Benítez-Buenache, A., Álvarez-Pérez, L., Mathews, V. J. y Figuieras-Vidal, A.R. (2019). Likelihood ratio equivalence and imbalanced binary classification. *Expert Systems with Applications*, 130, pp. 84-96.

DOI:https://doi.org/10.1016/j.eswa.2019.03.050

Likelihood ratio equivalence and imbalanced binary classification

Alexander Benítez-Buenache [a, *], Lorena Álvarez-Pérez [a], V. John Mathews [b],
Aníbal R. Figueiras-Vidal[a]

[a] *Signal Theory and Communications Department, Universidad Carlos III de Madrid, Avda. de la Universidad, No. 30, Leganés, Madrid 28911, Spain* [b] *School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, USA*

abstract

This contribution proves that neutral re-balancing mechanisms, that do not alter the likelihood ratio, and training discriminative machines using Bregman divergences as surrogate costs are necessary and sufficient conditions to estimate the likelihood ratio of imbalanced binary classification problems in a consistent manner. These two conditions permit the estimation of the theoretical Neyman–Pearson operating characteristic corresponding to the problem under study. In practice, a classifier operates at a certain working point corresponding to, for example, a given false positive rate. This perspective allows the introduction of an additional principled procedure to improve classification performance by means of a second design step in which more weight is assigned to the appropriate training samples. The paper includes a number of examples that demonstrate the performance capabilities of the methods presented, and concludes with a discussion of relevant research directions and open problems in the area.

Keywords:
Imbalance
Bregman divergences
Likelihood ratio
Informed re-balancing
Sample emphasis

1. Introduction

Imbalanced classification problems are those in which class population sizes or misclassification losses, or both, are clearly different. Such problems have much relevance, and are fre- quent in practice. Consequently, there is a long list of works addressing different applications, including Rao, Krishnan, and Niculescu (2006), Mazurowski et al. (2008), Mena and González (2009), Freitas (2011) and Nahar, Imam, Tickle, and Chen (2013) in medicine, Radivojac, Chawla, Dunker, and Obradovic (2004), Batuwita and Palade (2009), Yu, Ni, and Zhao (2013) and Triguero et al. (2015) in bioinformatics, Viola and Jones (2004), Tao, Tang, Li, and Wu (2006), Kwak (2008), Chen, Fang, Huo, and Li (2011) and De la Torre, Granger, Sabourin, and Gorod- nichy (2015) in image processing and retrieval, Liao (2008), Park, Oh, and Pedrycz (2013) and Seiffert, Khoshgoftaar, Van Hulse, and Folleco (2014) in production processes, Chan and Stolfo (1998), Phua, Alahakoon, and Lee (2004), Tavallaee, Stakhanova, and Ghorbani (2010) and Mehrotra, Singh, Vatsa, and Majhi (2016) in security and safety, Liu, Hsu, and Ma (1999) and Zhou (2013) in business and finance, Manevitz and Yousef (2001) and Tong and Koller (2001) in text classification, Tsai, Chang, and Chiang (2009) in meteorology and González et al. (2013) in biology.

The most widely used classifiers, also referred to as discriminative machines - including those employing multi-layer perceptrons (MLPs) and radial basis function networks (RBFNs), support vector machines (SVMs), and the corresponding machine ensembles - are sensitive to imbalance because their parameter values are established by algorithms that try to optimize performance measures that do not consider imbalance effects. For example, typical sampled surrogate cost functions have minor contributions from minority classes in highly imbalanced data sets, and minimizing such cost functions leads to poor results for minority classes. This means that the performance of classifiers that are designed with just the imbalanced data will be far from acceptable. The purpose of re-balancing techniques is to reduce this undesirable effect. These techniques serve to reduce the tendency of the classification machines to decide in favor of the majority class by reducing the degree of imbalance in several forms, by means of creating a problem which is easier to solve. It is important to recognize that the solution obtained after re-balancing must permit to recover the solution to the original problem associated with the imbalanced observations.

Beginning from the late 1990s, many procedures have been proposed to reduce the difficulties associated with imbalanced classification problems. These procedures can be broadly divided into three families: data pre-processing techniques, modified learning algorithms, and hybrid methods. Because of space limitations, we

[*] Corresponding author.
E-mail addresses: abuenache@tsc.uc3m.es (A. Benítez-Buenache), lalvarez@tsc.uc3m.es (L. Álvarez-Pérez), mathews@oregonstate.edu (V.J. Mathews), arfv@tsc.uc3m.es (A.R. Figueiras-Vidal).

will restrict our discussion to those methods that are directly related to the approach we will introduce in this paper, and to references of their earliest publications or studies. The interested reader is referred to tutorials (Branco, Torgo, & Ribeiro, 2016; He & García, 2009; López, Fernández, García, Palade, & Herrera, 2013; Sun, Wong, & Kamel, 2009) and the monograph (He & Ma, 2013) for more details.

Data pre-processing techniques include cost-sensitive methods (Domingos, 1999; Elkan, 2001; Kukar & Kononenko, 1998; Zadrozny, Langford, & Abe, 2003) that increase the weight of minority samples during training. Other methods in this class of algorithms are sample selection (Kubat & Matwin, 1997), random re-sampling (under and/or oversampling) (Batista, Prati, & Monard, 2004; Estabrooks, Jo, & Japkowicz, 2004; Hido, Kashima, & Takahashi, 2009) and sample generation (Chawla, Bowyer, Hall, & Kegelmeyer, 2002; Lee, 1999; 2000) that re-balance the population sizes prior to training. Extensions of active learning algorithms (Settles, 2010) to data pre-processing-based imbalanced classifica- tion have also been reported (Abe, 2003; Bordes, Ertekin, Weston, & Bottou, 2005; Ertekin, Huang, Bottou, & Giles, 2007). These algo- rithms employ procedures to progressively select training samples with the purpose of improving training time and/or performance results.

Re-balancing can also be achieved by modifying the design parameters of some algorithms. For example, the form proposed in Veropoulos, Campbell, and Cristianini (1999) for SVMs simply weights the slack variables that quantify the deviation from the ideal correct classification more for the minority samples. Similar modifications have been applied to boosting ensembles (Fan, Stolfo, Zhang, & Chan, 1999; Sun, Kamel, Wong, & Wang, 2007; Ting, 2000). It is also possible to weight the terms of the SVM solution (Imam, Ting, & Kamruzzaman, 2006). These approaches are essentially the same as sample weighting. However, Masnadi-Shirazi and Vasconcelos (2010, 2011) proposed modified learning algorithms that are based on statistical analyses. Another class of imbalance oriented learning algorithms use one-class SVM classifiers, such as those proposed in Manevitz and Yousef (2001) and Kowalczyk and Raskutti (2002). These methods appear to be effec- tive in highly imbalanced situations. Several modifications of SVM kernels have been proposed to deal with the imbalance-related dif- ficulties (Fung & Mangasarian, 2005; Hong, Chen, & Harris, 2007; Wu & Chang, 2005; Yang, Yang, & Wang, 2009). Fuzzy SVMs have also been modified to deal with imbalance, Batuwita and Palade (2010) being an interesting example.

A vast majority of the procedures that have been proposed to deal with imbalanced problems have a “qualitative” nature, *i.e.*, they are reasonable modifications of the training data set or/and the classification algorithms, but there is not an analysis of why and how they provide their results. Although this does not decrease their usefulness, there is not always a clear perspective on their capabilities and limitations. As stated by the authors of He and García (2009), pp. 1279–1280) and Branco et al. (2016), pp. (31–33), a better understanding of the mechanisms of these meth- ods is needed in order to avoid mistakes and for further progress.

Based on this perspective, we will follow the direction of the few precedents that formally analyze the re-balancing approaches, including (Domingos, 1999) and the “metacost” framework, the anal- ysis of Elkan (2001) and Zadrozny et al. (2003) and the equiv- alence theorem, and the rigorous study presented in Castro and Braga (2013), that have largely inspired our research. In addition to developing new algorithms and perspectives, this work will also help advance a better understanding of the properties of different techniques (Dal Pozzolo, Caelen, & Bontempi, 2015; Wallace, Small, Brodley, & Trikalinos, 2011) and the effects of data characteristics on imbalanced classification algorithms (Stefanowski, 2016).

In this paper, we will consider discriminative machines that are based on nonlinear trainable transformations, such as MLPs, to study how to use them to solve imbalanced classification problems. This excludes SVMs, whose nonlinear transformations are implicit and originate the kernel form of the solution. The use of these machines for imbalanced problems is based on a completely different foundation, as we will briefly discuss below. In conceiving this analysis, we start from a fact which is typically not considered when designing imbalanced classifiers to optimize classical re-balancing measures (such as the Area Under the Curve, F-measures, etc.): A classification machine will operate at a given working point - for example, a given false positive rate, - and therefore, the design objective must focus on its performance at that point, and not in optimizing other metrics. Later, we will discuss how this perspective also has consequences for the so-called “informed” re-balancing mechanisms.

Although it is relatively straightforward to extend this study to multi-class problems, we will limit our analysis to binary problems for the sake of clarity. We will establish necessary and sufficient conditions to obtain a principled solution by focusing the attention on estimating the likelihood ratio, and establishing likelihood ratio equivalent problems according to the classical Bayes formulation. By “principled solution” we mean that this approach creates re-balanced versions of the problem under study that permit the recovery of a solution for the original imbalanced problem without imposing any constraint on the classification cost policy. That is, our approach allows the selection of the solution corresponding to any relative importance of both kinds of errors in the classifica- tion process.

