance with the user preference bias at deployment time.

Table III shows a summary of the main advantages and disadvantages of each type of strategy. Figure 9 provides a general overview of the main approaches within these strategies, which will be reviewed in Sections 4.1, 4.2 and 4.3, including solutions for both classification and regression tasks. Hybrid solutions will be addressed in Section 4.4. Hybrid methods combine approaches of different strategies trying to take advantage of their best characteristics.

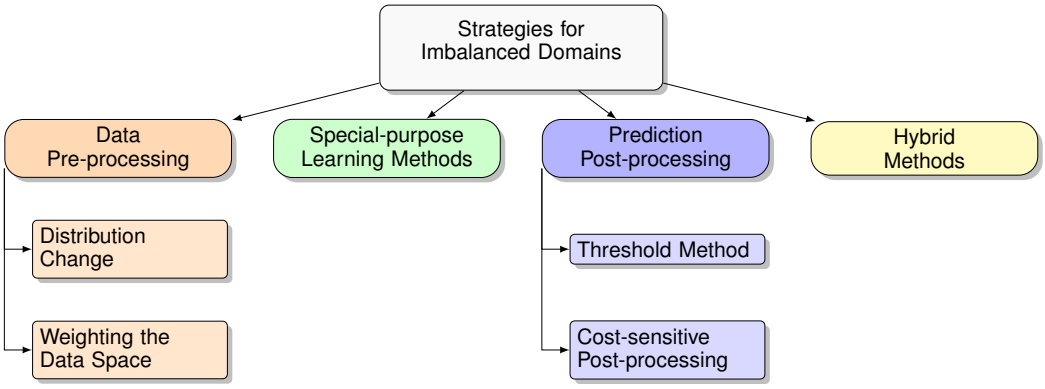

Fig. 9. Main strategies for handling imbalanced domains.

4.1. Data Pre-processing

Pre-processing strategies consist of methods of using the available data set in a way that is more in accordance with the user preference biases. This means that instead of applying a learning algorithm directly to the provided training sample, we will first somehow pre-process this data according to the goals of the user. Any standard learning algorithm can then be applied to the pre-processed data set.

Existing data pre-processing approaches can be grouped into two main types:

— **distribution change:** change the data distribution with the goal of addressing the issue of the poor representativeness of the more relevant cases; and

Table III. Main advantages and disadvantages of each type of strategy for imbalanced domains.

Strategy	Advantages	Disadvantages
Data Pre-processing	• can be applied to any learning tool • the chosen models are biased to the goals of the user • models more interpretable according to the user goals	• difficulty of relating the modifications in the data distribution and the user preferences
Special-purpose Learning Methods	• user goals are incorporated directly into the models • models obtained are more comprehensible to the user	• user is restricted in his choice of the learning algorithms that have been modified to be able to optimize his goals • models must be relearned if the target loss function changes • changes in the loss function may require further modifications in the algorithm • requires a deep knowledge of the learning algorithms implementations • not easy to map the user specification of his preferences into a loss function
Prediction Post-processing	• it is not necessary to be aware of the user preferences biases at learning time • the obtained model can, in the future, be applied to different deployment scenarios without the need of re-learning the models or even keeping the training data available • any standard learning tool can be used	• the models do not reflect the user preferences • models interpretability may be jeopardized as they were obtained optimizing a loss function that is not in accordance with the user preference bias

— **weighting the data space:** modify the training set distribution using information concerning misclassification costs, such that the learned model avoids costly errors.

Table IV summarizes the main bibliographic references for data pre-processing strategy approaches.

4.1.1. Distribution Change.

Applying a method that changes the data distribution to obtain a more balanced one is an effective solution to the imbalance problem [Estabrooks et al. 2004; Batuwita and Palade 2010a; Fernández et al. 2008, 2010].

However, changing the data distribution may not be as easy as expected. Deciding what is the optimal distribution for some user preference biases is not straightforward, even in cases where a complete specification of the utility function, $u(\hat{y}, y)$, is available. A frequently used approach consists of trying to balance the data distribu-

Table IV. Pre-processing strategy approaches, corresponding sections and main bibliographic references

Approaches (Section)			Main References
Distribution Change (4.1.1)	Stratified Sampling	Random Under/Over-sampling	Chawla et al. [2002]; Chang et al. [2003]; Drummond and Holte [2003]; Chen et al. [2004]; Estabrooks et al. [2004]; Tao et al. [2006]; Wang and Yao [2009]; Seiffert et al. [2010]; Wallace et al. [2011]; Torgo et al. [2013]
		Distance Based	Chyi [2003]; Mani and Zhang [2003]; Błaszczyński and Stefanowski [2015]
		Data Cleaning Based	Kubat and Matwin [1997]; Laurikkala [2001]; Batista et al. [2004]; Naganjaneyulu and Kuppa [2013]
		Recognition Based	Japkowicz [2000]; Chawla et al. [2004]; Raskutti and Kowalczyk [2004]; Lee and Cho [2006]; Zhuang and Dai [2006a,b]; Bellinger et al. [2012]; Wagstaff et al. [2013]
		Cluster Based	Jo and Japkowicz [2004]; Cohen et al. [2006]; Yen and Lee [2006, 2009] Sobhani et al. [2014]
		Evolutionary Sampling	Del Castillo and Serrano [2004]; García et al. [2006]; Doucette and Heywood [2008]; Drown et al. [2009]; García and Herrera [2009]; Maheshwari et al. [2011]; García et al. [2012]; Yong [2012]; Galar et al. [2013]
	Synthesizing New Data		Lee [1999, 2000]; Chawla et al. [2002, 2003]; Batista et al. [2004]; Han et al. [2005]; Liu et al. [2007]; He et al. [2008]; Bunkhumpornpat et al. [2009]; Hu et al. [2009]; Wang and Yao [2009]; Menardi and Torelli [2010]; Maciejewski and Stefanowski [2011]; Zhang et al. [2011]; Barua et al. [2012]; Bunkhumpornpat et al. [2012]; Martínez-García et al. [2012]; Ramentol et al. [2012a,b]; Verbiest et al. [2012]; Nakamura et al. [2013]; Torgo et al. [2013]; Gao et al. [2014]; Li et al. [2014]; Zhang and Li [2014];Bellinger et al. [2015]; Sáez et al. [2015]
	Combination of Methods		Liu et al. [2006]; Mease et al. [2007]; Li et al. [2008]; Stefanowski and Wilk [2008]; Chen et al. [2010]; Jeatrakul et al. [2010]; Napierała et al. [2010]; Songwattanasiri and Sinapiromsaran [2010]; Bunkhumpornpat et al. [2011]; Vasu and Ravi [2011]; Sharma et al. [2012];Yang and Gao [2012]; Ng et al. [2014]
Weighting the Data Space (4.1.2)			Zadrozny et al. [2003]

tion (e.g. make the classes have the same frequency). However, for some classifiers such as C4.5, Ripper or Naive Bayes, it was proved that a perfectly balanced distribution does not always provide optimal results [Weiss and Provost 2003]. In this context, some solutions were proposed to find the right amount of change in the distribution to be applied for a data set [Weiss and Provost 2003; Chawla et al. 2005, 2008]. For the case of extreme class imbalance, where the number of normal examples (D_N) is much larger than the number of rare examples (D_R), other class balancing methods are recommended such as 2:1 or 3:1 (majority:minority) [Khoshgoftaar et al. 2007]. These results were obtained based on experiments with 11 different types of classifiers.

For binary classification problems, changing the class distribution of the training data may improve classifiers performance on an imbalanced context because there is a connection with non-uniform misclassification costs. This equivalence between the two concepts of altering the data distribution and the misclassification cost ratio is well-known and was first pointed out by Breiman et al. [1984]. However, as mentioned by Weiss [2013], this equivalence does not hold in many real-world situations due to some of its assumptions on data availability.

The existing approaches for changing the data distribution can be of three types: stratified sampling, synthesizing new data, or combinations of the previous methods. Stratified sampling includes strategies that remove and/or add examples to the original data set. These are based on a diverse set of techniques such as: random

under/over-sampling, distance methods, data cleaning approaches, clustering algorithms or evolutionary algorithms. Approaches that synthesize new data are different because they involve the generation of new artificially generated examples that are added to the original data set. Finally, it is also possible to combine the previously described approaches. We now briefly describe the most significant techniques for changing the data distribution.

Two of the most simple approaches for data sampling that can be applied are under- and over-sampling. The first one removes data from the original data set reducing the sample size, while the second one adds data increasing the sample size. In random under-sampling, a random set of majority class examples are discarded. This may eliminate useful examples leading to a worse performance. Oppositely, in random over-sampling, a random set of copies of minority class examples is added to the data. This may increase the likelihood of overfitting, specially for higher over-sampling rates [Chawla et al. 2002; Drummond and Holte 2003]. Moreover, it may decrease the classifier performance and increase the computational effort.

Random under-sampling was also used in the context of ensembles. Namely, it was combined with boosting [Seiffert et al. 2010], bagging [Chang et al. 2003; Tao et al. 2006; Wang and Yao 2009; Wallace et al. 2011] and was applied to both classes in random forests in a method named Balanced Random Forest (BRF) [Chen et al. 2004]. An interesting theoretically-based motivation was provided in Wallace et al. [2011] for using bagging with balanced bootstrap samples obtained through random under-sampling. This theoretical approach is further explored in Section 6.

For regression tasks, Torgo et al. [2013] perform random under-sampling of the common values as a strategy for addressing the imbalance problem. This method uses a relevance function and a user defined threshold to determine which are the common and uninteresting values that should be under-sampled.

Despite the potential of randomly selecting examples, under- and over-sampling strategies can also be carried out by other, more informed, methods. For instance, under-sampling can be accomplished through the use of distance evaluations [Chyi 2003; Mani and Zhang 2003]. These approaches perform under-sampling based on a certain distance criterion that determines which are the examples from the majority class to include in the training set. Several proposals exist, ranging between the extreme cases of selecting the majority class examples that are closer to the minority class examples, or choosing the negative examples with the farthest distance to the positive examples. These strategies are very time consuming which is a major disadvantage, specially when dealing with large data sets.

Under-sampling can also be achieved through data cleaning methods. The main goal of these methods is to identify possibly noisy examples or overlapping regions and then decide on the removal of examples. One of those methods uses Tomek links [Tomek 1976] which consist of points that are each other's closest neighbors, but do not share the same class label. This method allows for two options: only remove Tomek links examples belonging to the majority class or eliminate Tomek links examples of both classes [Batista et al. 2004]. The notion of Condensed Nearest Neighbour Rule (CNN) [Hart 1968] was also applied to perform under-sampling [Kubat and Matwin 1997]. CNN is used to find a subset of examples consistent with the training set, i.e., a subset that correctly classifies the training examples using a 1-nearest neighbor classifier. CNN and Tomek links methods were combined in this order by Kubat and Matwin [1997] in a strategy called One-Sided-Selection (OSS), and in the reverse order in a proposal of Batista et al. [2004].

Recognition-based methods as one-class learning or autoencoders offer the possibility to perform the most extreme type of under-sampling where all the examples from the minority class are removed. In this type of approach, and contrary

to discrimination-based inductive learning, the model is learned using only examples of one class, and no counter examples are included. This lack of examples from the other class(es) is the key distinguishing feature between recognition-based and discrimination-based learning.

One-class learning tries to set up boundaries which surround the majority class concept. This method starts by measuring the similarity between the majority class and an object. Classification is then performed using a threshold on the obtained similarity score. One-class learning methods have the disadvantage of requiring the tuning of the threshold imposed on the similarity. In fact, this is a sensitive issue because if we choose a too narrow threshold the majority class examples are disregarded. However, too wide thresholds may lead to including examples from the minority class. Therefore, establishing an efficient threshold is vital with this method. Also, some learners actually need examples from more than one class and are unable to adapt to this method. Despite all these possible disadvantages, recognition-based learning algorithms have been shown to provide good prediction performance in most domains. Developments made in this context include one-class SVMs (e.g. Schölkopf et al. [2001]; Manevitz and Yousef [2002]; Raskutti and Kowalczyk [2004]; Zhuang and Dai [2006b,a]; Lee and Cho [2006]) and the use of an autoencoder (or autoassociator) (e.g. Japkowicz et al. [1995]; Japkowicz [2000]).

An innovative recognition based-method for large data sets was proposed by Wagstaff et al. [2013] that aims at both facilitating the discovery of novel observations and at providing an explanation for the detected cases. This is achieved through an incremental Singular Value Decomposition (SVD) method that allows the selection of examples with high novelty which is measured by reconstruction error.

Imbalanced domains can influence the performance and the efficiency of clustering algorithms [Xuan et al. 2013]. However, due to their flexibility, several approaches appeared for dealing with imbalanced data sets using clustering methods. For instance, the cluster-based oversampling (CBO) algorithm proposed by Jo and Japkowicz [2004] addresses both the imbalance problem and the problem of small disjuncts. Small disjuncts are subclusters of a certain class which have a low coverage, i.e., classify only few examples [Holte et al. 1989]. CBO consists of clustering the training data of each class separately with the k-means technique and then performing random oversampling in each cluster. All majority class clusters are over-sampled until they reach the cardinality of the largest cluster of this class. Then the minority class clusters are over-sampled until both classes are balanced maintaining all minority class subclusters with the same number of examples. Several other proposals based on clustering techniques exist (e.g. Yen and Lee [2006, 2009]; Cohen et al. [2006]). Recently, clustering techniques were also combined with ensembles Sobhani et al. [2014]. This proposal starts by clustering the majority class examples. Then, several classifiers are trained in balanced data sets that use all the minority class examples and at least one majority class example from each previously determined cluster. A majority voting scheme is used to obtain the final class label.

Another approach for data sampling concerns the use of Evolutionary Algorithms (EA). These algorithms started to be applied to imbalanced domains as a strategy to perform under-sampling through a prototype selection (PS) procedure (e.g. García et al. [2006]; García and Herrera [2009]).

García et al. [2006] made one of the first contributions with a new evolutionary method proposed for balancing the data set. The presented method uses a new fitness function designed to perform a prototype selection process. Some proposals have also emerged in the area of heuristics and metrics for improving several genetic programming classifiers performance in imbalanced domains [Doucette and Heywood 2008].