The rest of the paper is organized as follows: Section 2 reviews the classical Bayes classification theory. Starting from this theory, the basic form of our re-balancing approach is introduced in Section 3. Section 4 discusses the implications of the principled approach of this paper. Section 5 presents some experimental work that supports this perspective. In Section 6, a two-step version of our approach with improved performances over those of one-step procedures is presented and compared with conventional informed re-balancing techniques. Section 7 presents some experimental re- sults on the two-step re-balancing procedure. Section 8 introduces a principled (likelihood ratio invariance-based) re-balancing proce- dure that pays attention to the most relevant samples, and weighs them in an appropriate manner. Some experimental results illus- trating the improvements provided by this approach are provided in Section 9.

The goal of the experimental work presented in this paper is only to demonstrate the strengths of the methods of the paper. As a result, we do not seek to evaluate the methods on benchmark imbalanced problems. Similarly, we do not present results of ex- tensive explorations of several design parameters with cross vali- dation.

Finally, the main conclusions of this work and a brief discussion of open research directions close the paper.

2. A brief overview of Bayes classification theory

The description in this section follows the presentation in the classic text (Van Trees, 1968).

Consider random observations $\mathbf{x}$ coming from one of two classes, C_1 (positive) and C_0 (negative). Assume that the *a priori* probabilities $\{P_i\}$ and the likelihoods $\{p(\mathbf{x}|C_i)\}, i \in \{0, 1\}$, are known, and that a cost policy $\{c_{ji}\}$, $i \in \{0, 1\}$ is defined for the classification problem. The cost policy indicates the cost of selecting j when i is true, and $c_{ji} > c_{ii}$. Minimizing the average classification cost leads to

$$q_L(\mathbf{x}) = \frac{p(\mathbf{x}|C_1)}{p(\mathbf{x}|C_0)} \underset{C_0}{\overset{C_1}{\gtrless}} \frac{c_{10} - c_{00}}{c_{01} - c_{11}} \frac{P_0}{P_1} = Q_c Q_P = Q \tag{1}$$

where $Q_P = P_0/P_1$ and $Q_c = (c_{10} - c_{00})/(c_{01} - c_{11})$, as well as their product Q, are non-negative values. Thus, the classification is carried out by comparing the likelihood ratio $q_L(\mathbf{x})$ with a threshold Q. The performance of this Bayes classifier is completely defined by the cost policy and the false alarm (false positive) and detection (true positive) probabilities given by

$$P_{FA} = Pr(decide\, C_1|C_0\; is\, true) = \int_Q^\infty p(q_L|C_0)\, dq_L \tag{2a}$$

$$P_D = Pr(decide\, C_1|C_1\; is\, true) = \int_Q^\infty p(q_L|C_1)\, dq_L \tag{2b}$$

respectively, where $\{p(q_L|C_i\ )\}$ are the likelihood functions for the random variable $q_L(\mathbf{x})$ (a function of the random variable $\mathbf{x}$).

Problems that have the same likelihood ratio but different cost or/and *a priori* probability ratios are solved using the same formula in (1), with different values of Q. Plotting P_D against P_{FA} as Q varies monotonically from ∞ to 0 results in a non-decreasing and convex curve called the receiver operating characteristic (ROC), going from (0,0) for $Q \to \infty$ to (1, 1) for $Q = 0$. Assuming a one-to-one ROC curve (*i.e.*, no jump at $P_{F\ A} = 0$, and no saturation at $P_D = 1$, *i.e.*, $P_D = 1$ is reached just when $P_{F\ A} = 1$), the value of Q for each working point can be directly obtained as the slope of the tangent of ROC at that point. Only this ROC, and not any other similar curve obtained by replacing $q_L\ (\mathbf{x})$ by some "reasonable" $q(\mathbf{x}) \neq q_L\ (\mathbf{x})$, provides the optimal solution for any choice of Q. This curve, called the Neyman–Pearson ROC (NP-ROC), allows the user to select any P_{FA} and to obtain the highest possible P_D, because this solution is based on (1), which constitutes the optimal solution for any choice of Q. Note that an estimate of the likelihood ratio permits one to obtain an estimate of the NP-ROC by moving Q from ∞ to 0.

It is straightforward to obtain the *a posteriori* probabilities from the likelihoods and the *a priori* probabilities as

$$Pr(C_i|\mathbf{x}) = \frac{p(\mathbf{x}|C_i)P_i}{p(\mathbf{x})} = \frac{p(\mathbf{x}|C_i)P_i}{p(\mathbf{x}|C_0)P_0 + p(\mathbf{x}|C_1)P_1} \tag{3}$$

An alternative form for the Bayes classifier can be obtained from (3) as

$$\frac{Pr(C_1|\mathbf{x})}{Pr(C_0|\mathbf{x})} \underset{C_0}{\overset{C_1}{\gtrless}} Q_c \tag{4a}$$

or, equivalently, because $Pr(C_0|\mathbf{x}) = 1 - Pr(C_1|\mathbf{x})$,

$$Pr(C_1|\mathbf{x}) \underset{C_0}{\overset{C_1}{\gtrless}} \frac{Q_c}{Q_c + 1} \tag{4b}$$

Let C_0 be the majority class. Imbalance occurs when $Q \gg 1$. This can happen when $Q_P \gg 1$ and/or $Q_c \gg 1$. We can easily apply (1) to imbalanced classification problems if $q_L\ (\mathbf{x})$ is known. This is also true for generative machines, *i.e.*, machines that work with estimates $\{\hat{p}\ (\mathbf{x}|C_i\ )\}$ and $\{\hat{P_i}\ \}$ obtained from a set of training samples. However, the performance of generative classifiers is usually worse than that of discriminative machines. Generative machines are trained just to obtain good likelihood estimates and not to address the classification problem. Their estimate of $q_L(\mathbf{x})$ is obtained by dividing likelihoods, a process prone to large numerical errors.

Therefore, we seek to answer the following fundamental question in this paper: Is it possible to use the classification criterion in (1) to define re-balancing mechanisms for discriminative classification machines? We focus on the formula in (1) that includes $q_L(\mathbf{x})$ rather than the formula (4b) because $Pr(C_1|\mathbf{x})$ depends on Q_P and, consequently, it changes if re-balance is applied. The answer is given in Section 3.

3. Foundations of the principled re-balancing approach

Discriminative nonlinear classification machines with trainable transformations, such as MLPs, are designed using an indirect approach. The classification decision takes the form

$$o(\mathbf{x}; \mathbf{w}^*) \underset{C_0}{\overset{C_1}{\gtrless}} 0 \tag{5}$$

where $o(\mathbf{x}; \mathbf{w})$ is a function whose trainable parameters $\mathbf{w}$ are optimized by finding[1]

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}} \sum_n c(t^{(n)}, o(\mathbf{x}^{(n)}; \mathbf{w})) \tag{6}$$

and $\{t\ ^{(n)}, \mathbf{x}^{(n)}\}, n = 1, \ldots, N$ are the available labeled training samples. Typically, $t = +1/-1$ for C_1/C_0, respectively, and $-1 \leq o \leq 1$. Here, $c(t,\ o)$ is an appropriate surrogate cost, which takes higher values when the values of t and o are more different. In addition to suffering from under/overfitting risks, this classifier is also imbalance sensitive, *i.e.*, its performance seriously degrades when $Q \gg 1$, because the minority samples have a very minor contribution to the sampled surrogate cost in (6), and, consequently, o does not consider them in an appropriate manner.

Two conditions are required to solve imbalanced problems using (1) by means of this kind of classification machines:

- *Neutral re-balancing, i.e.*, constructing a less imbalanced problem without (substantially) modifying the likelihood ratio. In other words, the re-balancing process must keep invariant the likelihood ratio. Section 4 contains additional discussion on re-balancing processes that are neutral.
- Application of a machine and, in particular, a training cost that makes possible the estimation of the true $q_L(\mathbf{x})$. This can be done by using *Bregman divergences* as surrogate costs.

Among machines with trainable transformations, MLPs have theoretically unlimited representational capabilities (Cybenko, 1989; Hornik, Stinchcombe, & White, 1989). The extremely powerful deep neural networks (DNNs) (Bengio, 2009; Deng & Yu, 2014; Schmidhuber, 2015) also fall into this category. Employing Bregman divergences (Bregman, 1967) is a necessary and sufficient condition to obtain estimates of the *a posteriori* probabilities, and consequently the likelihood ratio, from the output of those machines (Cid-Sueiro, Arribas, Urbán-Muñoz, & Figueiras-Vidal, 1999; Cid-Sueiro & Figueiras-Vidal, 2001).

For the binary case, a Bregman divergence is a function $c_B(t,\ o)$ such that

$$\frac{\partial\, c_B(t, o)}{\partial\, o} = -g(o)(t - o) \tag{7}$$

where $g(o) > 0$ is an arbitrary function. Examples of Bregman divergences include some well-known surrogate costs, such as the squared error $(t - o)^2$ and the (symmetric) entropy cost, which, for $t = \pm 1$ and $-1 \leq o \leq 1$, has the form

$$c_E(t, o) = -(1 + t)log_e(1 + o) - (1 - t)log_e(1 - o) \tag{8}$$

It is easy to show that applying a Bregman divergence in the theoretical version of $(6)^2$ leads to

$$o_B(\mathbf{x}) = E\{t|\mathbf{x}\} \tag{9}$$

where E indicates statistical average. If the surrogate cost is not a Bregman divergence, (9) is not true (see references above). The Appendix contains a proof of this necessary and sufficient characteristic of Bregman divergences.

[1] Q_c can be included by simply weighting the majority samples in (6): It is equivalent to a population imbalance.

[2] The theoretical version is $\mathbf{w}^* = \operatorname{argmin}_{\mathbf{w}} \int c(t, o) p(t|\mathbf{x}) dt$

In practice, a machine whose trainable parameters minimize the sampled version (6) will produce an estimate of this *a posteriori* expectations at the output. This estimate will become better with better representational capabilities of the machine and more training samples. For a more detailed discussion of Bregman divergences in the machine learning context, we refer the reader to Cid-Sueiro et al. (1999) and Cid-Sueiro and Figueiras-Vidal (2001).

For a binary classifier for which t takes the value from $+1$ and -1

$$E\{t|\mathbf{x}\} = 1\,Pr(C_1|\mathbf{x}) - 1\,Pr(C_0|\mathbf{x}) = 2\,Pr(C_1|\mathbf{x}) - 1 \tag{10}$$

resulting in the estimate

$$\tilde{Pr}(C_1|\mathbf{x}) = \frac{1}{2}(o_B(\mathbf{x}) + 1) \tag{11}$$

obtained from the output of the classification machine. Since, from (3),

$$Pr(C_1|\mathbf{x}) = \frac{q_L(\mathbf{x})}{q_L(\mathbf{x}) + Q_P} \tag{12}$$

including Q_c as stated in Footnote 1, introducing (11) in (12) for estimated values, we can arrive at

$$\tilde{q}_L(\mathbf{x}) = \tilde{Q}\frac{\tilde{Pr}(C_1|\mathbf{x})}{1 - \tilde{Pr}(C_1|\mathbf{x})} = \tilde{Q}\frac{1 + o_B(\mathbf{x})}{1 - o_B(\mathbf{x})} \tag{13}$$

where $\tilde{Q} = \tilde{Q}_P\, \tilde{Q}_c$, $\tilde{Q}_P$ is the ratio of the class populations corresponding to the situation in which $\tilde{Pr}$ $(C_1|\mathbf{x})$ is obtained - in general, the situation corresponding to a re-balancing process, - and $\tilde{Q}_c$ is the ratio of the sample weighting.

The estimate $\tilde{q}_L$ ($\mathbf{x}$) in (13) is obtained as a ratio of *a posteri- ori* probabilities. Algorithms with improved quality of the estimates are discussed in Sections 6 and 8.

The estimate $\tilde{q}_L$ ($\mathbf{x}$) obtained for a neutrally re-balanced version of the problem is also valid for the original imbalanced problem. Therefore, the classification decision can be made by comparing the right-hand side of (13) with the threshold on the right-hand side of (1). Additional manipulation will result in a decision crite- rion based on the output of the classifier machine as

$$o_B(\mathbf{x}) \underset{C_0}{\overset{C_1}{\gtrless}} \frac{\hat{Q} - \tilde{Q}}{\hat{Q} + \tilde{Q}} = \eta \tag{14}$$

where $\hat{Q}$ is the threshold for the original imbalanced problem, *i.e.*, $\hat{Q} = \hat{Q}_c\, \hat{Q}_P$, where $\hat{Q}_c$ is calculated with the original costs and $\hat{Q}_P$ estimated with the original populations. Note that the classification rule comes from the output of the machine trained under the re-balanced conditions, changing the decision threshold.

Formula (14) gives the (approximate) optimal solution to the original imbalanced problem using the output of the machine which has been trained to solve the neutrally-rebalanced classification problem. The quality of this solution depends on the quality of the estimate $\tilde{q}_L$ ($\mathbf{x}$) of the likelihood ratio given by (13). In general, this estimate will be better with more powerful machines, more samples, and in regions of the observation space with more samples, as well as for low-dimensional problems. In any case, this approach estimates the likelihood ratio, and not each one of the likelihoods separately as generative methods do.

The designer has complete freedom to select the value of $\hat{Q}$, *i.e.*, the cost policy, or, from another perspective, the working point in the NP-ROC, which is produced by changing the value of η from 1 to -1 in (14). If needed, we can compute $\hat{P}r$ $(C_1|\mathbf{x})$, the *a posteriori* probability of C_1 for the imbalanced problem, using (13) and (12).

The above analysis leads to conclude that the two conditions we mentioned at the beginning of this section are necessary and sufficient to train a machine that provides a principled estimate of the likelihood ratio of an imbalanced binary classification problem - and, consequently, an estimate of the NP-ROC corresponding to that problem - via defining an associated re-balanced problem. The process will be as follows:

- First, apply an appropriate neutral re-balancing, in order to obtain an easier classification problem with the same likelihood ratio.
- Second, train a classification machine using a Bregman divergence as surrogate cost, in order to obtain a principled estimate of the *a posteriori* class probabilities (according to (11)), and, subsequently, of the likelihood ratio (according to (13)).
- Finally, solve the original imbalanced problem by comparing the likelihood ratio estimate with an appropriately selected threshold. The best option to do this is to select a satisfactory working point in the NP-ROC estimate by moving the threshold value from ∞ to 0. The machine output can be also used, according to (14).

The above steps serve to estimate the theoretical Neyman–Pearson operating characteristic, which is optimal from the classification point of view. This is not the case, in general, of the operating characteristic which a machine provides when it is trained under other conditions. We are compensating the re-balance to come back to the solution of the original imbalanced problem in a principled manner, that requires neutrality and Bregman divergences.

When any of these conditions is not applied, the re-balancing process has not a principled base. The consequence is a degraded performance, that can be even worse than that of directly solving the imbalanced classification. We will provide illustrative examples in which these negative effects appear in Section 5, after a brief discussion on re-balancing methods and their neutrality in the next section.

4. A preliminary discussion

4.1. Neutral re-balancing

As mentioned above, a neutral re-balance must keep invariant the likelihood ratio. Now, we will enumerate the procedures that offer this characteristic:

- Uniformly weighting all the samples of a class does not alter the class probability density. Therefore, this kind of re-balancing methods is neutral. In fact, it is equivalent to modify the *a priori* class probability, and it appears as a change in the classification threshold.
- Re-sampling methods that select samples with equal probability are neutral in the average. Thus, they require train- ing machine ensembles with diverse re-sampled populations to be practically neutral. Then, they are equivalent to neu- tral sample weighting. This equivalence was first observed in Breiman, Friedman, Olshen, and Stone (1984). Although some practical differences have been observed when applying extreme versions of re-sampling (or generation) (Dal Pozzolo et al., 2015; Japkowicz & Stephen, 2002; Ling & Li, 1998; Wallace et al., 2011), the use of machine ensembles will in gen- eral increase the quality of the classification results. The well-known bootstrap techniques fall into the category of neutral re-sampling methods.
- Sample generation methods are also neutral in the average if the probability of generating new samples and the generation mechanisms are the same for all the class samples. One of them is the classical generation from a Parzen window density model (Parzen, 1962), which has been successfully applied in noisy learning. SMOTE (Chawla et al., 2002) generation is also approximately neutral because all the minority samples equally serve to generate new samples. As was the case for re-sampling, the use of machine ensembles helps to improve the performance of sample generation methods also.

These three families of neutral re-sampling procedures can be combined, reducing the risk of minor likelihood deformations while maintaining the diversity advantage. The best combination is problem-dependent. A full re-balance (with $\tilde{Q} = 1$) is usually not the best option (Khoshgoftaar, Seiffert, Van Hulse, Napolitano, & Folleco, 2007). A slightly imbalanced problem can be solved with- out difficulties, and the effects of population changes will be mod- erate.

4.2. Non-neutral re-balancing

Many re-balancing methods that do not keep neutrality have been proposed. These so-called "informed" methods pay more attention, by increasing the re-balance intensity, to regions of the observation space that are considered more critical for re-balancing purposes. These regions are determined according to the results of a relatively simple classifier such as a K nearest neighbors (K-NN) or a similar scheme that is directly applied to the training samples. Examples include informed direct oversampling (Japkowicz, 2000; Jo & Japkowicz, 2004) and undersampling (Jo & Japkowicz, 2004; Kubat & Matwin, 1997; Laurikkala, 2001) methods. The basic informed versions of SMOTE (Barua, Islam, Yao, & Murase, 2014; Han, Wang, & Mao, 2005; He, Bai, Garcia, & Li, 2008; Hu, Liang, Ma, & He, 2009) are not very different from these re-sampling techniques. Other versions of SMOTE include sample pre- or post-processing (Batista et al., 2004; Ramentol, Caballero, Bello, & Her- rera, 2012) and informed modifications of SMOTE generation char- acteristics (Bunkhumpornpat, Sinapiromsaran, & Lursinsap, 2009; 2012). There are also hybrid methods that combine several designs under different conditions (Provost & Fawcett, 2001).

There is much experimental evidence of the improvements that informed methods can provide. However, they suffer from a number of intrinsic limitations that result in some implicit risks in their direct application.

Applying a non-neutral re-balancing scheme modifies the likelihood ratio. As a result, a machine designed for the re-balanced problem cannot be modified in the principled form we presented in Section 3 to obtain an approximate Bayesian solution to the original imbalanced problem. Consequently, the ROC of that machine is not an estimate of the NP-ROC. Therefore, if we select a working point by modifying the threshold using the ROC of this machine, we incur the risk of obtaining suboptimal results.