However, EA have been used for more than under-sampling. More recently, Genetic Algorithms (GA) and clustering techniques were combined to perform both under and over-sampling [Maheshwari et al. 2011; Yong 2012]. Evolutionary under-sampling has also been combined with boosting [Galar et al. 2013].

Another important approach for dealing with the imbalance problem as a pre-processing step, is the generation of new synthetic data. Several methods exist for building new synthetic examples. Most of the proposals are focused on classification tasks. Synthesizing new data has several known advantages [Chawla et al. 2002; Menardi and Torelli 2010], namely: (i) reduces the risk of overfitting which is introduced when replicas of the examples are inserted in the training set; (ii) improves the ability of generalization which was compromised by the over-sampling methods. The methods for synthesizing new data can be organized in two groups: (i) one that introduces perturbations, and (ii) another that uses interpolation of existing examples.

Lee [1999] proposed an over-sampling method that produces noisy replicates of the rare cases while keeping the majority class unchanged. The synthetic examples are generated by adding normally distributed noise to the minority class examples. This simple strategy was tested with success, and a new version was developed by Lee [2000]. This new approach generates, for a given data set, multiple versions of training sets with added noise. Then, an average of multiple model estimates is obtained.

Recently, Bellinger et al. [2015] proposed a new method for generating synthetic sample named DEAGO. This proposal is based on the capabilities of recontruction of denoising autoencoders [Vincent et al. 2010]. The denoising autoencoders are neural networks that are able to reconstruct at the output layer clean versions of the network input. DEAGO generates synthetic samples with Gaussian noise added which are then used as input of the denoising autoencoders. This proposal was evaluated for the gamma-ray spectral domain.

Another framework, named ROSE (Random Over Sampling Examples), for dealing with the problem of imbalanced classification was presented by Menardi and Torelli [2010] based on a smoothed bootstrap re-sampling technique. ROSE generates a more balanced and completely new data set from the given training set combining over- and under-sampling. One observation is drawn from the training set by giving the same probability to both existing classes. A new example is generated in the neighborhood of this observation, using a width for the neighborhood determined by a chosen smoothing matrix.

Zhang and Li [2014] use a random walk based approach as an over-sampling strategy to generate new examples from the minority class. This approach allows the extension of the classification border.

A famous method that uses interpolation is the synthetic minority over-sampling technique - SMOTE [Chawla et al. 2002]. SMOTE over-samples the minority class by generating new synthetic data. This technique is then combined with a certain percentage of random under-sampling of the majority class that depends on a user defined parameter. Artificial data is created using an interpolation strategy that introduces a new example along the line segment joining a seed example and one of its k minority class nearest neighbors. The number of minority class neighbors (k) is another user defined parameter. For each minority class example a certain number of examples is generated according to a predefined over-sampling percentage.

SMOTE algorithm has been applied with several different classifiers and was also integrated with boosting [Chawla et al. 2003] and bagging [Wang and Yao 2009].

Nevertheless, SMOTE generates synthetic examples with the positive class label disregarding the negative class examples which may lead to overgeneralization [Yen and Lee 2006; Maciejewski and Stefanowski 2011; Yen and Lee 2009]. This strategy may be specially problematic in the case of highly skewed class distributions where

the minority class examples are very sparse, thus resulting in a greater chance of class mixture.

Some of the drawbacks identified in SMOTE algorithm motivated the appearance of several variants of this method. We can identify three main types of variants: (i) application of some pre- or post-processing before or after the use of SMOTE; (ii) apply SMOTE only in some selected regions of the input space; or (iii) introducing small modifications to the SMOTE algorithm. Most of the first type of SMOTE variants start by applying the SMOTE algorithm and, afterwards, use a post-processing mechanism for removing some data. Examples of this type of approaches include: SMOTE+Tomek [Batista et al. 2004], SMOTE+ENN [Batista et al. 2004], SMOTE+FRST [Ramentol et al. 2012b] or SMOTE+RSB [Ramentol et al. 2012a]. An exception is the Fuzzy Rough Imbalanced Prototype Selection (FRIPS) [Verbiest et al. 2012] method that pre-processes the data set before applying the SMOTE algorithm. The second type of SMOTE variants only generates synthetic examples in specific regions that are considered useful for the learning algorithms. As the notion of what is a good region is not straightforward, several strategies were developed. Some of these variants focus the synthesizing effort on the borders between classes while others try to find which are the harder to learn instances and concentrate on these ones. Examples of these approaches are: Borderline-SMOTE [Han et al. 2005], ADASYN [He et al. 2008], Modified Synthetic Minority Oversampling Technique (MSMOTE) [Hu et al. 2009], MWMOTE [Barua et al. 2012], FSMOTE [Zhang et al. 2011], among others. Regarding the last type of SMOTE variants, some modifications are introduced in the way SMOTE generates the synthetic examples. For instance, the synthetic examples may be generated closer or further apart from a seed depending on some measure. The following proposals are examples within this group: Safe-Level-SMOTE [Bunkhumpornpat et al. 2009], Safe Level Graph [Bunkhumpornpat and Subpaiboonkit 2013], LN-SMOTE [Maciejewski and Stefanowski 2011] and DBSMOTE [Bunkhumpornpat et al. 2012].

For regression problems only one method for generating new synthetic data was proposed. Torgo et al. [2013] have adapted the SMOTE algorithm to regression tasks. Three key components of the SMOTE algorithm required adaptation for regression: (i) how to define which are the relevant observations and the "normal" cases; (ii) how to generate the new synthetic examples (i.e. over-sampling); and (iii) how to determine the value of the target variable in the synthetic examples. Regarding the first issue, a relevance function and a user-specified threshold were used to define D_R and D_N sets. The observations in D_R are over-sampled, while cases in D_N are under-sampled. For the generation of new synthetic examples the same interpolation method used in SMOTE for classification was applied. Finally, the target value of each synthetic example was calculated as an weighted average of the target variable values of the two seed examples. The weights were calculated as an inverse function of the distance of the generated case to each of the two seed examples.

Finally, several other interesting methods have appeared which combine some of the previous techniques [Stefanowski and Wilk 2008; Bunkhumpornpat et al. 2011; Songwattanasiri and Sinapiromsaran 2010; Yang and Gao 2012]. For instance, Jeatrakul et al. [2010] presents a method that uses Complementary Neural Networks (CMTNN) to perform under-sampling and combines it with SMOTE. The combination of strategies was also applied to ensembles (e.g. Liu et al. [2006]; Mease et al. [2007]; Chen et al. [2010]). An interesting approach that combines clustering with recognition-based methods was proposed by Sharma et al. [2012]. This method starts by applying a clustering algorithm and then, in each determined cluster a one-class learner is trained. The final model is obtained by combining the predictions of all the one-class learners trained.

Some attention has also been given to SVMs, leading to proposals such as the one of Kang and Cho [2006] where an ensemble of under-sampled SVMs is presented. Multiple different training sets are built by sampling examples from the majority class and combining them with the minority class examples. Each training set is used for training an individual SVM classifier. The ensemble is produced by aggregating the outputs of all individual classifiers. Another similar approach is the EnSVM [Liu et al. 2006] which adopts a rebalance strategy combining the over-sampling strategy of SMOTE algorithm and under-sampling to form a number of new training sets while using all the positive examples. Then, an ensemble of SVMs is built.

Several ensembles have been adapted and combined with approaches for changing the data distribution to better tackle the problem of imbalanced domains. Essentially, for every type of ensembles, some attempt has been made. For a more complete review on ensembles for the class imbalance problem see Galar et al. [2012].

4.1.2. Weighting the Data Space.

The strategy of weighting the data space is a way of implementing cost-sensitive learning and thus can be an effective method for handling imbalanced domains when information on the costs of errors is available. In fact, misclassification costs are applied to the given data set with the goal of selecting the best training distribution. Essentially, this method is based on the fact that changing the original sampling distribution by multiplying each case by a factor that is proportional to its importance (relative cost), allows any standard learner to accomplish expected cost minimization on the original distribution. Although it is a simple technique and easy to apply, it also has some drawbacks. There is a risk of model overfitting and it is also possible that the real cost values are unavailable which can introduce the extra difficulty of exploring effective cost setups.

This approach has a strong theoretical foundation, building on the *Translation Theorem* derived by Zadrozny et al. [2003]. Namely, to obtain a modified distribution biased towards the costly classes, the training set distribution is modified with regards to misclassification costs.

Zadrozny et al. [2003] presented two different ways of accomplishing this conversion: in a transparent box or in a black box way. In the first, the weights are provided to the classifier while for the second a careful subsampling is performed according to the same weights. The first approach cannot be applied to an arbitrary learner, while the second one results in severe overfitting if sampling with replacement is used. Thus, to overcome the drawbacks of the latter approach, the authors have presented a method called *cost-proportionate rejection sampling* which accepts each example in the input sample with probability proportional to its associated weight.

4.2. Special-purpose Learning Methods

The approaches at this level consist of solutions that modify existing algorithms to provide a better fit to the user preferences. The task of developing a solution based on algorithm modifications is not an easy one. It requires a deep knowledge of both the learning algorithm and also of the user preference biases. In order to perform a modification on a selected algorithm, it is essential to understand why it fails when the distribution does not match the user preferences. Moreover, any adaptation requires information on the full utility function, which is frequently hard to obtain. On the other hand, these methods have the advantage of being very effective in the contexts for which they were designed.

Existing solutions for dealing with imbalanced domains at the learning level are focused on the introduction of modifications in the algorithm preference criterion. Table V summarizes the main bibliographic references for this type of approaches.

Table V. Special-purpose Learning Methods, corresponding section and main bibliographic references

Strategy type (Section)	Main References
Special-purpose Learning Methods (4.2)	Joshi et al. [2001]; Barandela et al. [2003]; Maloof [2003]; Ribeiro and Torgo [2003]; Tan et al. [2003]; Torgo and Ribeiro [2003]; Wu and Chang [2003]; Akbani et al. [2004]; Chen et al. [2004]; Huang et al. [2004]; Wu and Chang [2005]; Imam et al. [2006]; Tang and Zhang [2006]; Zhou and Liu [2006]; Alejo et al. [2007]; Sun et al. [2007]; Cieslak and Chawla [2008]; Li et al. [2009]; Song et al. [2009]; Tang et al. [2009]; Batuwita and Palade [2010b]; Liu et al. [2010]; Wang and Japkowicz [2010]; Hwang et al. [2011]; Oh [2011]; Ribeiro [2011]; Cieslak et al. [2012]; Rodríguez et al. [2012]; Weiguo et al. [2012]; Xiao et al. [2012]; Cao et al. [2013]; Castro and de Pádua Braga [2013]

The incorporation of benefits and/or costs (negative benefits) in existing algorithms, as a way to express the utility of different predictions, is one of the known approaches to cope with imbalanced domains. This includes the well known cost-sensitive algorithms for classification tasks which directly incorporate costs in the learning process. In this case, the goal of the prediction task is to minimize expected cost, knowing that misclassified examples may have different costs.

The research literature includes several works describing the adaptation of different classifiers in order to make them cost-sensitive. For decision trees, the impact of the incorporation of costs under imbalanced domains was addressed by Maloof [2003]. Regarding support vector machines, several ways of integrating costs have been considered such as assigning different penalties to false negatives and positives [Akbani et al. 2004] or including a weighted attribute strategy [Yuanhong et al. 2009] among others [Weiguo et al. 2012]. Regarding neural networks, the possibility of making them cost-sensitive has also been considered (e.g. Zhou and Liu [2006]; Alejo et al. [2007]; Oh [2011]). A Cost-Sensitive Multilayer Perceptron (CSMLP) algorithm was proposed by Castro and de Pádua Braga [2013] for asymmetrical learning of MLPs via a modified (backpropagation) weight update rule. Cao et al. [2013] present a framework for improving the performance of cost-sensitive neural networks that uses Particle Swarm Optimization (PSO) for optimizing misclassification cost, feature subset and intrinsic structure parameters. Alejo et al. [2007] propose two strategies for dealing with imbalanced domains using RBF neural networks which include a cost function in the training phase.

Ensembles have also been considered in the cost-sensitive framework to handle imbalanced domains. Several ensemble methods have been successfully adapted to include costs during the learning phase. However, boosting was the most extensively explored. AdaBoost is the most representative algorithm of the boosting family. When the target class is imbalanced, AdaBoost biases the learning (through the weights) towards the majority class, as it contributes more to the overall accuracy. Several proposals appeared which modify AdaBoost weight update process by incorporating cost items so that examples from different classes are treated unequally. Important proposals in the context of imbalanced domains are: RareBoost [Joshi et al. 2001], AdaC1, AdaC2 and AdaC3 [Sun et al. 2007], and BABoost [Song et al. 2009]. All of them modify the AdaBoost algorithm by introducing costs in the used weight updating formula. These proposals differ in how they modify the update rule. Wang and Japkowicz [2010] proposes an ensemble of SVMs with asymmetric misclassification costs. The proposed system works by modifying the base classifier (SVM) using costs and uses boosting as

the combination scheme. Random Forests have also been adapted to better cope with imbalanced domains undergoing a cost-sensitive transformation. Chen et al. [2004] proposes a method called Weighted Random Forest (WRF) for dealing with highly imbalanced domains based on the Random Forest algorithm. WRF strategy operates by assigning a higher misclassification cost to the minority class. For an extensive review on ensembles for handling class imbalance see Galar et al. [2012].

Several other solutions exist that also modify the preference criteria of the algorithms while not relying directly on the definition of a cost/cost-benefit matrix. Regarding SVMs, several proposals try to bias the algorithm so that the hyperplane is further away from the positive class because the skew associated with imbalanced data sets pushes the hyperplane closer to the positive class. Wu and Chang [2003] accomplish this with an algorithm that changes the kernel function. Fuzzy Support Vector Machines for Class Imbalance Learning (FSVM-CIL) was a method proposed by Batuwita and Palade [2010b]. This algorithm is based on an SVM variant for handling the problem of outliers and noise called FSVM [Lin and Wang 2002] and improves it for also dealing with imbalanced data sets. Potential Support Vector Machine (P-SVM) [Mangasarian and Wild 2001] differs from standard SVM learners by defining a new objective function and constraints. An improved P-SVM algorithm [Li et al. 2009] was proposed to better cope with imbalanced data sets.

k-NN learners were also adapted to cope with the imbalance problem. Barandela et al. [2003] present a weighted distance function to be used in the classification phase of k-NN without changing the class distribution. This method assigns different weights to the respective classes and not to the individual prototypes. Since more weight is given to the majority class, the distance to minority class examples becomes much lower than the distance to examples from the majority class. This biases the learner to find their nearest neighbor among examples of the minority class.