It is possible to compensate for distortions in the likelihood ra- tio by modifying the distorted version in forms that can be ob- tained from an informed weighting mechanism. We will discuss this in detail in Section 6. But some risks remain. The sample weighting procedure determines what regions of the ROC will be better estimated. If the working point which is selected does not fall in such regions, the classification performance decreases, and the results may even be poorer than those of non-informed ap- proaches.

As a consequence of the above, it appears that a two-step process for applying informed re-balancing schemes is the appropriate way for designing classifiers for imbalanced data. The first step will use neutral re-balancing, thus allowing a reasonable determination of the working point in the resulting NP-ROC estimate. Once this working point is selected, an appropriate informed re-balancing scheme can be applied. If necessary, a likelihood ratio compensa- tion can be included. Section 6 introduces and discusses this two- step process in more detail.

4.3. Re-balancing with SVMs and ensembles

Support vector machines (SVMs) are designed under a hinge cost, which is not a Bregman divergence. Consequently, the approach of this paper cannot be applied to SVMs. They require a completely different analysis. They can be considered linear-in-the-trainable-parameters structures, and this leads to a solution of the form

$$\sum_{n^*} \alpha_{n^*} t^{(n^*)} k(\mathbf{x}, \mathbf{x}^{(n^*)}) = \boldsymbol{\beta}^T \mathbf{k}(\mathbf{x}) \underset{C_0}{\overset{C_1}{\gtrless}} U \qquad (15a)$$

for the basic SVM. Here, $k(.,.)$ is a Mercer kernel, and the (possibly sparse) weights $\{ \alpha_{n^*} \}$ are obtained by means of an optimization procedure to maximize the margin between training samples around the classification border, and $\{ \mathbf{x}^{(n^*)} \}$ are the support vec- tors. The criterion in (15a) is not a likelihood ratio classification formula. The left side is not necessarily a monotonically increasing function from 0 to ∞. However, it is possible to write

$$f(\boldsymbol{\gamma}^T \mathbf{k}(\mathbf{x})) \underset{C_0}{\overset{C_1}{\gtrless}} f(U') \qquad (15b)$$

where f is a nonlinear, strictly-increasing and, therefore, invertible function, f: $(min\{ \boldsymbol{\gamma}^T k(x)\}, max\{ \boldsymbol{\gamma}^T k(x)\}) \rightarrow (0, \infty)$. Formula (15b) is possible if, as is generally the case, the representational capability of the kernels is high enough. From this perspective, (15a) is similar to (15b). Applying f^{-1} to both sides of (15b) leads to

$$\boldsymbol{\gamma}^T \mathbf{k}(\mathbf{x}) \underset{C_0}{\overset{C_1}{\gtrless}} U' \qquad (15c)$$

and $\boldsymbol{\gamma}$, U', could be related to $\boldsymbol{\beta}$, U, if f were known.

A key problem is to select $k(.,.)$ in a manner that (15b) can be approximated for some f. Deeper analysis of this connection would likely provide insights into kernel selection or design, including the forms and modifications that have been proposed for imbalanced problems (Fung & Mangasarian, 2005; Hong et al., 2007; Wu & Chang, 2005; Yang et al., 2009). In any case, the linear character of (15a) means that SVMs are stable classifiers, and a consequence is their reduced sensitivity to imbalance difficulties (Japkowicz & Stephen, 2002). However, in spite of this advantage, the impossibility of applying the likelihood ratio equivalence subsists.

Similar to SVMs, some machine ensembles - see Galar, Fernandez, Barrenechea, Bustince, and Herrera (2012) for a thorough overview - either cannot be included in the likelihood ratio equivalence framework. For example, boosting ensembles aggregate their learners by means of a non-Bregman cost, implying that the likelihood ratio equivalence is not applicable in this case.

5. Some illustrative examples

The purpose of the examples in this section is to illustrate the consequences of applying non-neutral re-balancing techniques and non-Bregman surrogate costs to solve imbalanced classification problems. Additional discussions about optimizing the classifier and more examples will be provided after a more rigor- ous discussion in Section 6 of the two-step process introduced in Section 4.

5.1. Effects of a non-neutral re-balance

This example shows the distortion in the estimate of the likelihood ratio which can appear when informed, non-neutral re-sampling techniques are applied. Subsequently, the ROC of the resulting classifier differs from the NP-ROC. We choose an easy synthetic problem whose NP-ROC can be analytically established.

5.1.1. Problem description

Let the class likelihoods be

$$p(\mathbf{x}|C_1) = \begin{cases} \frac{1}{4}(1 + x_2), & x_1, x_2 \in [-1, 1] \\ 0, & \text{otherwise} \end{cases} \qquad (16a)$$

$$p(\mathbf{x}|C_0) = \begin{cases} \frac{1}{3.6}, & x_1 \in [-0.9, 0.9], x_2 \in [-1, 1] \\ 0, & \text{otherwise} \end{cases} \tag{16b}$$

For these probability densities, it follows immediately that the detection probability P_D and the false alarm probability P_{FA} are related by the following relationship:

$$P_D = 0.775 - 0.45(1 - 2P_{FA}) - 0.225(1 - 2P_{FA})^2 \tag{17}$$

This represents the analytical expression of the NP-ROC, which we consider as a reference.

We consider training the classifier with $N_1 = 300$ and $N_0 = 19200$ samples, generated according to (16a) and (16b) for classes C_1 and C_0, respectively. Thus, the imbalance ratio N_0/N_1 is 64.

5.1.2. Classification machines

We employ MLPs with a single hidden layer with $H = 4$ hidden neurons, a number which is appropriate for dealing with the complexity of the problem. The machine was trained using the back propagation (BP) algorithm, with a learning rate $\mu = 10^{-3}$. The input and output weights were initialized with Gaussian values having a zero mean value and variances $\sigma_1^2 = 1/\sqrt{(D+1)H}$ and $\sigma_2^2 = 1/\sqrt{1+H}$, respectively, where D is the input dimension ($D = 2$). The surrogate cost was the sum of the squared errors between the target values and the corresponding machine outputs. Finally, to avoid instability effects and to work with a reasonably powerful classifier, we used an ensemble of $M = 11$ independently trained MLPs with different sample populations (sample generation is applied) whose outputs were averaged.

5.1.3. Re-balancing methods

The first re-balancing scheme we applied here was the SMOTE (Chawla et al., 2002), which is an approximately neutral method. It generates minority samples randomly along the connection of each training sample with each of its K Nearest Neighbors (NNs) of the same class, the examples and its neighbors being also randomly selected. The parameter $K = 3$ was used in this example.

Borderline-SMOTE (B-SMOTE) (Han et al., 2005) is a non-neutral re-balancing generation method. B-SMOTE works like SMOTE, but only with those training samples “in danger”, *i.e.*, sam- ples that have at least one half of their m NNs belonging to the majority class. It also excludes the minority samples whose m NNs are majority samples, interpreting them as noisy cases. For large values of m, B-SMOTE works similarly to SMOTE. In this example, we maintain $K = 3$ and we consider $m = 3$ and $m = 7$, to check what the consequences of the separation from SMOTE are.

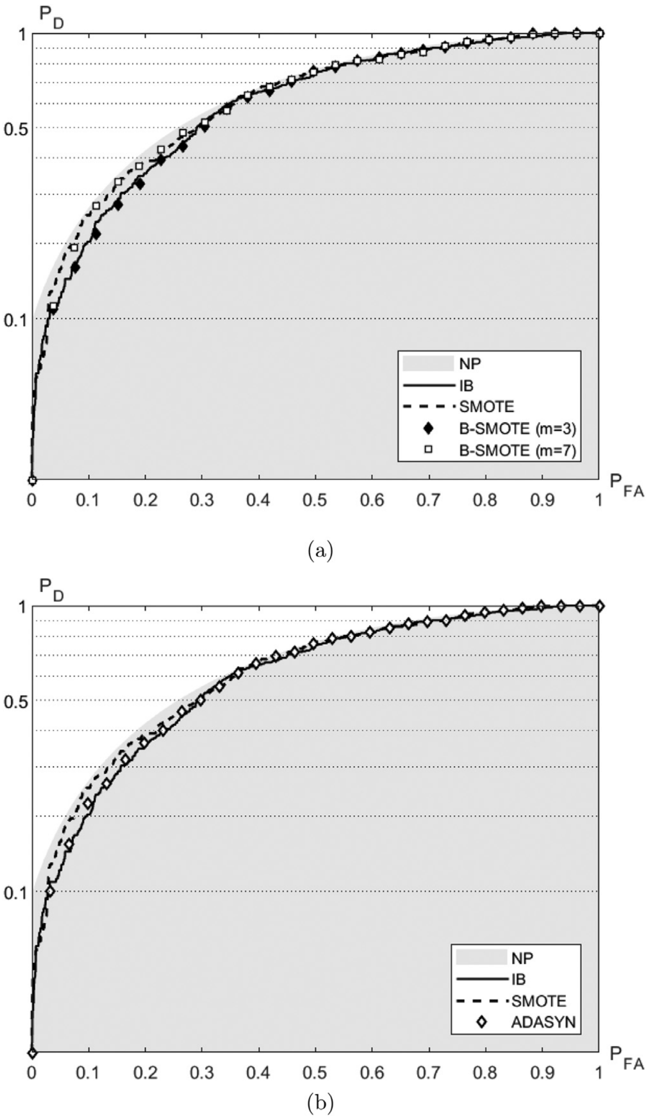
ADASYN (He et al., 2008) is also a non-neutral re-balancing method. It generates samples for the minority class according to the SMOTE procedure, but proportional to the number of majority samples which the training sample has among its K' NNs. Here, we will use $K' = K$, and $K = 3$ for generation.

Full re-balancing was imposed with $\tilde{Q} = 1$ for all cases considered.

To obtain the ROC estimates for the different designs, a new sample set was generated for testing the classifiers, with the size and *IR* of the test set being the same as those of the training set. This size was enough to obtain reasonably accurate estimates. The ROC estimates were established by sorting samples according to their output values, and moving the threshold from ∞ to 0.

5.1.4. Results and discussion

Fig. 1 shows the ROCs for different sample generation procedures, as well as the theoretical NP-ROC. P_D is presented along a nonlinear scale to make perceiving the differences easier. The ROCs were obtained by sorting the machine outputs for the test samples



(a)

(b)

Fig. 1. ROCs for the classifiers with non-neutral re-balancing. The border of the shadowed area corresponds to the NP-ROC. The other ROCs displayed are for IB (without re-balancing), SMOTE, and B-SMOTE ($m = 3$), B-SMOTE ($m = 7$) in (a) and ADASYN ($K' = 3$) in (b).

for each value of η - see formula (14, - and subsequently comput- ing the P_D and P_{FA} values.

It can be observed that there is a degradation for all the curves for low values of P_{FA}. When P_{FA} increases, both SMOTE and B-SMOTE with $m = 7$ (*i.e.*, the case in which more minority samples are selected) tend to the NP-ROC, while the direct design and B-SMOTE with $m = 3$ remain more degraded until (approximately) $P_{F\ A} = 0.\ 3$. The performance degradation of B-SMOTE with $m = 3$ (and the direct design) is more than 0.05 for $0.1 < P_{FA} < 0.2$, and note that B-SMOTE with $m = 3$ is even slightly worse than a direct design in this interval, *i.e.*, the performance obtained with the imbalanced dataset, indicated as IB in Fig. 1a and b, is poor, as expected. ADASYN works in a similar way, offering results not very far from those of a direct imbalanced training. We have checked that this is also the case of other informed re-balancing schemes, with minor differences. These degradations demonstrate that the performance resulting from applying a non-neutral re-balance can be substantially worse.

We have analyzed other data sets with different numbers of samples and different imbalance ratios that showed similar qualitative effects. However, such degradations have also been observed for smaller data sets and/or very high imbalance ratios. This might make the relative differences in the performance of neutral and non-neutral re-balancing less relevant.

5.2. Effects of non-Bregman surrogate costs

For these examples, we selected the Electricity data set from the OpenML data set repository (Harries, Gama, & Bifet, 2009). This is a large, (almost) balanced data set, which will help obtain a good estimate of the NP-ROC by processing the training data without modifications. Each sample in the data set consists of 8 features. The target is an increase or decrease in electricity price. The data set consists of $N_0 = 26075$ and $N_1 = 19237$ samples corresponding to increasing and decreasing prices.

To estimate the NP-ROC, all the majority and minority samples are randomly partitioned into training and test sets, with 80% of both classes going to the training set and the remaining 20% becoming the test set. The NP-ROC was estimated from the test data as explained earlier (*i.e.,* by sorting outputs of the test set) after designing an appropriate classifier using the training data.

5.2.1. Problems

We study two problems designed by randomly sub-sampling the original dataset, but with $IR \approx 50$. The first problem was based on $N_1 = 261$ and $N_0 = 13037$ samples, and the second problem involved a smaller data set with $N_1 = 33$ and $N_0 = 1629$ samples. Test sets with the same IR were built from the rest of the full data set.

5.2.2. Classification machine

We again employed a classifier based on an MLP architecture with $H = 7$ hidden neurons, which was found to be more appropriate for these problems. We use a single machine, which was enough to get good results.

5.2.3. Re-balancing methods

We use SMOTE ($K = 3$) in these examples, as well as a pure cost-sensitive full re-balancing, *i.e.*, including a cost factor equal to IR for the minority samples in the surrogate cost.

5.2.4. Surrogate costs

The convex combination

$$c(t, o) = \alpha (t - o)^2 + (1 - \alpha)|t - o| \tag{18}$$

$0 \leq \alpha \leq 1$, was employed as the surrogate cost for training the MLPs. When $\alpha = 1$, we have the squared error, which is a Bregman cost. When $\alpha \neq 1$, the cost is non-Bregman, and it differs from the squared error more and more as α decreases, until arriving at the absolute error when $\alpha = 0$. Performance differences must occur accordingly.

5.2.5. Results and discussion

Fig. 2 presents the results for the examples of this subsection. [3] In all the cases, the degradation when α decreases is obvious, making the need of a Bregman surrogate cost for principled re-balancing clearly evident. In addition, these degradations were higher when the training set was smaller, and SMOTE gave better results than the pure cost-sensitive re-balancing. All these results agree with expectations.

6. A two-step weighting procedure

6.1. Compensating non-neutral re-balancing

We will present here a method to compensate for the detrimental effects of non-neutral re-balancing in order to obtain accurate estimates of the NP-ROC. As we will see in Sub-section 6.2, this analysis will allow the introduction of truly informed re-balancing procedures that first estimate what samples are near the desired working border, and, after it, apply a non-neutral re-balancing step which puts more emphasis on these samples. This produces a new ROC estimate that is better around the working point, although it becomes worse in other regions. Obviously, this produces a beneficial effect.

Typically, the effect of an informed re-balancing method on the classifier design is to weight the minority samples in the optimization problem according to a factor $e_1(\mathbf{x})$ (re-sampling or generation). The weight function $e_1(\mathbf{x})$ depends on the training sample $\mathbf{x}$, and is typically larger when $\mathbf{x}$ is in the region that the particular method selects. Here, we will weight $\mathbf{x}$ more when it is close to the classification border corresponding to the working conditions. These conditions are determined by selecting a value for the P_{FA}, P_{FAW}, on an estimate of the NP-ROC (which is provided by a first step using a neutral re-balancing mechanism). This factor is established by means of a preliminary classification mechanism. We will also include in this analysis a possible factor $e_0(\mathbf{x})$ for the majority samples. We assume that these factors are not zero for any value of $\mathbf{x}$. Thus, the theoretical likelihood for the re-balanced problem is

$$q'_L(\mathbf{x}) = \frac{p(\mathbf{x}|C_1)e_1(\mathbf{x})/a_1}{p(\mathbf{x}|C_0)e_0(\mathbf{x})/a_0} = \frac{q_L(\mathbf{x})q_E(\mathbf{x})}{Q_A} \tag{19}$$

where $q_E(\mathbf{x}) = e_1(\mathbf{x})/e_0(\mathbf{x})$, $Q_A = a_1/a_0$, and a_1 and a_0 are normalization constants. Estimation of these normalization parameters is discussed later. Note that $e_i(\mathbf{x})$ changes the likelihood profiles, and that we are considering the resulting functions as new likelihoods, normalizing them to unity volume.

Now, we will see how to recover in a consistent manner an estimate of $q_L(\mathbf{x})$ from the estimate of $q'_L(\mathbf{x})$ obtained by an informed re-balancing process.

To simplify the initial discussion, let us consider a classifier that is trained using a cost-sensitive informed re-balancing approach employing a Bregman surrogate cost. In this case, we can estimate $q'_L(\mathbf{x})$ using (13) as

$$\hat{q}'_L(\mathbf{x}) = \frac{\hat{q}_L(\mathbf{x})q_E(\mathbf{x})}{\hat{Q}_A} = \tilde{Q}_2(\mathbf{x})\frac{1 + o_{B2}(\mathbf{x})}{1 - o_{B2}(\mathbf{x})} \tag{20}$$

If, as is usually the case, the classifier is trained with the weighted samples to minimize the error probability, $\tilde{Q}_2(\mathbf{x}) = q_E(\mathbf{x})$, and

$$\frac{1 + o_{B2}(\mathbf{x})}{1 - o_{B2}(\mathbf{x})} \underset{C_0}{\overset{C_1}{\gtrless}} \frac{\hat{Q}}{\hat{Q}_A} \tag{21}$$