A new decision tree algorithm - Class Confidence Proportion Decision Tree (CCPDT) - was proposed by Liu et al. [2010]. CCPDT is robust and insensitive to class distribution and generates rules that are statistically significant. The algorithm adopts a new proposed measure, called Class Confidence Proportion (CCP), which forms the basis of CCPDT. CCP measure is embedded in the information gain and used as the splitting criterion. In this algorithm, a new approach, using Fisher exact test, to prune branches of the tree that are not statistically significant is presented.

Hellinger distance was introduced as a decision tree splitting criterion to build Hellinger Distance Decision Trees (HDDT) [Cieslak and Chawla 2008]. This proposal was shown to be insensitive towards class imbalanced domains. More recently, Cieslak et al. [2012] recommended the use of bagged HDDTs as the preferred method for dealing with imbalanced domains when using decision trees.

For regression tasks, some works have addressed the problem of imbalanced domains by changing the splitting criterion of regression trees (e.g. Torgo and Ribeiro [2003]; Ribeiro and Torgo [2003]).

The Kernel Boundary Alignment algorithm (KBA) is proposed in Wu and Chang [2005]. This method adjusts the boundary towards the majority class by modifying the kernel matrix generated by a kernel function according to the imbalanced domain.

An ensemble method for learning over multi-class imbalanced data sets, named ensemble Knowledge for Imbalance Sample Sets (eKISS), was proposed by Tan et al. [2003]. This algorithm was specifically designed to increase classifiers sensitivity without losing the corresponding specificity. The eKISS approach combines the rules of the base classifiers to generate new classifiers for final decision making.

Recently, more sophisticated approaches were proposed as the Dynamic Classifier Ensemble method for Imbalanced Data (DCEID) presented by Xiao et al. [2012].

DCEID combines dynamic ensemble learning with cost-sensitive learning and is able to adaptively select the more appropriate ensemble approach.

For regression problems, one work exists that is able to tackle the problem of imbalanced domains through an utility-based algorithm. The utility-based Rules (ubaRules) approach was proposed by Ribeiro [2011]. ubaRules is an utility-based regression rule ensemble system designed for obtaining models biased according to a specific utility function. The system main goal is to obtain accurate and interpretable predictions in the context of regression problems with non-uniform utility. It consists in two main steps: generation of different regression trees, which are converted to rule ensembles, and selection of the best rules to include in the final ensemble. An utility function is used as criterion at several stages of the algorithm.

4.3. Prediction Post-processing

For dealing with imbalanced domains at the post-processing level, we will consider two main types of solutions:

— **threshold method:** uses the ranking provided by a score, that expresses the degree to which an example is a member of a class, to produce several learners by varying the threshold for class membership;
— **cost-sensitive post-processing:** associates costs to prediction errors and minimizes the expected cost.

Table VI summarizes the main bibliographic references of post-processing strategy approaches.

Table VI. Post-processing strategy approaches, corresponding sections and main bibliographic references

Approaches (Section)	Main References
Threshold Method (4.3.1)	Maloof [2003]; Weiss [2004] Hernández-Orallo et al. [2012]
Cost-sensitive Post-processing (4.3.2)	Hernández-Orallo [2012, 2014]

4.3.1. Threshold Method.

Some classifiers are named soft classifiers because they provide a score which expresses the degree to which an example is a member of a class. Together with a threshold, this score can be used to generate other classifiers. This can be accomplished by varying the threshold for an example belonging to a class [Weiss 2004]. A study of this method [Maloof 2003] concluded that the operations of moving the decision threshold, applying a sampling strategy, and adjusting the cost matrix produce classifiers with the same performance.

The proposal of Hernández-Orallo et al. [2012] explores several threshold choice methods and provides an interesting interpretation for a diversity of performance metrics. The threshold choice methods are categorized according to the operating conditions. Guidelines are provided regarding the performance metric that should be used based on the information available on the threshold choice method.

4.3.2. Cost-sensitive Post-processing.

Several methods exist for making models cost-sensitive in a post hoc manner. This technique was mainly explored in classification tasks and aims at changing the model predictions for making it cost-sensitive (e.g. Domingos [1999]; Sinha and May [2004]). This means that this technique could potentially be applicable to imbalanced domains.

Table VII. Hybrid strategies, corresponding sections and main bibliographic references

Strategy type (Section)	Main References
Hybrid Strategies (4.4)	Estabrooks and Japkowicz [2001]; Kotsiantis and Pintelas [2003]; Estabrooks et al. [2004]; Phua et al. [2004]; Yoon and Kwek [2005]; Ertekin et al. [2007a,b]; Zhu and Hovy [2007]; Liu et al. [2009]; Ghasemi et al. [2011a,b]; Ertekin [2013]; Mi [2013]; Barnab-Lortie et al. [2015]

However, to the best of our knowledge, these methods have never been applied or evaluated on these tasks.

In regression, introducing costs at a post-processing level has only recently been proposed [Bansal et al. 2008; Zhao et al. 2011]. It is an issue still under-explored with few limited solutions. Similarly to what happens in classification, no progress was yet made for evaluating these solutions in imbalanced domains. However, one interesting proposal called reframing [Hernández-Orallo 2012, 2014] was recently presented. Although not developed specifically for imbalanced domains, this framework aims at adjusting the predictions of a previously built model to different deployment contexts. Therefore, it is also potentially suitable for being applied to the problem of imbalanced domains. The notion of reframing was established as the process of applying a previously built model to a new operating context by the proper transformation of inputs, outputs and patterns. The reframing framework acts at a post-processing level, changing the obtained predictions by adapting them to a different distribution.

The reframing method essentially consists of two steps:

— the conversion of any traditional crisp regression model with one parameter into a soft regression model with two parameters, seen as a normal conditional density estimator (NCDE), by the use of enrichment methods;
— the reframing of an enriched soft regression model to new contexts by an instance-dependent optimization of the expected loss derived from the conditional normal distribution.

4.4. Hybrid Methods

In recent years, several methods involving the combination of some of the basic approaches described in the previous sections, have appeared in the research literature. Due to their characteristics, these methods can be seen as hybrid methods to handle imbalanced domains. They try to capitalize on some of the main advantages of the different approaches we have described previously.

Existing hybrid approaches combine the use of pre-processing approaches with special-purpose learning algorithms. Table VII summarizes the main bibliographic references concerning these hybrid strategies.

One of the first hybrid strategies was presented by Estabrooks and Japkowicz [2001] and Estabrooks et al. [2004]. The motivation for this proposal is related to the fact that a perfectly balanced data may not be optimal and that the right amount of over/under-sample to apply is difficult to determine. To overcome these difficulties, a mixture-of-experts framework was proposed in an architecture with three levels: a classifier level, an expert level and an output level. The system has two experts in the expert level: an under-sampling expert and an over-sampling expert. The architecture incorporates 10 classifiers on the over-sampling expert and another 10 classifiers on the under-sampling expert. All these classifiers are trained in data sets sampled at different

rates of over- and under-sampling, respectively. At the classifier level, an elimination strategy is applied for removing the learners that are considered unreliable according to a predefined test. Then, a combination scheme is applied both at the expert and output levels. These combination schemes use the following simple heuristic: if one of the classifiers decides that the example is positive so does the expert, and if one of the two experts decides that the example is positive so does the output level. This strategy is clearly heavily biased towards the minority (positive) class.

A different idea involving sampling and the combination of different learners was proposed by Kotsiantis and Pintelas [2003]. The proposed approach uses a facilitator agent and three learning agents each one with its own learning system. The facilitator starts by filtering the features of the data set. The filtered data is then passed to the three learning agents. Each learning agent samples the data set, learns using the respective system (Naive Bayes, C4.5 and 5NN) and returns the predictions for each instance back to the facilitator agent. Finally, the facilitator makes the final prediction according to majority voting.

In the proposal of Phua et al. [2004] sampling is performed and, afterwards, stacking and boosting are used together. The applied sampling strategy partitions the data set into eleven new data sets which include all the minority class examples and a portion of the majority class examples. The proposed system uses three different learners (Naive Bayes, C4.5 and back-propagation classifier) each one processing the eleven partitions of the data. Bagging is used to combine the classifiers trained by the same algorithm. Then stacking is used to combine the multiple classifiers generated by the different algorithms identifying the best mix of classifiers.

Other approaches combine pre-processing techniques with bagging and boosting, simultaneously, composing an ensemble of ensembles. EasyEnsemble and BalanceCascade algorithms [Liu et al. 2009] are examples of this type of approach. Both algorithms use bagging as the main ensemble method and use AdaBoost for training each bag. As for the pre-processing technique, both construct balanced bags by randomly under-sampling examples from the majority class. In EasyEnsemble algorithm all AdaBoost iterations can be performed simultaneously because each AdaBoost ensemble uses a previously determined subset of the data. All the generated classifiers are combined for a final solution. On the other hand, in the BalanceCascade algorithm, after the AdaBoost learning, the majority examples correctly classified with higher confidence are discarded from further iterations.

Wang [2008] presents an approach that combines the SMOTE algorithm with Biased-SVM [Veropoulos et al. 1999]. The proposed approach applies the Biased-SVM in the imbalanced data and stores the obtained support vectors from both classes. Then SMOTE is used to over-sample the support vectors with two alternatives: using only the obtained support vectors or using the entire minority class. A final classification is obtained with the new data using the biased-SVM.

Active learning is a semi-supervised strategy in which the learning algorithm is able to interactively obtain information from the user. Although this method is traditionally used with unlabelled data, it can also be applied when all class labels are known. In this case, the active learning strategy provides the ability of actively selecting the best, i.e. the most informative, examples to learn from. Active Learning by itself is a technique that is able to deal with moderate imbalanced distributions. However, when a more severe imbalance occurs in the data, special techniques developed for active learning that incorporate a preference towards the least represented and more relevant cases (D_R) should be used [Attenberg and Ertekin 2013].

Several approaches for imbalanced domains based on active learning have been proposed [Ertekin et al. 2007a,b; Zhu and Hovy 2007; Ertekin 2013]. These approaches are concentrated on SVM learning systems and are based on the fact that, for this type

of learners, the most informative examples are the ones closest to the hyperplane. This property is used to guide under-sampling by selecting the most informative examples, i.e., choosing the examples closer to the hyperplane.

More recent developments try to combine active learning with other techniques to further improve the learner's performance. Ertekin [2013] presents a novel adaptive over-sampling algorithm named Virtual Instances Resampling Technique Using Active Learning (VIRTUAL), that combines the benefits of over-sampling and active learning. Contrary to traditional sampling methods, which are applied before the training stage, VIRTUAL generates synthetic examples for the minority class during the training process. Therefore, the need for a separate pre-processing step is discarded. In the context of learning with SVMs, VIRTUAL outperforms competitive over-sampling techniques both in terms of generalization performance and computational complexity. Mi [2013] developed a method that combines SMOTE and active learning with SVMs.

Some efforts have also been made for integrating active learning with other classifiers. Hu [2012] proposed an active learning method for imbalance data using the Localized Generalization Error Model (L-GEM) of radial basis function neural networks (RBFNN).

Ghasemi et al. [2011a,b] presented a new approach that also uses active learning methods but only requires examples from the majority class. In these works several scoring functions for selecting the most informative examples were experimented.

A proposal considering the integration of active learning and one-class classifiers was also presented by Barnab-Lortie et al. [2015].

Still, we must highlight that, overall, active learning-based methods tend to show a degradation in performance as the imbalance of the domain increases [Attenberg and Ertekin 2013].

Finally, a strategy using a clustering method based on class purity maximization is proposed by Yoon and Kwek [2005]. This method generates clusters of pure majority class examples and non-pure clusters based on the improvement of the clusters class purity. When the clusters are formed, all minority class examples are added to the non-pure clusters and a decision tree is built for each cluster. An unlabelled example is clustered according to the same algorithm. If it falls on a non-pure cluster, the decision tree committee votes the prediction, but if it falls on a pure majority class cluster the final prediction is the majority class. If the committee votes for a majority class prediction, then that will be the final prediction. On the other hand, if it is a minority class prediction, then the example will be submitted to a final classifier which is constructed using a neural network.

5. STUDIES ON THE EFFECTIVENESS OF THE METHODS

The task of evaluating and comparing all the proposed solutions for handling the problem of imbalanced domains is not simple. First of all, there is a huge amount of proposals to deal with imbalanced domains. Secondly, the impact of the strategies on different learning algorithms is not uniform (e.g. Van Hulse et al. [2007]), meaning that any conclusions are frequently algorithm-dependent. Finally, there is also the issue of assessing the impact in performance of different levels of imbalance in the domain and of different data set characteristics such as separability of data or the training set size.

The main questions that we would like to answer regarding the performance assessment under imbalanced domains are:

— Which data characteristics contribute to further hinder the performance under imbalanced domains?
— Can we find approaches that generally provide the best improvement in the performance for these domains?

— Is the performance of the used learning algorithms affected in different degrees under imbalanced domains?
— How does the different degree of imbalance in the data distribution affects the performance?

Japkowicz and Stephen [2002] conducted one of the first studies to address these questions in a classification setting. This work appeared in an early stage of the development of these approaches and therefore only five strategies were compared (random under/over-sampling, under/over-sampling at random but focused in parts of the input space far/close to the decision boundary and finally change the misclassification costs of the classes). Unfortunately, most of the conclusions of this paper were based on comparisons of the error rate as the performance assessment measure, which is an unsuitable measure for these domains. The main conclusions were the following:

— When using decision trees:
  — the impact of the imbalanced domain increases as the data separability decreases;
  — by increasing the training set size, the impact of the imbalance in the domain is reduced;
  — the imbalance of the domain is only a problem when small disjuncts are present in the data;
  — oversampling generally outperforms undersampling;
  — changing the misclassification cost of the classes generally performs better than random or focused oversampling.
— Decision trees were found to be the classifier most sensitive to the problem of imbalanced domains, multi-layer perceptrons came next showing less sensitivity and, finally, support vector machines are identified as showing no sensitivity at all to this problem.