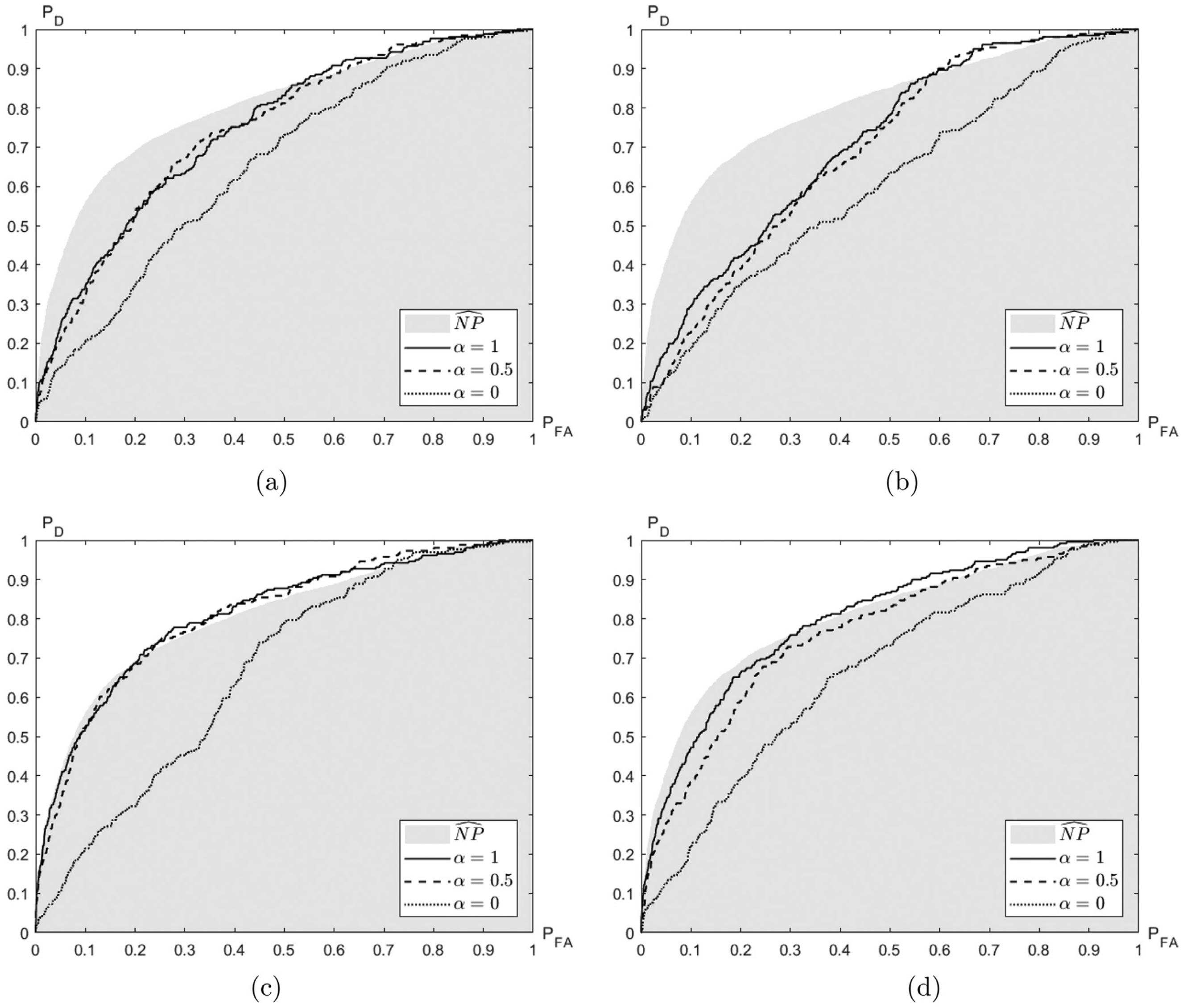
As before, we can compare directly with $o_{B2}(\mathbf{x})$ to find an equivalent test:

$$o_{B2}(\mathbf{x}) \underset{C_0}{\overset{C_1}{\gtrless}} \frac{\hat{Q} - \hat{Q}_A}{\hat{Q} + \hat{Q}_A} = \eta_2 \tag{22}$$

Formula (22) serves to estimate the classifier's NP-ROC by sorting the output values and moving η_2 from 1 to −1. A previously estimated NP-ROC (by applying a neutral re-balance) and the corresponding sample sorting will guide the choice of appropriate weights $e_i(\mathbf{x})$, according to the position of the working point (P_{FAW}).

Implementing the above system requires estimation of the normalization parameters a_1 and a_0. A straightforward way is to use

[3] Some machine performance curves are above the estimated NP ROCs in some regions because of sampling effects.



(a) (b) (c) (d)

Fig. 2. ROCS of the classifiers for non-Bregman costs. The border of the shaded region shows the NP-ROC estimated from the original Electricity data set (Harries et al., 2009). The values of α correspond to versions of surrogate cost (18). (a) First problem, cost-sensitive re-balancing; (b) Second (smaller) problem, cost-sensitive re-balancing;(c) First problem, SMOTE re-balancing; (d) Second problem, SMOTE re-balancing.

sample-based estimates for a given weighting scheme as

$$\hat{a}_i = \frac{1}{N_i} \sum_{n_i} e_i(\mathbf{x}^{(n_i)}) \tag{23a}$$

for $i = 0$ and 1, where $\mathbf{x}^{(n_i)}$, N_i, are the training samples and their number for the i^{th} class, respectively.

Re-sampling or/and generation can also be applied, but they only permit integer weights. In such cases, diversity, provided, for example, by machine ensembles, is advantageous and must be applied along with averaging. On the other hand, if generation is present,

$$\hat{a}_i = \frac{1}{N'_i} \sum_{n'_i} e'_i(\mathbf{x}^{(n'_i)}) \tag{23b}$$

where $e'_i(\cdot)$ excludes the generation rate, $\{\mathbf{x}^{(n'_i)}\}$ is the sample $\mathbf{x}^{(n_i)}$ from the training set and those generated from it, and N'_i is the total number of samples in the ith class. Formula (23a) is acceptable if the generated samples are sufficiently close to the original training set. Estimation errors of the normalization factors can directly affect the classifier performance. Therefore, careful attention must be paid to this process.

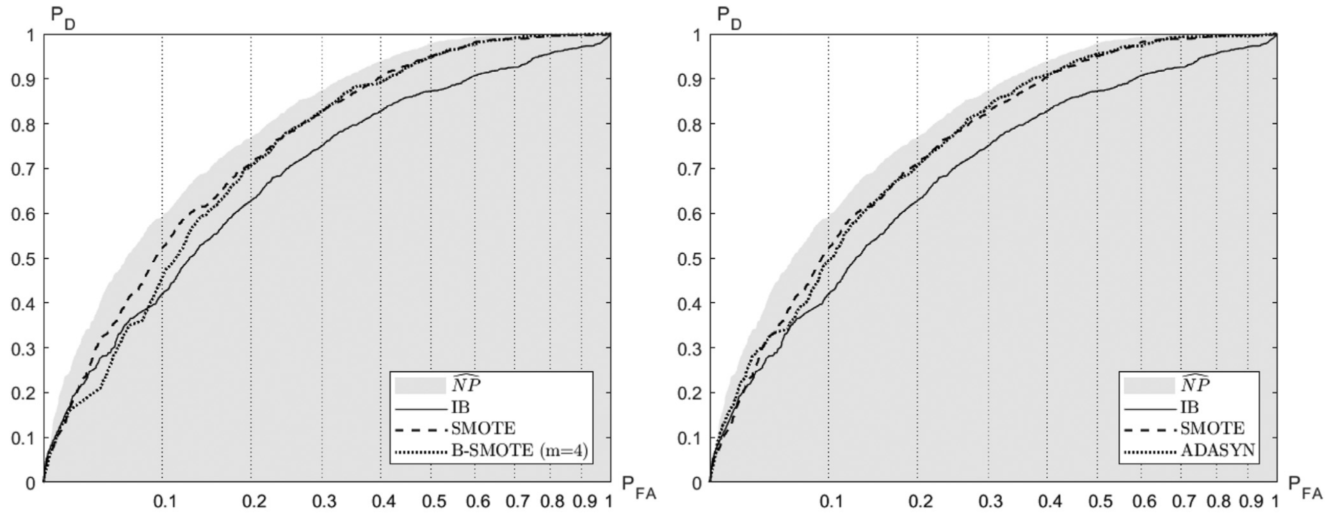
6.2. A two-step re-balancing procedure

The likelihood ratio $q'_L(\mathbf{x})$ is estimated more accurately where the "density" of the samples used for estimating the likelihood (including their weights) is higher. Without a good preliminary classifier capable of determining what samples are close to the (desired) working border, there is a real risk of modifying the sample density in an inappropriate manner. Such a classifier must be designed based on a Bregman surrogate cost and neutral re-balancing. This is the foundation of the two-step procedure we present here. The components of each step are enumerated below for the case of selecting a working point P_{FAW}. The case of given classification costs is similar:

Step 1

1.1. Design an auxiliary classifier, with output $o_{B1}(\mathbf{x})$, under a Bregman surrogate cost and with a neutral re-balance. It will preserve the likelihood ratio and, therefore, it will allow accurate estimation of the NP-ROC.

1.2. Select the desired working point P_{FAW} based on the estimated NP-ROC.



(a) (b)

Fig. 3. ROC of the preliminary classifier in the first stage of the two-step classifier for selected classes of “BNG:Page-blocks”. The border of the shaded area indicates the estimated NP-ROC. The machine ROCs are: without re-balancing (IB), SMOTE, B-SMOTE (in (a)), and ADASYN (in (b)).

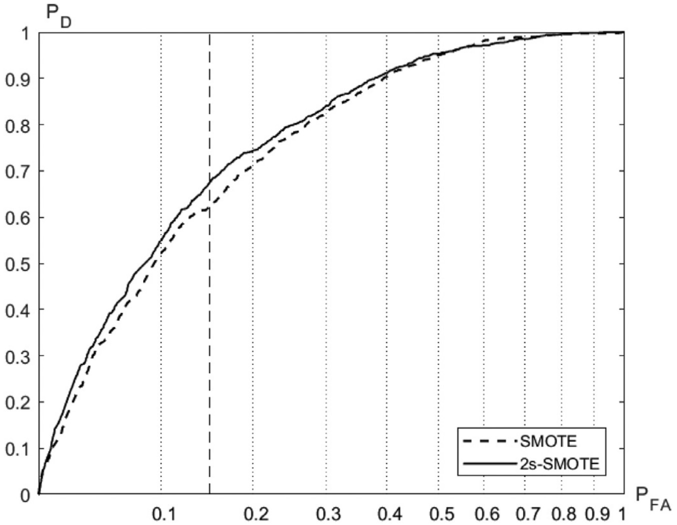


Fig. 4. 2s-SMOTE is the ROC curve for the second step SMOTE re-balance applied to the minority samples that are between $P_{FA} = 0$ and $P_{FA} = 0.3$ in the first step. The selected working point was at $P_{FAW} = 0.15$. The SMOTE curve from the first stage is also included here for an easier comparison.

1.3. Determine the weighting functions $e_i(\mathbf{x})$ based on the estimated NP-ROC and the selected working point.

Step 2

2.1. Apply $e_i(\mathbf{x})$ and train the second classifier with a Bregman surrogate cost.

2.2. Obtain the value of η_2^* for getting P_{FAW}.

Operation

Apply $o_{B2}(\mathbf{x}) \underset{C_0}{\overset{C_1}{\gtrless}} \eta_2^*$

The second re-balancing mechanism can weight the training samples as Borderline-SMOTE does. However, the two-step process described above pays more attention to those samples that are close to the desired working point. This is a non-trivial difference, as we will see in the examples below. There is freedom in the design process to try different forms of informed weight- ing, and to use one of them based on its performance in the prob- lem under analysis. Thus, it is possible to use parametric weight- ing forms such as those with profiles similar to those successfully applied to improve boosting (Ahachad, Álvarez-Pérez, & Figueiras- Vidal, 2017) and stacked denoising auto-encoding (SDAE) classification (Alvear-Sandoval & Figueiras-Vidal, 2018), extending previous simpler versions (Gómez-Verdejo, Arenas-García, & Figueiras-Vidal, 2008; Gómez-Verdejo, Ortega-Moral, Arenas-García, & Figueiras- Vidal, 2006), and determining parameter values by cross validation.

We emphasize that two-step re-balancing procedures provide designs that are appropriate for a given working point. This means that, if there is any reason to change the working point, a new design has to be carried out accordingly: Two-step designs are adapted to the working point and, contrarily to one-step (neu- tral) machines, they cannot provide a consistent solution for other working conditions.

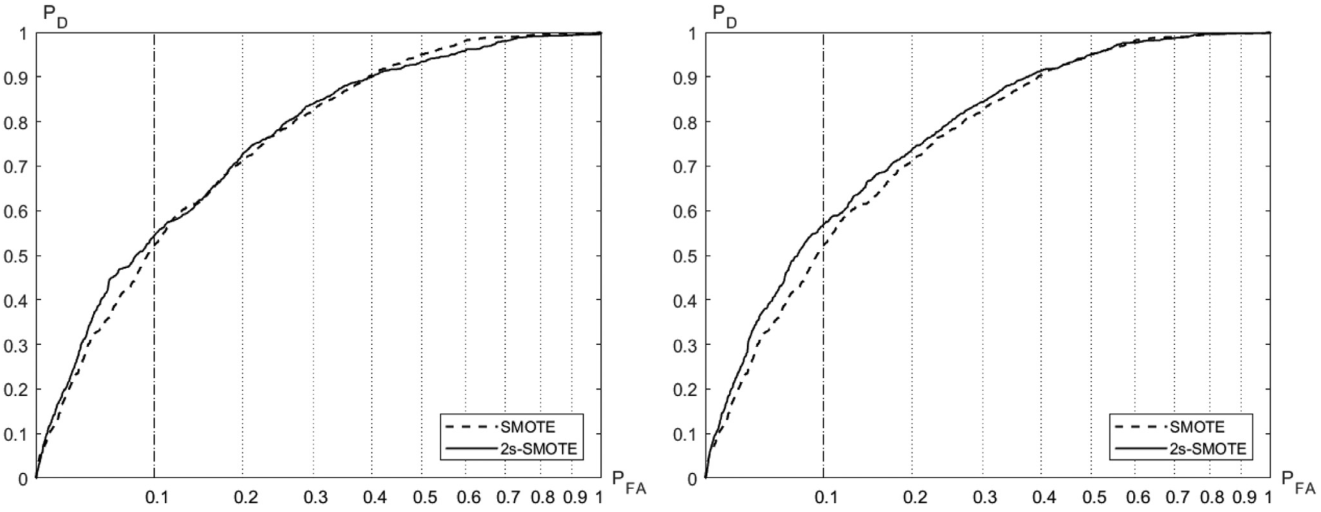
7.Examples on the two-step classification

7.1. Database

In these examples, we work with classes 1 and 5 of the multi-class database “BNG: Page-blocks” (Holmes, Pfahringer, van Rijn, & Vanschoren, 2014) for document page design element classification. Block 1 is the majority class C_0, having $N_0 = 265{,}174$ sam- ples, and Block 5 has $N_1 = 6238$ samples. Each sample in the data set was a ten-dimensional vector, corresponding to ten different features used by the classifier. Our training set was obtained by randomly selecting one third of the samples of each class, which resulted in the same imbalance ratio as the original dataset. The training data was further randomly divided into two subsets, one containing 75% of the data for training the classification machines, and the remaining 25% were used for testing the classifier. A pre- liminary estimate of the NP-ROC was obtained with a $K = 3$ SMOTE full re-balancing process working with all the samples. The classi- fication machine was the same for both steps of the algorithm.

7.2. Classification machine

A single-hidden-layer MLP network with $H = 27$ hyperbolic tangent hidden activations was employed. This machine offered good



(a)

(b)

Fig. 5. 2s-SMOTE ROC curves for the second SMOTE re-balancing on minority samples laying between a) $P_{FA}=0$ and $P_{FA}=0.2$ and b) $P_{FA}=0$ and $P_{FA}=0.25$; in the first step. The selected working point is $P_{FAW}=0.1$. The step-one SMOTE curve is included as a reference for comparison.

quality results for this classification problem. The results are given for ensembles of $M=5$ machines which provided sufficient stability.

7.3. Re-balancing process

The first step was carried out with a $K=3$ fully re-balancing SMOTE algorithm. Subsequently, we applied a fully re-balancing SMOTE with $K=3$ again, but just in an interval of P_{FA} values around the selected working point, P_{FAW}.

7.4. Results and discussion

Fig. 3 shows the results for the first step of the two-step classifier (SMOTE), as well as an estimate of the NP-ROC. In addition, this figure displays the ROCs associated with direct classification without re-balancing, first-step SMOTE as described above, B-SMOTE with $m=4$ and $K=3$, and ADASYN with $K'=3$ and $K=3$. These results are as expected: SMOTE gives the best NP-ROC estimates for low values of P_{FA}, while the differences between SMOTE and B-SMOTE or ADASYN are small when P_{FA} is high. We can also observe that the performance of B-SMOTE for low P_{FA} values is even worse than that of a direct imbalanced training. While this is not true for ADASYN, its disadvantage with respect to SMOTE is also clear. The advantage of ADASYN with respect to B-SMOTE can be attributed to the lower number of samples that ADASYN excludes from the generation process, 82, while B-SMOTE excludes 485. Once again, direct designs with imbalanced data (IB) offer poor results.

For the second step, we selected $P_{FAW}=0.15$, and applied SMOTE to the minority samples between the thresholds corresponding to $P_{FA}=0$ and $P_{FA}=0.3$ in the first step. The results appear in Fig. 4 as the 2s-SMOTE curve. The advantage at $P_{FAW}=0.15$ with respect to the SMOTE results is clear in this figure.

The selection of the re-sampling region is not a trivial task. Fig. 5 shows the 2s-SMOTE results obtained when $P_{FAW}=0.1$ and the SMOTE generation is applied for samples between $P_{FA}=0$ and $P_{FA}=0.2$ (a symmetric interval) and between $P_{FA}=0$ and $P_{FA}=0.25$. Performance difference can be easily noticed in these figures. For example, P_D is clearly higher for the asymmetric emphasis. Thus, it is important to explore how to select these second step characteristics.

8. An emphasized neutral re-balancing

There is an interesting and easy version of the general two-step re-balancing approach. In the second step, a neutral re-balancing procedure is applied, and an over-weighting of all (minority and majority) samples is carried out according to an appropriate profile of weights around the threshold η which corresponds to the desired P_{FAW}. The likelihood ratio invariance is maintained in this process, and more attention is paid to those samples that are more important to define the classification border. Consequently, the quality of the estimates - in particular, that of the likelihood ratio - will be higher in the corresponding region. As a result, better classification will be provided by the second classifier.

For this re-balancing scheme, we can derive the classification rule using techniques similar to those employed in the previous sections as

$$o_{B2}(\mathbf{x}) \underset{C_0}{\overset{C_1}{\gtrless}} \frac{\hat{Q}-\tilde{Q}_2\hat{Q}_A}{\hat{Q}+\tilde{Q}_2\hat{Q}_A} = \eta_2' \qquad (24)$$

where $\tilde{Q}_2$ corresponds to the neutrally re-balanced component, and $\hat{Q}_A$ comes from the joint weighting.

The best profile for the emphasis will be problem dependent. Even the intensity of the emphasis must be explored during the design process. We will do so in the examples that follow.

To clarify this approach, the skeleton of its algorithmic version (for a given P_{FAW}) is provided below.

Step 1

1.1. As in Section 6.2.
1.2. As in Section 6.2.
1.3. Determine a weighting form for the second step according to the previous NP-ROC estimate and the selected working point.

Step 2

2.1. Prepare the training samples for the second classifier by means of a (possibly new) neutral re-balancing, and apply the weights of 1.3 to all the training samples.
2.2. Design the second classifier using a Bregman surrogate cost.

The final classification of any sample $\mathbf{x}$ will be obtained directly from (24).