Batista et al. [2004] highlighted the importance of the contribution of other factors, such as small sample size and class overlap, in the performance degradation when learning under imbalanced data sets. This work uses only decision trees and compares 10 pre-processing strategies using AUC. In general, it is concluded that oversampling-based strategies have more advantages than undersampling.

The results obtained in the two previously mentioned works do not always agree with other works on this issue where oversampling is reported to be ineffective when using decision trees (e.g. Drummond and Holte [2003]). In fact, random undersampling is nowadays generally considered as one of the most efficient approaches to deal with imbalanced domains.

More recently, a new experimental design was proposed [Batista et al. 2012; Prati et al. 2014] to overcome the difficulty in assessing the capability of recovering from the losses in performance caused by imbalance. One of the main conclusion of this work is in agreement with the previously mentioned papers regarding the poor sensitivity of support vector machines to the imbalance in the domain. These were found to be the classifiers least affected by imbalanced domains, only presenting some sensitivity to the most severely imbalanced domains.

The authors used real data sets and for each data set several training set distributions were generated with the same number of examples and different degrees of imbalance. The performance loss was measured relatively to the perfectly balanced distribution using the following metric,

$$L = \frac{B - I}{B} \tag{44}$$

where B represents the performance on the perfectly balanced distribution and I the performance obtained on the imbalanced distribution. The AUC was the metric selected for these experiments.

For all degrees of imbalance in the distribution some degradation in performance was observed. As expected, this is more pronounced at higher levels of imbalance. In this study, the following five strategies were analysed: random oversampling, SMOTE, borderline-SMOTE, ADASYN and Metacost. One of the main conclusions for highly imbalanced domains (1/99, 5/95 and 10/90) is the general failure of all considered strategies. SMOTE was found not to be so competitive as expected when compared to random oversampling. Moreover, the results obtained for borderline-SMOTE and ADASYN did not showed a clear advantage compared to standard SMOTE. Regarding Metacost, its performance was also quite poor when compared to the other strategies considered in the study.

López et al. [2013] compared three types of classifiers (SVM, decision tree and k-NN) on 66 data sets using the AUC metric. The approaches tested were clustered into: pre-processing (SMOTE, SMOTE+ENN, borderline-SMOTE, safe-level-SMOTE, ADASYN, SPIDER2 and DBsmote), cost-sensitive learning (Weighted-Classifier which simply introduces weights on the training set, Metacost, and the cost-sensitive classifier from the Weka environment) and ensemble-based techniques (AdaBoost-M1, AdaC2, RusBoost, smoteBagging and EasyEnsemble).

The main conclusions from this study were:

— regarding pre-processing strategies, SMOTE and SMOTE+ENN are the best performers; Borderline-SMOTE and ADASYN also present a robust performance on average;
— for the tested cost-sensitive learning methods, Metacost and Weighted-Classifier were the ones that presented the best performance;
— SmoteBagging was the best ensemble method tested; RusBoost and EasyEnsemble also performed well;
— For decision trees and k-NN, the best performing strategy was smoteBagging, while for SVMs SMOTE obtained the best performance closely followed by the remaining evaluated pre-processing strategies.

We must highlight that some results in López et al. [2013] disagree with the ones presented by Batista et al. [2012], in particular with respect to the Metacost approach. Another problem with these two latter works is the fact that both dropped from evaluation the random undersampling method which was shown to be quite competitive in other studies.

Recently, Stefanowski [2016] studied the impact of several data characteristics in the performance of both learning algorithms and pre-processing strategies. These data characteristics, called data difficulty factors, include the class overlap problem, the existence of small disjuncts and some characteristics of the minority class examples. Stefanowski [2016] proposes a categorization of the minority class cases with respect to their local characteristics into the following four types: safe, borderline, rare and outliers. Then, Stefanowski [2016] studies the relation between the dominant type of minority examples in a data set and both the performance obtained by several learning algorithms and pre-processing strategies.

As a final remark, we stress that in all these cases, only binary classification tasks have been considered and usually only one measure is used to assess the performance. This entails some limitations in the conclusions. Particularly, because it was shown that different assessment measures may provide different evaluation results (e.g. Van Hulse et al. [2007]). Moreover, these papers always assumed that the best is to perfectly balance the distribution which has also been shown not to be the most

favorable setting in terms of performance (e.g. Weiss and Provost [2003]; Khoshgoftaar et al. [2007]).

6. THEORETICAL ADVANCES

The problem of imbalanced domains is a relevant problem with important applications in a wide range of fields. The scientific community has been producing several approaches to this problem as we have surveyed in the previous sections. These proposals typically solve the problem in a particular domain or on a small set of tasks. However, many of the developed techniques fail under different imbalanced problems. An important question that arises then is: *Why* and *when* will a particular technique developed for the problem of imbalanced domains fail or succeed? The reasons behind this unstable behavior are not understood, and we believe that only with more efforts regarding the theoretical foundations of imbalanced domains we will be able to answer this question. The lack of a theoretical understanding of the problem is holding back the evolution of the solutions.

In spite of its relevance, the fact is that only a few theoretical contributions have been produced by the research community. While the range of approaches for handling imbalanced problems is increasing, the work on the theoretical foundations of the problem is scarce. We consider that one of the reasons for this is related with the lack of a precise definition of the problem, that includes the diversity of applications of imbalanced domains.

The lack of a precise definition of the problem frequently leads to some misconceptions being widely spread throughout the scientific community. One example is the equivalence between sampling methods and misclassification costs. This connection was first established by Breiman et al. [1984]. However, for real-world applications, Weiss [2013] has shown that the equivalence frequently does not hold. Consider, for instance, a binary classification problem with 1100 examples and an imbalanced domain with a class distribution of 10:1. This means that the positive class consists of 100 examples and the negative class is formed by 1000 cases. Let us set the cost of false negatives to 10 and the cost of false positives to 1. In this case, we have theoretically a situation of equivalence between the definition of misclassification costs and a balanced domain. A balanced domain could be obtained by under-sampling the negative class (multiplying it by $\frac{1}{10}$), or by over-sampling the minority class (multiplying it by 10). However, when performing under-sampling potentially useful data may be discarded and, when performing over-sampling there is the risk of overfitting if replicas are introduced. The equivalence would only hold if new minority class examples were available from the original distribution. Even the generation of synthetic examples from the minority class would not be sufficient to hold the equivalence because these new examples are not drawn from the original distribution and are only approximations of that distribution. This means that, the equivalence would only hold in real-world scenarios if new minority class examples were available for training. But, if this was possible, then the problem of imbalanced domains would not exist, because extra new data would be available as needed.

Regarding further theoretical contributions, we must highlight that this equivalence was further explored by Elkan [2001]. A theorem was proved, for binary classification tasks, that established a general formula regarding how to resample the negative class examples to obtain optimal cost-sensitive decisions using a standard non cost-sensitive learning algorithm. In spite of being more general, this formulation also suffers from the problems mentioned above on real-world applications.

A theoretical analysis of imbalance was presented by Wallace et al. [2011] and used to support a new proposal for tackling the problem of imbalanced domains. The analysis tries to answer a question raised by several researchers (e.g. Van Hulse et al. [2007])

and that is still not well understood: Why does under-sampling often presents a better performance when compared to other, sometimes more complex, techniques? The fact is that, empirically, under-sampling tends to outperform other approaches (ranging from simple random over-sampling to the generation of new synthetic examples). Still, several problems exist with random under-sampling strategy: it involves discarding potentially relevant information and it is a high-variance strategy. It is exactly by focusing on the latter problem that Wallace et al. [2011] proposed their solution: the use of bagging because it is a variance-reduction technique. The authors present a theoretical analysis and are able to establish the necessary and sufficient conditions for obtaining a suboptimal separator of the positive and negative distributions. Among other results, the authors show that by increasing the degree of imbalance there is a decrease in the probability of a weighted empirical cost minimization being effective. The theoretical framework developed justifies that, in the majority of imbalanced domains, the use of bagging with classifiers induced over balanced bootstrap sets is the best option.

More recently, Dal Pozzolo et al. [2015] also contributed for the theoretical advances regarding imbalanced domains. The focus of this work was also in the under-sampling strategy. In this paper, the authors study two aspects that are consequences of applying under-sampling: the potentially increase of variance (due to the reduction in the number of examples), and the warping effect produced in the posterior data distribution (due to the modification introduced in the prior probabilities). The first aspect may be addressed by averaging strategies to reduce the variability (as suggested by Wallace et al. [2011]), while for the second issue it is necessary to calibrate the probability of the new priors. Dal Pozzolo et al. [2015] analyse the interaction between under-sampling and the ranking error of the posterior probability, and the following formula was obtained:

$$\frac{\beta}{(p+\beta(1-p))^2} > \sqrt{\frac{v_s}{v}} \tag{45}$$

where β is the under-sampling rate, p is the posterior probability of the testing point and v and v_s are the variances of the classifier before and after sampling. If the formula is satisfied then under-sampling is effective. However, it is difficult to determine when the condition holds because it implies knowing the posterior probability and requires the estimation of the ratio of variances before and after under-sampling.

Still, this is an useful theoretical condition for understanding the under-sampling technique and some of the results obtained when applying it. In fact, the inequality of Equation 45 can explain why there are several contradictory results because it shows that there is a dependency between a good effect of under-sampling and some task related aspects (such as the degree of imbalance and the classifier variance).

In summary, it seems that the research community is finally understanding the importance of studying the theoretical foundations of the problem of imbalance domains. However, much remains to be done regarding theoretical foundations for this difficult problem, and easy heuristic solutions keep appearing at a fast rate.

7. PROBLEMS THAT HINDER PREDICTIVE MODELING UNDER IMBALANCED DOMAINS

In this section we describe some problems that frequently coexist with imbalanced domains and further contribute to degrade the performance of predictive models. These problems have been addressed mainly within a classification setting. Problems such as small disjuncts, class overlap and small sample size, usually coexist with imbalanced classification domains and are also identified as possible causes of classifiers perfor-

mance degradation [Weiss 2004; He and Garcia 2009; Sun et al. 2009; Stefanowski 2016].

We will briefly describe some works that address the relationship between imbalanced domains and the following problems: (i) class overlapping or class separability, (ii) small sample size and lack of density in the training set, (iii) high dimensionality of the data set, (iv) noisy data, (v) small disjuncts and (vi) data shift.

The overlap problem occurs when a given region of the data space contains an identical number of training cases of each class. In this situation, a learner will have an increased difficulty in distinguishing between the classes present on the overlapping region. In the last decade, some attention was given to the relationship between these two problems [Prati et al. 2004a; García et al. 2006]. The combination of imbalanced domains with overlapping regions causes much more difficulties than expected when considering their effects individually [Denil and Trappenberg 2010]. Recent works [Alejo Eleuterio et al. 2011; Alejo et al. 2013] presented combinations of solutions for handling, simultaneously, both the class imbalance and the class overlap problem.

The small sample problem is also related with imbalanced domains. In effect, having too few examples from the minority class will prevent the learner from capturing their characteristics and will hinder the generalization capability of the algorithm. The relation between imbalanced domains and small sample problems was addressed by Japkowicz and Stephen [2002] and Jo and Japkowicz [2004], where it was highlighted that minority class examples are easier to learn as their number increases.

The small sample problem may trigger problems such as rare cases [Weiss 2005], which bring an additional difficulty to the learning system. Rare examples are extremely scarce cases that are difficult to detect and use for generalization. The small sample problem may also be accompanied by a variable class distribution that may not match the target distribution.

Some imbalanced domains have a high number of predictor variables. The main challenge here is to adequately select features that contain the key information of the problem. Feature selection is recommended [Wasikowski and Chen 2010] and is also pointed as the solution for addressing the class imbalance problem [Mladenic and Grobelnik 1999; Zheng et al. 2004; Chen and Wasikowski 2008; Van Der Putten and Van Someren 2004; Forman 2003]. Several proposals exist for handling the imbalance problem in conjunction with the high dimensionality problem, all using a feature selection strategy [Zheng et al. 2004; Del Castillo and Serrano 2004; Forman and Cohen 2004; Chu et al. 2010]. In imbalanced domains, noisy data has a greater impact on the least represented classes [Weiss 2004]. Recently, Seiffert et al. [2011] concluded that, generally, class noise has a more significant impact on learners than imbalance. The interaction between the levels of imbalance and noise is a relevant issue and the two aspects should be studied together.

One of the most studied related problems is the problem of small disjuncts which is associated to the imbalance in the subclusters of each class in the data set [Japkowicz 2001; Jo and Japkowicz 2004]. When a subcluster has a low *coverage*, i.e., it classifies few examples, it is called small [Holte et al. 1989]. Small disjuncts are a problem because the learners are typically biased towards classifying large disjuncts and therefore they will tend to overfit and misclassify the cases in the small disjuncts. Due to the importance of these two problems, several works address the relation between the problem of small disjuncts and the class imbalance problem (e.g. Japkowicz [2003]; Weiss and Provost [2003]; Jo and Japkowicz [2004]; Pearson et al. [2003]; Japkowicz [2001]; Prati et al. [2004b]), although the connection between the two problems is not yet well understood [Jo and Japkowicz 2004]. Weiss [2010] analyses the impact of several factors on small disjuncts and in the error distribution across disjuncts. Pruning was not considered an effective strategy for dealing with small disjuncts in the pres-

ence of class imbalance [Prati et al. 2004b; Weiss 2010]. Weiss [2010] also concluded that even with a balanced data set, errors tend to be concentrated towards the smaller disjuncts. However, when there is class imbalance, the error concentration increases. Moreover, the increase in the class imbalance also increases the error concentration. Thus, class imbalance is partly responsible for the problem with small disjuncts, and artificially balancing the data distribution, causes a decrease in the error concentration.

The data shift problem has also deserved the attention of the research community. The problem of data shift occurs when there is a difference in the distribution of the train and test sets. The data shift occurs frequently, and it usually leads to a small performance degradation. However, on imbalanced domains severe performance losses may happen caused by this problem. López et al. [2013] mentions two different perspectives of this problem under imbalanced domains: intrinsic and induced data shift. The first one regards shifts in the data distribution that are already present in the data. This is an unexplored issue that still has no solution. As for induced data shift, it is related with the evaluation techniques used which may introduce this problem by themselves. Moreno-Torres et al. [2012] mentions that sample selection bias may occur due to a non-uniform random selection and this may produce the data shift problem. This may happen when using, for instance, the well known k-fold cross validation procedure. López et al. [2014] present a new validation procedure, named *distribution optimally balanced stratified cross-validation*, that tries to maintain the data distribution across all the partitions, trying to avoid inducing data shift.

The co-occurrence of the problems we have mentioned with imbalanced domains tends to further degrade the classifiers performance and therefore this relationship should not be ignored. We emphasize that these problems have been studied only in the context of classification tasks. It would be important to generalise these studies to regression tasks as these issues may also have a negative impact when happening in conjunction with imbalanced domains in these contexts.

8. CONCLUSIONS

Imbalanced domains pose important challenges to existing approaches to predictive modeling. In this paper we propose a formulation of the problem of predictive modeling with imbalanced data sets, including both classification and regression tasks. We present a survey of the state of the art solutions for obtaining and evaluating predictive models for both classification and regression tasks. We propose a new taxonomy for the existing approaches grouping them into: (i) data pre-processing, (ii) special-purpose learning methods, (iii) prediction post-processing and (iv) hybrid strategies.

For the last decade, the problem of predictive modeling under imbalanced domains has been focused on classification tasks. Existing proposals were developed specifically for classification problems, and existing surveys presented this topic only from a classification perspective. More recently, the research community started to address this problem within other contexts such as regression [Torgo et al. 2013], ordinal classification [Pérez-Ortiz et al. 2014], multi-label classification [Charte et al. 2015b], association rules mining [Luna et al. 2015], multi-instance learning [Wang et al. 2013b] and data streams [Wang and Abraham 2015]. It is now recognized that imbalanced domains are a broader and important problem posing relevant challenges in several contexts.

We present a summary of recent theoretical contributions on the study of imbalanced domains. This is certainly one of the most important open problems in this area. The relevance of the problem has pushed the community to provide an huge amount of heuristic solutions. Still, it is necessary to understand why, when and how they work, and to achieve this we need further theoretical advances.

We briefly describe some problems that are strongly related with imbalanced domains, highlighting works that explore the relationship of these other problems with imbalance data sets. The issue of the coexistence of other problems that may hinder the learners performance has been addressed solely for classification tasks and this is mostly an unexplored question for other tasks.

With the goal of understanding the current research directions in this area we identify a few recent trends:

— Wallace and Dahabreh [2012, 2014] have raised the issue of the reliability of probability estimates when using data sets with imbalanced domains. Although much was done for other domains, this had never been considered for the case of imbalanced domains. A proposal was presented for the assessment of this problem and an approach for solving it was also provided.
— Recently, a few papers have appeared that focus their contribution on the theoretical analysis of the properties of some approaches to imbalanced domains. This is a very important issue because it will provide a better understanding of the many existing approaches.
— Regarding performance assessment, the issue of correct experimental procedures for obtaining reliable estimates on data sets with imbalanced domains was recently raised [Japkowicz and Shah 2011; Raeder et al. 2012; López et al. 2014].
— The study of the problem of imbalanced domains has been extended to other data mining tasks. This is the case of regression tasks (e.g Torgo et al. [2013]), multiclass tasks (e.g. Alejo et al. [2014]; Fernández-Baldera et al. [2015]), learning from data streams (e.g. Ghazikhani et al. [2014]; Wang and Abraham [2015]), ordinal target variables (e.g.Baccianella et al. [2009]; Sánchez-Monedero et al. [2013]; Pérez-Ortiz et al. [2014]), multi-label classification (e.g. Tahir et al. [2012]; Charte et al. [2015b,a]), multi-instance learning (e.g. Wang et al. [2013b,a]) and mining association rules (e.g. Mangat and Vig [2014]; Luna et al. [2015]).

Finally, in terms of the open research issues within imbalanced domain problems, we consider the following to be the most relevant ones:

— Establishing the optimal way of translating the user preference biases into concrete settings of the different approaches to the problem (e.g. what is the right amount of under-sampling for some given user preferences?).
— More thorough and extensive experimental comparisons among the different approaches. Although some comparison studies exist, mainly for data pre-processing strategies within a classification setting, not much exists involving comparisons among the main different types of approaches (pre-processing, special-purpose learning methods, post-processing and hybrid). Moreover, there is still no comparison of the performance of the approaches across different task types (classification and regression).
— Creating a repository of benchmark data sets for this problem. In fact, although several open-access data set repositories exist, no collection of problems with imbalanced domains is currently available for the research community. This is an important issue whose resolution could provide a common baseline for comparison of different solutions in a fair and unified way [He and Ma 2013].
— Establishing what are the adequate metrics for evaluating and comparing different methods of addressing imbalanced domain problems. Currently, different papers select different metrics for comparing the methods, this being often the reason for some contradictory results.
— Further theoretical analysis of the existing proposals needs to be carried out. The knowledge about many of the existing approaches is still mostly based on collected

experimental evidence across some concrete data sets. Further understanding of the properties, advantages and limitations of the methods is necessary.

— Extension and/or development of approaches that can cope with other tasks apart from binary classification. Most of the existing work on imbalanced domains is focused on binary classification tasks. Recent studies have shown that similar imbalance problems exist in other tasks.

References

Rehan Akbani, Stephen Kwek, and Nathalie Japkowicz. 2004. Applying support vector machines to imbalanced datasets. In *Machine Learning: ECML 2004*. Springer, 39–50.

Roberto Alejo, J. A Antonio, Rosa Maria Valdovinos, and J. Horacio Pacheco-Sánchez. 2013. Assessments Metrics for Multi-class Imbalance Learning: A Preliminary Study. In *Pattern Recognition*. Springer, 335–343.

Roberto Alejo, Vicente García, and J. Horacio Pacheco-Sánchez. 2014. An Efficient Over-sampling Approach Based on Mean Square Error Back-propagation for Dealing with the Multi-class Imbalance Problem. *Neural Processing Letters* (2014), 1–15.

Roberto Alejo, Vicente García, José Martínez Sotoca, Ramón Alberto Mollineda, and José Salvador Sánchez. 2007. Improving the performance of the RBF neural networks trained with imbalanced samples. In *Computational and Ambient Intelligence*. Springer, 162–169.

Roberto Alejo, Rosa Maria Valdovinos, Vicente García, and J. Horacio Pacheco-Sanchez. 2013. A hybrid method to face class overlap and class imbalance on neural networks and multi-class scenarios. *Pattern Recognition Letters* 34, 4 (2013), 380–388.

Roberto Alejo Eleuterio, José Martínez Sotoca, Vicente García Jiménez, and Rosa María Valdovinos Rosas. 2011. Back propagation with balanced MSE cost Function and nearest neighbor editing for handling class overlap and class imbalance. (2011).

Josh Attenberg and Seyda Ertekin. 2013. Class Imbalance and Active Learning. In *Imbalanced learning: foundations, algorithms, and applications*, Haibo He and Yunqian Ma (Eds.). John Wiley & Sons.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2009. Evaluation measures for ordinal regression. In *Intelligent Systems Design and Applications, 2009. ISDA'09. Ninth International Conference on*. IEEE, 283–287.

Gaurav Bansal, Atish P. Sinha, and Huimin Zhao. 2008. Tuning data mining methods for cost-sensitive regression: a study in loan charge-off forecasting. *Journal of Management Information Systems* 25, 3 (2008), 315–336.

Ricardo Barandela, José Salvador Sánchez, Vicente Garcıa, and Edgar Rangel. 2003. Strategies for learning in class imbalance problems. *Pattern Recognition* 36, 3 (2003), 849–851.

Vincent Barnab-Lortie, Colin Bellinger, and Nathalie Japkowicz. 2015. Active Learning for One-class Classification. In *Proceedings of ICMLA'2015*.

Sukarna Barua, Monirul Islam, Xin Yao, and Kazuyuki Murase. 2012. MWMOTE-Majority Weighted Minority Oversampling Technique for Imbalanced Data Set Learning. (2012).

Guilherme Batista, Danilo Silva, and Ronaldo Prati. 2012. An Experimental Design to Evaluate Class Imbalance Treatment Methods. In *Machine Learning and Applications (ICMLA), 2012 11th International Conference on*, Vol. 2. IEEE, 95–101.

Gustavo EAPA Batista, Ronaldo C Prati, and Maria Carolina Monard. 2004. A study of the behavior of several methods for balancing machine learning training data. *ACM*

SIGKDD Explorations Newsletter 6, 1 (2004), 20–29.
Rukshan Batuwita and Vasile Palade. 2009. A New Performance Measure for Class Imbalance Learning. Application to Bioinformatics Problems. In *Machine Learning and Applications, 2009. ICMLA'09. International Conference on*. IEEE, 545–550.
Rukshan Batuwita and Vasile Palade. 2010a. Efficient resampling methods for training support vector machines with imbalanced datasets. In *Neural Networks (IJCNN), The 2010 International Joint Conference on*. IEEE, 1–8.
Rukshan Batuwita and Vasile Palade. 2010b. FSVM-CIL: fuzzy support vector machines for class imbalance learning. *Fuzzy Systems, IEEE Transactions on* 18, 3 (2010), 558–571.
Rukshan Batuwita and Vasile Palade. 2012. Adjusted geometric-mean: a novel performance measure for imbalanced bioinformatics datasets learning. *Journal of Bioinformatics and Computational Biology* 10, 04 (2012).
Colin Bellinger, Nathalie Japkowicz, and Christopher Drummond. 2015. Synthetic Oversampling for Advanced Radioactive Threat Detection. In *Proceedings ICML'2015*.
Colin Bellinger, Shiven Sharma, and Nathalie Japkowicz. 2012. One-Class versus Binary Classification: Which and When?. In *Machine Learning and Applications (ICMLA), 2012 11th International Conference on*, Vol. 2. IEEE, 102–106.
Jinbo Bi and Kristin P Bennett. 2003. Regression Error Characteristic Curves. In *Proc. of the 20th Int. Conf. on Machine Learning*. 43–50.
Jerzy Błaszczyński and Jerzy Stefanowski. 2015. Neighbourhood sampling in bagging for imbalanced data. *Neurocomputing* 150 (2015), 529–542.
Andrew P. Bradley. 1997. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern recognition* 30, 7 (1997), 1145–1159.
Paula Branco. 2014. *Re-sampling Approaches for Regression Tasks under Imbalanced Domains*. Master's thesis. Dep. Computer Science, Faculty of Sciences - University of Porto.
Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone. 1984. Classification and regression trees. Wadsworth & Brooks. *Monterey, CA* (1984).
Chumphol Bunkhumpornpat, Krung Sinapiromsaran, and Chidchanok Lursinsap. 2009. Safe-level-smote: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem. In *Advances in Knowledge Discovery and Data Mining*. Springer, 475–482.
Chumphol Bunkhumpornpat, Krung Sinapiromsaran, and Chidchanok Lursinsap. 2011. MUTE: Majority under-sampling technique. In *Information, Communications and Signal Processing (ICICS) 2011 8th International Conference on*. IEEE, 1–4.
Chumphol Bunkhumpornpat, Krung Sinapiromsaran, and Chidchanok Lursinsap. 2012. DBSMOTE: Density-based synthetic minority over-sampling technique. *Applied Intelligence* 36, 3 (2012), 664–684.
Chumphol Bunkhumpornpat and Sitthichoke Subpaiboonkit. 2013. Safe level graph for synthetic minority over-sampling techniques. In *Communications and Information Technologies (ISCIT), 2013 13th International Symposium on*. IEEE, 570–575.
Michael Cain and Christian Janssen. 1995. Real estate price prediction under asymmetric loss. *Annals of the Institute of Statistical Mathematics* 47, 3 (1995), 401–414.
Peng Cao, Dazhe Zhao, and Osmar R Zaïane. 2013. A PSO-Based Cost-Sensitive Neural Network for Imbalanced Data Classification. In *Trends and Applications in Knowledge Discovery and Data Mining*. Springer, 452–463.
Cristiano Leite Castro and Antônio de Pádua Braga. 2013. Novel Cost-Sensitive Approach to Improve the Multilayer Perceptron Performance on Imbalanced Data. *IEEE Trans. Neural Netw. Learning Syst.* 24, 6 (2013), 888–899.
Edward Y Chang, Beitao Li, Gang Wu, and Kingshy Goh. 2003. Statistical learning

for effective visual information retrieval.. In *ICIP (3)*. 609–612.
Francisco Charte, Antonio J Rivera, María J del Jesus, and Francisco Herrera. 2015a. Addressing imbalance in multilabel classification: Measures and random resampling algorithms. *Neurocomputing* 163 (2015), 3–16.
Francisco Charte, Antonio J Rivera, María J del Jesus, and Francisco Herrera. 2015b. MLSMOTE: Approaching imbalanced multilabel learning through synthetic instance generation. *Knowledge-Based Systems* 89 (2015), 385–397.
Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. P. Kegelmeyer. 2002. SMOTE: Synthetic Minority Over-sampling Technique. *JAIR* 16 (2002), 321–357.
Nitesh V. Chawla, David A. Cieslak, Lawrence O. Hall, and Ajay Joshi. 2008. Automatically countering imbalance and its empirical relationship to cost. *Data Mining and Knowledge Discovery* 17, 2 (2008), 225–252.
Nitesh V. Chawla, Lawrence O. Hall, and Ajay Joshi. 2005. Wrapper-based computation and evaluation of sampling methods for imbalanced datasets. In *Proceedings of the 1st international workshop on Utility-based data mining*. ACM, 24–33.
Nitesh V. Chawla, Nathalie Japkowicz, and Aleksander Kotcz. 2004. Editorial: special issue on learning from imbalanced data sets. *ACM SIGKDD Explorations Newsletter* 6, 1 (2004), 1–6.
Nitesh V. Chawla, Aleksandar Lazarevic, Lawrence O. Hall, and Kevin W. Bowyer. 2003. SMOTEBoost: Improving prediction of the minority class in boosting. In *Knowledge Discovery in Databases: PKDD 2003*. Springer, 107–119.
Chao Chen, Andy Liaw, and Leo Breiman. 2004. Using random forest to learn imbalanced data. *University of California, Berkeley* (2004).
Sheng Chen, Haibo He, and Edwardo A. Garcia. 2010. Ramoboost: Ranked minority oversampling in boosting. *Neural Networks, IEEE Transactions on* 21, 10 (2010), 1624–1642.
Xue-wen Chen and Michael Wasikowski. 2008. Fast: a roc-based feature selection metric for small samples and imbalanced data classification problems. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 124–132.
Peter F. Christoffersen and Francis X. Diebold. 1996. Further results on forecasting and model selection under asymmetric loss. *Journal of applied econometrics* 11, 5 (1996), 561–571.
Peter F. Christoffersen and Francis X. Diebold. 1997. Optimal prediction under asymmetric loss. *Econometric theory* 13, 06 (1997), 808–817.
Leilei Chu, Hui Gao, and Wenbo Chang. 2010. A new feature weighting method based on probability distribution in imbalanced text classification. In *Fuzzy Systems and Knowledge Discovery (FSKD), 2010 Seventh International Conference on*, Vol. 5. IEEE, 2335–2339.
Yu-Meei Chyi. 2003. Classification Analysis Techniques for Skewed Class Distribution Problems. *Master Thesis, Department of Information Management, National Sun Yat-Sen University* (2003).
David A. Cieslak and Nitesh V. Chawla. 2008. Learning decision trees for unbalanced data. In *Machine Learning and Knowledge Discovery in Databases*. Springer, 241–256.
David A. Cieslak, Thomas R. Hoens, Nitesh V. Chawla, and W Philip Kegelmeyer. 2012. Hellinger distance decision trees are robust and skew-insensitive. *Data Mining and Knowledge Discovery* 24, 1 (2012), 136–158.
Gilles Cohen, Mélanie Hilario, Hugo Sax, Stéphane Hugonnet, and Antoine Geissbuhler. 2006. Learning from imbalanced data in surveillance of nosocomial infection. *Artificial Intelligence in Medicine* 37, 1 (2006), 7–18.
Sven F. Crone, Stefan Lessmann, and Robert Stahlbock. 2005. Utility based data min-

ing for time series analysis: cost-sensitive learning for neural network predictors. In *Proceedings of the 1st international workshop on Utility-based data mining*. ACM, 59–68.

Andrea Dal Pozzolo, Olivier Caelen, and Gianluca Bontempi. 2015. When is undersampling effective in unbalanced classification tasks? In *Machine Learning and Knowledge Discovery in Databases*. Springer, 200–215.

Sophia Daskalaki, Ioannis Kopanas, and Nikolaos M. Avouris. 2006. Evaluation of classifiers for an uneven class distribution problem. *Applied Artificial Intelligence* 20, 5 (2006), 381–417.

Jesse Davis and Mark Goadrich. 2006. The relationship between Precision-Recall and ROC curves. In *ICML'06: Proc. of the 23rd Int. Conf. on Machine Learning (ACM ICPS)*. ACM, 233–240.

María Dolores Del Castillo and José Ignacio Serrano. 2004. A multistrategy approach for digital text categorization from imbalanced documents. *ACM SIGKDD Explorations Newsletter* 6, 1 (2004), 70–79.

Misha Denil and Thomas Trappenberg. 2010. Overlap versus imbalance. In *Advances in Artificial Intelligence*. Springer, 220–231.

Pedro Domingos. 1999. MetaCost: A General Method for Making Classifiers Cost-Sensitive. In *KDD'99: Proceedings of the 5th International Conference on Knowledge Discovery and Data Mining*. ACM Press, 155–164.

John Doucette and Malcolm I. Heywood. 2008. GP classification under imbalanced data sets: Active sub-sampling and AUC approximation. In *Genetic Programming*. Springer, 266–277.

Dennis J. Drown, Taghi M. Khoshgoftaar, and Naeem Seliya. 2009. Evolutionary sampling and software quality modeling of high-assurance systems. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 39, 5 (2009), 1097–1107.

Chris Drummond and Robert C Holte. 2000. Explicitly representing expected cost: An alternative to ROC representation. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 198–207.

Chris Drummond and Robert C. Holte. 2003. C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *Workshop on Learning from Imbalanced Datasets II*, Vol. 11. Citeseer.

James P. Egan. 1975. Signal detection theory and {ROC} analysis. (1975).

Charles Elkan. 2001. The Foundations of Cost-Sensitive Learning. In *IJCAI'01: Proc. of 17th Int. Joint Conf. of Artificial Intelligence*, Vol. 1. Morgan Kaufmann Publishers, 973–978.

Şeyda Ertekin. 2013. Adaptive Oversampling for Imbalanced Data Classification. In *Information Sciences and Systems 2013*. Springer, 261–269.

Şeyda Ertekin, Jian Huang, Leon Bottou, and Lee Giles. 2007b. Learning on the border: active learning in imbalanced data classification. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. ACM, 127–136.

Şeyda Ertekin, Jian Huang, and C Lee Giles. 2007a. Active learning for class imbalance problem. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 823–824.

Andrew Estabrooks and Nathalie Japkowicz. 2001. A mixture-of-experts framework for learning from imbalanced data sets. In *Advances in Intelligent Data Analysis*. Springer, 34–43.

Andrew Estabrooks, Taeho Jo, and Nathalie Japkowicz. 2004. A multiple resampling method for learning from imbalanced data sets. *Computational Intelligence* 20, 1 (2004), 18–36.

Tom Fawcett. 2006. An introduction to ROC analysis. *Pattern recognition letters* 27, 8 (2006), 861–874.

Alberto Fernández, María José del Jesus, and Francisco Herrera. 2010. On the 2-tuples based genetic tuning performance for fuzzy rule based classification systems in imbalanced data-sets. *Information Sciences* 180, 8 (2010), 1268–1291.

Alberto Fernández, Salvador García, María José del Jesus, and Francisco Herrera. 2008. A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets. *Fuzzy Sets and Systems* 159, 18 (2008), 2378–2398.

Antonio Fernández-Baldera, José M Buenaposada, and Luis Baumela. 2015. Multi-class Boosting for Imbalanced Data. In *Pattern Recognition and Image Analysis*. Springer, 57–64.

César Ferri, Peter Flach, José Hernández-Orallo, and Athmane Senad. 2005. Modifying ROC curves to incorporate predicted probabilities. In *Proceedings of the second workshop on ROC analysis in machine learning*. 33–40.

César Ferri, José Hernández-orallo, and Peter A Flach. 2011a. Brier curves: a new cost-based visualisation of classifier performance. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. 585–592.

César Ferri, José Hernández-Orallo, and Peter A Flach. 2011b. A coherent interpretation of AUC as a measure of aggregated classification performance. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. 657–664.

César Ferri, José Hernández-Orallo, and R Modroiu. 2009. An experimental comparison of performance measures for classification. *Pattern Recognition Letters* 30, 1 (2009), 27–38.

George Forman. 2003. An extensive empirical study of feature selection metrics for text classification. *The Journal of machine learning research* 3 (2003), 1289–1305.

George Forman and Ira Cohen. 2004. Learning from little: Comparison of classifiers given little training. In *Knowledge Discovery in Databases: PKDD 2004*. Springer, 161–172.

Mikel Galar, Alberto Fernández, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera. 2012. A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 42, 4 (2012), 463–484.

Mikel Galar, Alberto Fernández, Edurne Barrenechea, and Francisco Herrera. 2013. Eusboost: Enhancing Ensembles for Highly Imbalanced Data-sets by Evolutionary Undersampling. *Pattern Recognition* (2013).

Ming Gao, Xia Hong, Sheng Chen, Chris J Harris, and Emad Khalaf. 2014. PDFOS: PDF estimation based over-sampling for imbalanced two-class problems. *Neurocomputing* 138 (2014), 248–259.

Joaquín García, Salvador Derrac, Isaac Triguero, Cristobal J Carmona, and Francisco Herrera. 2012. Evolutionary-based selection of generalized instances for imbalanced classification. *Knowledge-Based Systems* 25, 1 (2012), 3–12.

Salvador García, José Ramón Cano, Alberto Fernández, and Francisco Herrera. 2006. A proposal of evolutionary prototype selection for class imbalance problems. In *Intelligent Data Engineering and Automated Learning–IDEAL 2006*. Springer, 1415–1423.

Salvador García and Francisco Herrera. 2009. Evolutionary undersampling for classification with imbalanced datasets: Proposals and taxonomy. *Evolutionary Computation* 17, 3 (2009), 275–306.

Vicente García, Roberto Alejo, José Salvador Sánchez, José Martínez Sotoca, and Ramón Alberto Mollineda. 2006. Combined effects of class imbalance and class overlap on instance-based classification. In *Intelligent Data Engineering and Automated*

Learning–IDEAL 2006. Springer, 371–378.
Vicente García, Ramón Alberto Mollineda, and José Salvador Sánchez. 2008. A New Performance Evaluation Method for Two-Class Imbalanced Problems. In *Structural, Syntactic, and Statistical Pattern Recognition*. Springer, 917–925.
Vicente García, Ramón Alberto Mollineda, and José Salvador Sánchez. 2009. Index of balanced accuracy: A performance measure for skewed class distributions. In *Pattern Recognition and Image Analysis*. Springer, 441–448.
Vicente García, Ramón Alberto Mollineda, and José Salvador Sánchez. 2010. Theoretical analysis of a performance measure for imbalanced data. In *Pattern Recognition (ICPR), 2010 20th International Conference on*. IEEE, 617–620.
Alireza Ghasemi, Mohammad T Manzuri, Hamid R Rabiee, Mohammad H Rohban, and Siavash Haghiri. 2011a. Active one-class learning by kernel density estimation. In *Machine Learning for Signal Processing (MLSP), 2011 IEEE International Workshop on*. IEEE, 1–6.
Alireza Ghasemi, Hamid R Rabiee, Mohsen Fadaee, Mohammad T Manzuri, and Mohammad H Rohban. 2011b. Active learning from positive and unlabeled data. In *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*. IEEE, 244–250.
Adel Ghazikhani, Reza Monsefi, and Hadi Sadoghi Yazdi. 2014. Online neural network model for non-stationary and imbalanced data stream classification. *International Journal of Machine Learning and Cybernetics* 5, 1 (2014), 51–62.
Clive W. Granger. 1999. Outline of forecast theory using generalized cost functions. *Spanish Economic Review* 1, 2 (1999), 161–173.
Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. 2005. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In *Advances in intelligent computing*. Springer, 878–887.
David J. Hand. 2009. Measuring classifier performance: a coherent alternative to the area under the ROC curve. *Machine learning* 77, 1 (2009), 103–123.
Peter. E. Hart. 1968. The condensed nearest neighbor rule. *IEEE Transactions on Information Theory* 14 (1968), 515–516.
Haibo He, Yang Bai, Edwardo A. Garcia, and Shutao Li. 2008. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*. IEEE, 1322–1328.
Haibo He and Edwardo A. Garcia. 2009. Learning from imbalanced data. *Knowledge and Data Engineering, IEEE Transactions on* 21, 9 (2009), 1263–1284.
Haibo He and Yunqian Ma. 2013. *Imbalanced learning: foundations, algorithms, and applications*. John Wiley & Sons.
José Hernández-Orallo. 2012. Soft (Gaussian CDE) regression models and loss functions. *arXiv preprint arXiv:1211.1043* (2012).
José Hernández-Orallo. 2013. {ROC} curves for regression. *Pattern Recognition* 46, 12 (2013), 3395 – 3411. DOI:http://dx.doi.org/10.1016/j.patcog.2013.06.014
José Hernández-Orallo. 2014. Probabilistic Reframing for Cost-Sensitive Regression. *ACM Trans. Knowl. Discov. Data* 8, 4, Article 17 (Aug. 2014), 55 pages. DOI:http://dx.doi.org/10.1145/2641758
José Hernández-Orallo, Peter Flach, and César Ferri. 2012. A unified view of performance metrics: Translating threshold choice into expected classification loss. *The Journal of Machine Learning Research* 13, 1 (2012), 2813–2869.
Robert C. Holte, Liane E. Acker, and Bruce W. Porter. 1989. Concept Learning and the Problem of Small Disjuncts.. In *IJCAI*, Vol. 89. Citeseer, 813–818.
Junjie Hu. 2012. Active learning for imbalance problem using L-GEM of RBFNN.. In *ICMLC*. 490–495.

Shengguo Hu, Yanfeng Liang, Lintao Ma, and Ying He. 2009. MSMOTE: improving classification performance when training data is imbalanced. In *Computer Science and Engineering, 2009. WCSE'09. Second International Workshop on*, Vol. 2. IEEE, 13–17.

Kaizhu Huang, Haiqin Yang, Irwin King, and Michael R. Lyu. 2004. Learning classifiers from imbalanced data based on biased minimax probability machine. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, Vol. 2. IEEE, II–558.

Jae Pil Hwang, Seongkeun Park, and Euntai Kim. 2011. A new weighted approach to imbalanced data classification problem via support vector machine with quadratic cost function. *Expert Systems with Applications* 38, 7 (2011), 8580–8585.

Tasadduq Imam, Kai Ming Ting, and Joarder Kamruzzaman. 2006. z-SVM: An SVM for improved classification of imbalanced data. In *AI 2006: Advances in Artificial Intelligence*. Springer, 264–273.

Nathalie Japkowicz. 2000. Learning from imbalanced data sets: a comparison of various strategies. In *AAAI workshop on learning from imbalanced data sets*, Vol. 68. Menlo Park, CA.

Nathalie Japkowicz. 2001. Concept-learning in the presence of between-class and within-class imbalances. In *Advances in Artificial Intelligence*. Springer, 67–77.

Nathalie Japkowicz. 2003. Class imbalances: are we focusing on the right issue. In *Workshop on Learning from Imbalanced Data Sets II*, Vol. 1723. 63.

Natalie Japkowicz. 2013. Assessment Metrics for Imbalanced Learning. In *Imbalanced learning: foundations, algorithms, and applications*, Haibo He and Yunqian Ma (Eds.). John Wiley & Sons.

Nathalie Japkowicz, Catherine Myers, and Mark Gluck. 1995. A novelty detection approach to classification. In *IJCAI*. 518–523.

Nathalie Japkowicz and Mohak Shah. 2011. *Evaluating learning algorithms: a classification perspective*. Cambridge University Press.

Nathalie Japkowicz and Shaju Stephen. 2002. The class imbalance problem: A systematic study. *Intelligent data analysis* 6, 5 (2002), 429–449.

Piyasak Jeatrakul, Kok Wai Wong, and Chun Che Fung. 2010. Classification of imbalanced data by combining the complementary neural network and SMOTE algorithm. In *Neural Information Processing. Models and Applications*. Springer, 152–159.

Taeho Jo and Nathalie Japkowicz. 2004. Class imbalances versus small disjuncts. *ACM SIGKDD Explorations Newsletter* 6, 1 (2004), 40–49.

Mahesh V. Joshi, Vipin Kumar, and Ramesh C. Agarwal. 2001. Evaluating boosting algorithms to classify rare classes: Comparison and improvements. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*. IEEE, 257–264.

Pilsung Kang and Sungzoon Cho. 2006. EUS SVMs: Ensemble of under-sampled SVMs for data imbalance problems. In *Neural Information Processing*. Springer, 837–846.

Taghi M Khoshgoftaar, Chris Seiffert, Jason Van Hulse, Amri Napolitano, and Andres Folleco. 2007. Learning with limited minority class data. In *Machine Learning and Applications, 2007. ICMLA 2007. Sixth International Conference on*. IEEE, 348–353.

Sotiris Kotsiantis, Dimitris Kanellopoulos, and Panayiotis Pintelas. 2006. Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering* 30, 1 (2006), 25–36.

Sotiris Kotsiantis and Panagiotis Pintelas. 2003. Mixture of expert agents for handling imbalanced data sets. *Annals of Mathematics, Computing & Teleinformatics* 1, 1 (2003), 46–55.

Miroslav Kubat, Robert C Holte, and Stan Matwin. 1998. Machine learning for the detection of oil spills in satellite radar images. *Machine learning* 30, 2-3 (1998), 195–215.

Miroslav Kubat and Stan Matwin. 1997. Addressing the Curse of Imbalanced Training Sets: One-Sided Selection. In *Proc. of the 14th Int. Conf. on Machine Learning*. Morgan Kaufmann, 179–186.

Jorma Laurikkala. 2001. *Improving identification of difficult small classes by balancing class distribution*. Springer.

Hyoung-joo Lee and Sungzoon Cho. 2006. The novelty detection approach for different degrees of class imbalance. In *Neural Information Processing*. Springer, 21–30.

Sauchi Stephen Lee. 1999. Regularization in skewed binary classification. *Computational Statistics* 14, 2 (1999), 277.

Sauchi Stephen Lee. 2000. Noisy replication in skewed binary classification. *Computational statistics & data analysis* 34, 2 (2000), 165–191.

Tae-Hwy Lee. 2008. Loss functions in time series forecasting. *International encyclopedia of the social sciences* (2008).

Chen Li, Chen Jing, and Gao Xin-tao. 2009. An improved P-SVM method used to deal with imbalanced data sets. In *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, Vol. 1. IEEE, 118–122.

Kewen Li, Wenrong Zhang, Qinghua Lu, and Xianghua Fang. 2014. An Improved SMOTE Imbalanced Data Classification Method Based on Support Degree. In *Identification, Information and Knowledge in the Internet of Things (IIKI), 2014 International Conference on*. IEEE, 34–38.

Peng Li, Pei-Li Qiao, and Yuan-Chao Liu. 2008. A hybrid re-sampling method for SVM learning from imbalanced data sets. In *Fuzzy Systems and Knowledge Discovery, 2008. FSKD'08. Fifth International Conference on*, Vol. 2. IEEE, 65–69.

Chun-Fu Lin and Sheng-De Wang. 2002. Fuzzy support vector machines. *Neural Networks, IEEE Transactions on* 13, 2 (2002), 464–471.

Alexander Liu, Joydeep Ghosh, and Cheryl E. Martin. 2007. Generative Oversampling for Mining Imbalanced Datasets.. In *DMIN*. 66–72.

Wei Liu, Sanjay Chawla, David A. Cieslak, and Nitesh V. Chawla. 2010. A Robust Decision Tree Algorithm for Imbalanced Data Sets.. In *SDM*, Vol. 10. SIAM, 766–777.

Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. 2009. Exploratory undersampling for class-imbalance learning. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* 39, 2 (2009), 539–550.

Yang Liu, Aijun An, and Xiangji Huang. 2006. Boosting prediction accuracy on imbalanced datasets with SVM ensembles. In *Advances in Knowledge Discovery and Data Mining*. Springer, 107–118.

Victoria López, Alberto Fernández, Salvador García, Vasile Palade, and Francisco Herrera. 2013. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences* 250 (2013), 113–141.

Victoria López, Alberto Fernández, and Francisco Herrera. 2014. On the importance of the validation technique for classification with imbalanced datasets: Addressing covariate shift when data is skewed. *Information Sciences* 257 (2014), 1–13.

José María Luna, Cristóbal Romero, José Raúl Romero, and Sebastián Ventura. 2015. An evolutionary algorithm for the discovery of rare class association rules in learning management systems. *Applied Intelligence* 42, 3 (2015), 501–513.

Tomasz Maciejewski and Jerzy Stefanowski. 2011. Local neighbourhood extension of SMOTE for mining imbalanced data. In *Computational Intelligence and Data Mining (CIDM), 2011 IEEE Symposium on*. IEEE, 104–111.

Satyam Maheshwari, Jitendra Agrawal, and Sanjeev Sharma. 2011. A New approach for Classification of Highly Imbalanced Datasets using Evolutionary Algorithms. *Intl. J. Sci. Eng. Res* 2 (2011), 1–5.

Marcus A Maloof. 2003. Learning when data sets are imbalanced and when costs are unequal and unknown. In *ICML-2003 workshop on learning from imbalanced data sets II*, Vol. 2. 2–1.

Larry Manevitz and Malik Yousef. 2002. One-class SVMs for document classification. *the Journal of machine Learning research* 2 (2002), 139–154.

Olvi L Mangasarian and Edward W Wild. 2001. Proximal support vector machine classifiers. In *Proceedings KDD-2001: Knowledge Discovery and Data Mining*. Citeseer.

Veenu Mangat and Renu Vig. 2014. Intelligent Rule Mining Algorithm for Classification over Imbalanced Data. *Journal of Emerging Technologies in Web Intelligence* 6, 3 (2014), 373–379.

Inderjeet Mani and Jianping Zhang. 2003. kNN approach to unbalanced data distributions: a case study involving information extraction. In *Proceedings of Workshop on Learning from Imbalanced Datasets*.

José Manuel Martínez-García, Carmen Paz Suárez-Araujo, and Patricio García Báez. 2012. SNEOM: a sanger network based extended over-sampling method. application to imbalanced biomedical datasets. In *Neural Information Processing*. Springer, 584–592.

David Mease, Abraham Wyner, and Andreas Buja. 2007. Cost-weighted boosting with jittering and over/under-sampling: JOUS-boost. *J. Machine Learning Research* 8 (2007), 409–439.

Giovanna Menardi and Nicola Torelli. 2010. Training and assessing classification rules with imbalanced data. *Data Mining and Knowledge Discovery* (2010), 1–31.

Charles E Metz. 1978. Basic principles of ROC analysis. In *Seminars in nuclear medicine*, Vol. 8. Elsevier, 283–298.

Ying Mi. 2013. Imbalanced Classification Based on Active Learning SMOTE. *Research Journal of Applied Sciences* 5 (2013).

Dunja Mladenic and Marko Grobelnik. 1999. Feature selection for unbalanced class distribution and naive bayes. In *ICML*, Vol. 99. 258–267.

Jose G Moreno-Torres, Troy Raeder, Rocío Alaiz-Rodríguez, Nitesh V Chawla, and Francisco Herrera. 2012. A unifying view on dataset shift in classification. *Pattern Recognition* 45, 1 (2012), 521–530.

Douglas Mossman. 1999. Three-way rocs. *Medical Decision Making* 19, 1 (1999), 78–89.

Satuluri Naganjaneyulu and Mrithyumjaya Rao Kuppa. 2013. A novel framework for class imbalance learning using intelligent under-sampling. *Progress in Artificial Intelligence* 2, 1 (2013), 73–84.

Munehiro Nakamura, Yusuke Kajiwara, Atsushi Otsuka, and Haruhiko Kimura. 2013. LVQ-SMOTE–Learning Vector Quantization based Synthetic Minority Over–sampling Technique for biomedical data. *BioData mining* 6, 1 (2013), 16.

Krystyna Napierała, Jerzy Stefanowski, and Szymon Wilk. 2010. Learning from imbalanced data in presence of noisy and borderline examples. In *Rough Sets and Current Trends in Computing*. Springer, 158–167.

Wing WY Ng, Jiankun Hu, Daniel S Yeung, Sha Yin, and Fabio Roli. 2014. Diversified Sensitivity-Based Undersampling for Imbalance Classification Problems. (2014).

Sang-Hoon Oh. 2011. Error back-propagation algorithm for classification of imbalanced data. *Neurocomputing* 74, 6 (2011), 1058–1061.

Ronald Pearson, Gregory Goney, and James Shwaber. 2003. Imbalanced clustering for microarray time-series. In *Proceedings of the ICML*, Vol. 3.

María Pérez-Ortiz, Pedro Antonio Gutiérrez, and César Hervás-Martínez. 2014. Projection-based ensemble learning for ordinal regression. *Cybernetics, IEEE Transactions on* 44, 5 (2014), 681–694.

Clifton Phua, Damminda Alahakoon, and Vincent Lee. 2004. Minority report in fraud

detection: classification of skewed data. *ACM SIGKDD Explorations Newsletter* 6, 1 (2004), 50–59.
Ronaldo C Prati, Gustavo EAPA Batista, and Maria Carolina Monard. 2004a. Class imbalances versus class overlapping: an analysis of a learning system behavior. In *MICAI 2004: Advances in Artificial Intelligence*. Springer, 312–321.
Ronaldo C Prati, Gustavo EAPA Batista, and Maria Carolina Monard. 2004b. Learning with class skews and small disjuncts. In *Advances in Artificial Intelligence–SBIA 2004*. Springer, 296–306.
Ronaldo C Prati, Gustavo EAPA Batista, and Diego F Silva. 2014. Class imbalance revisited: a new experimental setup to assess the performance of treatment methods. *Knowledge and Information Systems* (2014), 1–24.
Foster J Provost and Tom Fawcett. 1997. Analysis and Visualization of Classifier Performance: Comparison under Imprecise Class and Cost Distributions.. In *KDD*, Vol. 97. 43–48.
Foster J Provost, Tom Fawcett, and Ron Kohavi. 1998. The Case against Accuracy Estimation for Comparing Induction Algorithms. In *ICML'98: Proc. of the 15th Int. Conf. on Machine Learning*. Morgan Kaufmann Publishers, 445–453.
Troy Raeder, George Forman, and Nitesh V Chawla. 2012. Learning from imbalanced data: evaluation matters. In *Data mining: Foundations and intelligent paradigms*. Springer, 315–331.
Enislay Ramentol, Yailé Caballero, Rafael Bello, and Francisco Herrera. 2012a. SMOTE-RSB*: a hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using SMOTE and rough sets theory. *Knowledge and Information Systems* 33, 2 (2012), 245–265.
Enislay Ramentol, Nelle Verbiest, Rafael Bello, Yailé Caballero, Chris Cornelis, and Francisco Herrera. 2012b. SMOTE-FRST: a new resampling method using fuzzy rough set theory. In *10th International FLINS conference on uncertainty modelling in knowledge engineering and decision making (to appear)*.
Romesh Ranawana and Vasile Palade. 2006. Optimized Precision-A new measure for classifier performance evaluation. In *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on*. IEEE, 2254–2261.
Bhavani Raskutti and Adam Kowalczyk. 2004. Extreme re-balancing for SVMs: a case study. *ACM Sigkdd Explorations Newsletter* 6, 1 (2004), 60–69.
Rita P Ribeiro. 2011. *Utility-based Regression*. Ph.D. Dissertation. Dep. Computer Science, Faculty of Sciences - University of Porto.
Rita P Ribeiro and Luís Torgo. 2003. Predicting harmful algae blooms. In *Progress in Artificial Intelligence*. Springer, 308–312.
Cornelis V. Rijsbergen. 1979. Information Retrieval. Dept. of Computer Science, University of Glasgow, 2nd edition. (1979).
Juan J Rodríguez, José-Francisco Díez-Pastor, Jesús Maudes, and César García-Osorio. 2012. Disturbing Neighbors Ensembles of Trees for Imbalanced Data. In *Machine Learning and Applications (ICMLA), 2012 11th International Conference on*, Vol. 2. IEEE, 83–88.
José A Sáez, Julián Luengo, Jerzy Stefanowski, and Francisco Herrera. 2015. SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering. *Information Sciences* 291 (2015), 184–203.
Juan Pablo Sánchez-Crisostomo, Roberto Alejo, Erika López-González, Rosa María Valdovinos, and J Horacio Pacheco-Sánchez. 2014. Empirical Analysis of Assessments Metrics for Multi-class Imbalance Learning on the Back-Propagation Context. In *Advances in Swarm Intelligence*. Springer, 17–23.
Javier Sánchez-Monedero, Pedro Antonio Gutiérrez, and Cesar Hervás-Martínez.

2013. Evolutionary ordinal extreme learning machine. In *Hybrid Artificial Intelligent Systems*. Springer, 500–509.

Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. 2001. Estimating the support of a high-dimensional distribution. *Neural computation* 13, 7 (2001), 1443–1471.

Chris Seiffert, Taghi M Khoshgoftaar, Jason Van Hulse, and Andres Folleco. 2011. An empirical study of the classification performance of learners on imbalanced and noisy software quality data. *Information Sciences* (2011).

Chris Seiffert, Taghi M Khoshgoftaar, Jason Van Hulse, and Amri Napolitano. 2010. RUSBoost: A hybrid approach to alleviating class imbalance. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 40, 1 (2010), 185–197.

Shiven Sharma, Colin Bellinger, and Nathalie Japkowicz. 2012. Clustering based one-class classification for compliance verification of the comprehensive nuclear-test-ban treaty. In *Advances in Artificial Intelligence*. Springer, 181–193.

Atish P Sinha and Jerrold H May. 2004. Evaluating and tuning predictive data mining models using receiver operating characteristic curves. *Journal of Management Information Systems* 21, 3 (2004), 249–280.

Parinaz Sobhani, Herna Viktor, and Stan Matwin. 2014. Learning from Imbalanced Data Using Ensemble Methods and Cluster-Based Undersampling. In *New Frontiers in Mining Complex Patterns*. Springer, 69–83.

Marina Sokolova and Guy Lapalme. 2009. A systematic analysis of performance measures for classification tasks. *Information Processing & Management* 45, 4 (2009), 427–437.

Jie Song, Xiaoling Lu, and Xizhi Wu. 2009. An improved AdaBoost algorithm for unbalanced classification data. In *Fuzzy Systems and Knowledge Discovery, 2009. FSKD'09. Sixth International Conference on*, Vol. 1. IEEE, 109–113.

Panote Songwattanasiri and Krung Sinapiromsaran. 2010. SMOUTE: Synthetics Minority Over-sampling and Under-sampling TEchniques for class imbalanced problem. In *Proceedings of the Annual International Conference on Computer Science Education: Innovation and Technology, Special Track: Knowledge Discovery*. 78–83.

Jerzy Stefanowski. 2016. Dealing with Data Difficulty Factors While Learning from Imbalanced Data. In *Challenges in Computational Statistics and Data Mining*. Springer, 333–363.

Jerzy Stefanowski and Szymon Wilk. 2008. Selective pre-processing of imbalanced data for improving classification performance. In *Data Warehousing and Knowledge Discovery*. Springer, 283–292.

Yanmin Sun, Mohamed S Kamel, and Yang Wang. 2006. Boosting for learning multiple classes with imbalanced class distribution. In *Data Mining, 2006. ICDM'06. Sixth International Conference on*. IEEE, 592–602.

Yanmin Sun, Mohamed S Kamel, Andrew KC Wong, and Yang Wang. 2007. Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognition* 40, 12 (2007), 3358–3378.

Yanmin Sun, Andrew KC Wong, and Mohamed S Kamel. 2009. Classification of imbalanced data: A review. *International Journal of Pattern Recognition and Artificial Intelligence* 23, 04 (2009), 687–719.

Muhammad Atif Tahir, Josef Kittler, and Fei Yan. 2012. Inverse random under sampling for class imbalance problem and its application to multi-label classification. *Pattern Recognition* 45, 10 (2012), 3738–3750.

Aik Tan, David Gilbert, and Yves Deville. 2003. Multi-class protein fold classification using a new ensemble machine learning approach. (2003).

Yuchun Tang and Yan-Qing Zhang. 2006. Granular SVM with repetitive undersam-

pling for highly imbalanced protein homology prediction. In *Granular Computing, 2006 IEEE International Conference on*. IEEE, 457–460.

Yuchun Tang, Yan-Qing Zhang, Nitesh V. Chawla, and Sven Krasser. 2009. SVMs modeling for highly imbalanced classification. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* 39, 1 (2009), 281–288.

Dacheng Tao, Xiaoou Tang, Xuelong Li, and Xindong Wu. 2006. Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 28, 7 (2006), 1088–1099.

Nguyen Thai-Nghe, Zeno Gantner, and Lars Schmidt-Thieme. 2011. A new evaluation measure for learning from imbalanced data. In *Neural Networks (IJCNN), The 2011 International Joint Conference on*. IEEE, 537–542.

Ivan Tomek. 1976. Two modifications of CNN. *IEEE Trans. Syst. Man Cybern.* 11 (1976), 769–772.

Luís Torgo. 2005. Regression Error Characteristic Surfaces. In *KDD'05: Proc. of the 11th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*. ACM Press, 697–702.

Luís Torgo and Rita P Ribeiro. 2003. Predicting outliers. In *Knowledge Discovery in Databases: PKDD 2003*. Springer, 447–458.

Luís Torgo and Rita P Ribeiro. 2007. Utility-Based Regression. In *PKDD'07: Proc. of 11th European Conf. on Principles and Practice of Knowledge Discovery in Databases*. Springer, 597–604.

Luís Torgo and Rita P Ribeiro. 2009. Precision and Recall in Regression. In *DS'09: 12th Int. Conf. on Discovery Science*. Springer, 332–346.

Luís Torgo, Rita P Ribeiro, Bernhard Pfahringer, and Paula Branco. 2013. SMOTE for Regression. In *Progress in Artificial Intelligence*. Springer, 378–389.

Peter Van Der Putten and Maarten Van Someren. 2004. A bias-variance analysis of a real world learning problem: The CoIL challenge 2000. *Machine Learning* 57, 1-2 (2004), 177–195.

Jason Van Hulse, Taghi M Khoshgoftaar, and Amri Napolitano. 2007. Experimental perspectives on learning from imbalanced data. In *Proceedings of the 24th international conference on Machine learning*. ACM, 935–942.

Madireddi Vasu and Vadlamani Ravi. 2011. A hybrid under-sampling approach for mining unbalanced datasets: applications to banking and insurance. *International Journal of Data Mining, Modelling and Management* 3, 1 (2011), 75–105.

Nele Verbiest, Enislay Ramentol, Chris Cornelis, and Francisco Herrera. 2012. Improving SMOTE with Fuzzy Rough Prototype Selection to Detect Noise in Imbalanced Classification Data. In *Advances in Artificial Intelligence–IBERAMIA 2012*. Springer, 169–178.

Konstantinos Veropoulos, Colin Campbell, and Nello Cristianini. 1999. Controlling the sensitivity of support vector machines. In *Proceedings of the international joint conference on artificial intelligence*, Vol. 1999. Citeseer, 55–60.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *The Journal of Machine Learning Research* 11 (2010), 3371–3408.

Kiri L Wagstaff, Nina L Lanza, David R Thompson, Thomas G Dietterich, and Martha S Gilmore. 2013. Guiding Scientific Discovery with Explanations Using DEMUD.. In *AAAI*.

Byron C Wallace and Issa J Dahabreh. 2012. Class probability estimates are unreliable for imbalanced data (and how to fix them). In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*. IEEE, 695–704.

Byron C Wallace and Issa J Dahabreh. 2014. Improving class probability estimates for imbalanced data. *Knowledge and Information Systems* 41, 1 (2014), 33–52.

Byron C Wallace, Kevin Small, Carla E Brodley, and Thomas A Trikalinos. 2011. Class imbalance, redux. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*. IEEE, 754–763.

Benjamin X Wang and Nathalie Japkowicz. 2010. Boosting support vector machines for imbalanced data sets. *Knowledge and information systems* 25, 1 (2010), 1–20.

Heng Wang and Zubin Abraham. 2015. Concept Drift Detection for Imbalanced Stream Data. *arXiv preprint arXiv:1504.01044* (2015).

He-Yong Wang. 2008. Combination approach of SMOTE and biased-SVM for imbalanced datasets. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*. IEEE, 228–231.

Shuo Wang and Xin Yao. 2009. Diversity analysis on imbalanced data sets by using ensemble models. In *Computational Intelligence and Data Mining, 2009. CIDM'09. IEEE Symposium on*. IEEE, 324–331.

Xiaoguang Wang, Xuan Liu, Nathalie Japkowicz, and Stan Matwin. 2013a. Resampling and cost-sensitive methods for imbalanced multi-instance learning. In *Data Mining Workshops (ICDMW), 2013 IEEE 13th International Conference on*. IEEE, 808–816.

Xiaoguang Wang, Stan Matwin, Nathalie Japkowicz, and Xuan Liu. 2013b. Cost-sensitive boosting algorithms for imbalanced multi-instance datasets. In *Advances in Artificial Intelligence*. Springer, 174–186.

Mike Wasikowski and Xue-wen Chen. 2010. Combating the small sample class imbalance problem using feature selection. *Knowledge and Data Engineering, IEEE Transactions on* 22, 10 (2010), 1388–1400.

Deng Weiguo, Wang Li, Wang Yiyang, and Qian Zhong. 2012. An Improved SVM-KM Model for Imbalanced Datasets. In *Industrial Control and Electronics Engineering (ICICEE), 2012 International Conference on*. IEEE, 100–103.

Gary M Weiss. 2004. Mining with Rarity: a Unifying Framework. *SIGKDD Explorations Newsletter* 6, 1 (2004), 7–19.

Gary M Weiss. 2005. Mining with rare cases. In *Data Mining and Knowledge Discovery Handbook*. Springer, 765–776.

Gary M Weiss. 2010. The impact of small disjuncts on classifier learning. In *Data Mining*. Springer, 193–226.

Gary M Weiss. 2013. Foundations of Imbalanced Learning. In *Imbalanced learning: foundations, algorithms, and applications*, Haibo He and Yunqian Ma (Eds.). John Wiley & Sons.

Gary M Weiss and Foster J Provost. 2003. Learning when training data are costly: the effect of class distribution on tree induction. *J. Artif. Intell. Res.(JAIR)* 19 (2003), 315–354.

Cheng G Weng and Josiah Poon. 2008. A new evaluation measure for imbalanced datasets. In *Proceedings of the 7th Australasian Data Mining Conference-Volume 87*. Australian Computer Society, Inc., 27–32.

Gang Wu and Edward Y Chang. 2003. Class-boundary alignment for imbalanced dataset learning. In *ICML 2003 workshop on learning from imbalanced data sets II, Washington, DC*. 49–56.

Gang Wu and Edward Y Chang. 2005. KBA: Kernel boundary alignment considering imbalanced data distribution. *Knowledge and Data Engineering, IEEE Transactions on* 17, 6 (2005), 786–795.

Shaomin Wu, Peter Flach, and César Ferri. 2007. An improved model selection heuristic for AUC. In *ECML*. Springer, 478–489.

Jin Xiao, Ling Xie, Changzheng He, and Xiaoyi Jiang. 2012. Dynamic classifier ensemble model for customer classification with imbalanced class distribution. *Expert Systems with Applications* 39, 3 (2012), 3668–3675.

Li Xuan, Chen Zhigang, and Yang Fan. 2013. Exploring of clustering algorithm on class-imbalanced data. In *Computer Science & Education (ICCSE), 2013 8th International Conference on*. IEEE, 89–93.

Zeping Yang and Daqi Gao. 2012. An Active Under-Sampling Approach for Imbalanced Data Classification. In *Computational Intelligence and Design (ISCID), 2012 Fifth International Symposium on*, Vol. 2. IEEE, 270–273.

Show-Jane Yen and Yue-Shi Lee. 2006. Under-sampling approaches for improving prediction of the minority class in an imbalanced dataset. In *Intelligent Control and Automation*. Springer, 731–740.

Show-Jane Yen and Yue-Shi Lee. 2009. Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications* 36, 3 (2009), 5718–5727.

Yang Yong. 2012. The research of imbalanced data set of sample sampling method based on K-means cluster and genetic algorithm. *Energy Procedia* 17 (2012), 164–170.

Kihoon Yoon and Stephen Kwek. 2005. An unsupervised learning approach to resolving the data imbalanced issue in supervised learning problems in functional genomics. In *Hybrid Intelligent Systems, 2005. HIS'05. Fifth International Conference on*. IEEE, 6–pp.

Dai Yuanhong, Chen Hongchang, and Peng Tao. 2009. Cost-Sensitive Support Vector Machine Based on Weighted Attribute. In *Information Technology and Applications, 2009. IFITA'09. International Forum on*, Vol. 1. IEEE, 690–692.

Bianca Zadrozny, John Langford, and Naoki Abe. 2003. Cost-sensitive learning by cost-proportionate example weighting. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*. IEEE, 435–442.

Arnold Zellner. 1986. Bayesian estimation and prediction using asymmetric loss functions. *J. Amer. Statist. Assoc.* 81, 394 (1986), 446–451.

Dongmei Zhang, Wei Liu, Xiaosheng Gong, and Hui Jin. 2011. A novel improved SMOTE resampling algorithm based on fractal. *Journal of Computational Information Systems* 7, 6 (2011), 2204–2211.

Huaxiang Zhang and Mingfang Li. 2014. RWO-Sampling: A random walk oversampling approach to imbalanced data classification. *Information Fusion* 20 (2014), 99–116.

Huimin Zhao, Atish P Sinha, and Gaurav Bansal. 2011. An extended tuning method for cost-sensitive regression and forecasting. *Decision Support Systems* 51, 3 (2011), 372–383.

Zhaohui Zheng, Xiaoyun Wu, and Rohini Srihari. 2004. Feature selection for text categorization on imbalanced data. *ACM SIGKDD Explorations Newsletter* 6, 1 (2004), 80–89.

Zhi-Hua Zhou and Xu-Ying Liu. 2006. Training cost-sensitive neural networks with methods addressing the class imbalance problem. *Knowledge and Data Engineering, IEEE Transactions on* 18, 1 (2006), 63–77.

Jingbo Zhu and Eduard H Hovy. 2007. Active Learning for Word Sense Disambiguation with Methods for Addressing the Class Imbalance Problem.. In *EMNLP-CoNLL*, Vol. 7. 783–790.

Ling Zhuang and Honghua Dai. 2006a. Parameter estimation of one-class SVM on imbalance text classification. In *Advances in Artificial Intelligence*. Springer, 538–549.

Ling Zhuang and Honghua Dai. 2006b. Parameter optimization of kernel-based one-

class classifier on imbalance learning. *Journal of Computers* 1, 7 (2006), 32–40.