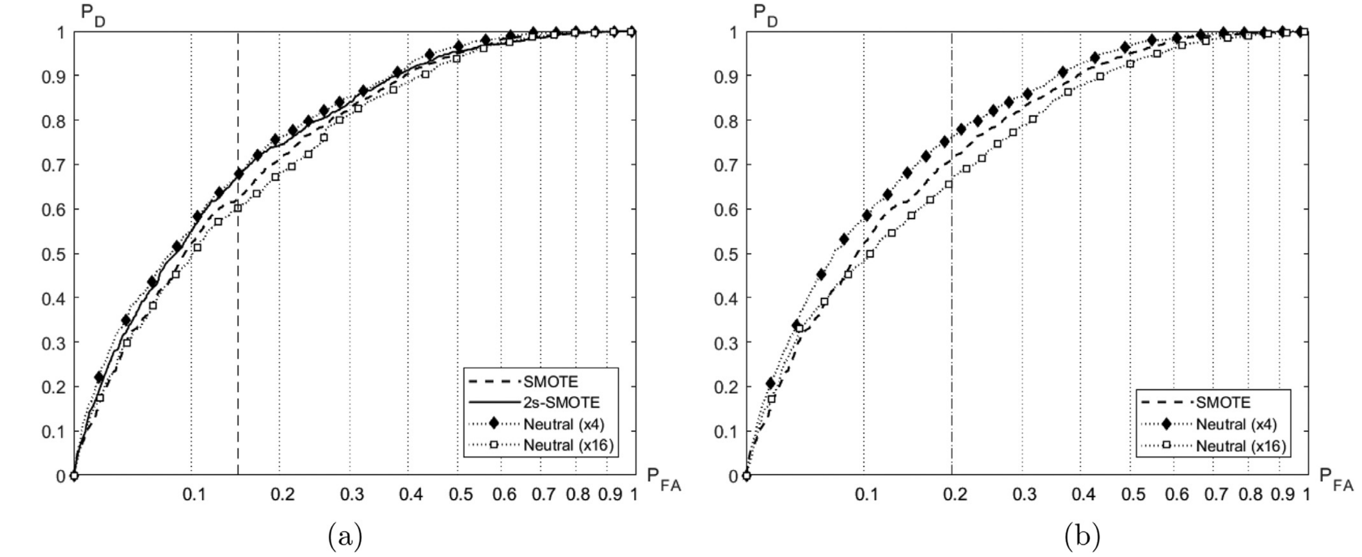


Fig. 6. ROC curves for the two-step emphasized neutral re-balancing: a) A $K = 3$ SMOTE fully re-balanced classifier is used in the first step and as the neutral component of the second step. The system also included an emphasis with values 4 and 16 for samples in $P_{FA} \in [0.1, 0.2]$, $P_{FAW} = 0.15$ being the working point. SMOTE and 2s-SMOTE of Section 7 are included for an easier comparison. b) Results corresponding to the same process for $P_{FAW} = 0.2$ and emphasizing samples in $P_{FA} \in [0.15, 0.25]$.

9. Examples in two-step neutral re-balancing

These examples employed the same database, problem, and machine architecture that were used in Section 7. The first-stage classifier was a fully re-balanced, $K = 3$ SMOTE design. The second- stage classifier included the same generative full re-balance plus a weighting of all the samples between two P_{FA} values around the selected working point, P_{FAW}.

Fig. 6a shows the results for $P_{F\ AW} = 0.\ 15$ and weight values 4 and 16 for samples between $P_{F\ A} = 0.\ 1$ and $P_{F\ A} = 0.\ 2$. Results for SMOTE and the 2s-SMOTE design of Section 7 are also included in the figure for comparison. The emphasis interval for the results from Section 7 was kept at [0, 0.25] because changing it de- graded the results. The two-step neutral method with weight 4 performed as good as the 2s-SMOTE at the working point, and is slightly better far from P_{FAW}. This is due to the neutral character of the new design. On the contrary, using a weight 16 offers results even worse than those of SMOTE: This was as expected because a very high emphasis produces an excessive deformation of the like- lihoods, and consequently, a worse estimate of their ratio.

Fig. 6b demonstrates that the method is effective for different working points.

Once again, we are estimating the Neyman–Pearson operation characteristic when applying a 2-step re-balancing procedure, but emphasizing the quality of the estimate around the desired working point.

10. Conclusions

In this contribution, we introduced a principled method to solve imbalanced binary classification problems. It is based on applying necessary and sufficient conditions to get an estimate of the likelihood ratio. It is important to remark that these necessary and sufficient conditions allow to get estimates of the Neyman–Pearson operating characteristic for the problem, which is optimal for classification. This is a very relevant advantage in the re-balancing processes. Our approach eliminates the imprecision risks that many empirical re-balancing procedures suffer from. The method is based on keeping the likelihood ratio invariant when constructing the re-balanced problem. This principled approach makes designing the classifier for the original imbalanced problem from the output of the machine solving a re-balanced problem straightforward and avoids the many risks associated with alternate approaches available in the literature. To accomplish this, the algorithm must be designed based on two required characteristics. First, the surrogate training cost must be a Bregman divergence. Second, re-balancing techniques employed must be neutral.

Additionally, we introduced a two-step method to design informed re-balancing techniques without the difficulties such methods typically present, allowing further improvements in performance. We also introduced a form of this two-step method in which a neutral re-balance and a common emphasis is applied to both minority and majority samples around a working point, also producing performance improvements.

We presented several illustrative examples that clearly demonstrated the importance of the re-balancing principles we introduced and the effectiveness of the two-step methods. Real-world applications require additional exploration of the parameter choices for the re-balancing algorithms.

There are several avenues to extend this research. One of the most important is to address the imbalanced multi-class problems (Bi & Zhang, 2018; Fernández, López, Galar, Del Jesus, & Herrera, 2013; Haixiang et al., 2017; Krawczyk, 2016; Sáez, Krawczyk, & Woźniak, 2016; Wang & Yao, 2012), that are frequent and relevant. These are more difficult problems because, in general, many imbalance ratios are present. We have established the principled formulations for the single machine classification and for the binary forms - One *vs* One, One *vs* Rest, and ECOC-based ensembles, - and the results of preliminary experiments are promising. These multi-class formulations can also be applied to other interesting problems such as ordinal and multi-label classification.

Another line of research to be explored involves problems with example-dependent costs. These problems are pervasive in many real-world finance, business, and health applications, and they appear usually under imbalanced conditions. Equally interesting is the question of how to adapt all these designs to online learning and big data applications.

Credit authorship contribution statement

Alexander Benítez-Buenache: Conceptualization, Investigation, Methodology, Software, Validation, Writing - original draft, Writing - review & editing. **Lorena Álvarez-Pérez:** Software, Writing - original draft, Writing - review & editing. **V. John Mathews:** Conceptualization, Supervision. **Aníbal R. Figueiras-Vidal:** Conceptualization, Formal analysis, Supervision, Writing - review & editing.

Acknowledgments

This work was partly supported by research grants CASI-CAM- CM (S2013/ ICE-2845, Madrid C./ FEDER, EUSF), Macro-ADOBE (TEC2015-67719-P, MINECO/FEDER, EU), and BBVA Foundation scientific research grant 2017 (2-BARBAS). V J. Mathews' work was supported by a Santander Chair of Excellence award at the Universidad Carlos III de Madrid.

The authors wish to thank the reviewers, who greatly helped to improve the manuscript.

Appendix A

The theoretical analysis of Bregman divergences is presented here.

Estimating a random variable t by means of a function $o(\mathbf{x})$ of a random vector $\mathbf{x}$ to minimize a cost $c(t, o)$ requires to minimize the average cost

$$\int_{\mathbf{x}}\int_{t} c(t,o)p(\mathbf{x},t)d\mathbf{x}dt = \int_{\mathbf{x}}\left[\int_{t} c(t,op(t|\mathbf{x})dt \quad p(x)d\mathbf{x} \qquad \text{(A.1)}$$

and, accepting that $c(t, o)$ is non-negative, it is enough to minimize the inner integral, because $p(\mathbf{x})$ is non-negative:

$$o_c(\mathbf{x}) = \operatorname*{argmin}_{o} \int c(t,o)p(t|\mathbf{x})dt \qquad \text{(A.2)}$$

Obviously, we can always write

$$\frac{\partial\, c(t,o)}{\partial\, o} = -g(t,o)(t-o) \qquad \text{(A.3)}$$

by appropriately defining $g(\ \ t, o\ )$. Assuming that the integral in (A.2) is (absolutely) convergent, the solution is given by

$$\int g(t,o)(t-o)p(t|\mathbf{x})dt = 0 \qquad \text{(A.4)}$$

If $c(t,\ o)$ is a Bregman divergence, $g(t,o) = g(o),$ and (A.3) becomes

$$g(o_c)\int (t-o_c)p(t|\mathbf{x})dt = 0 \qquad \text{(A.5)}$$

and, since $g(o_c) \neq 0,$ we get

$$\int o_c p(t|\mathbf{x})dt = o_c(\mathbf{x}) = \int t\,p(t|\mathbf{x})dt = E\{t|\mathbf{x}\} \qquad \text{(A.6)}$$

However, if $g(t,\ o)\ \not= g(o),$ (A.6) does not result.

When working with sampled data, the integrals are replaced by summations and we obtain estimates of the *a posteriori* average of t, whose quality depends on the number and information amount of the samples and the capability of the functional form which is selected for $o(\mathbf{x})$ to approximate $E\{t\ |\mathbf{x}\}$. Additionally, the particular function $g(\ \ o)$ which is selected has, in practice, weighting effects that affect the characteristics of the obtained estimate of $E\{t\ |\mathbf{x}\}$.

References

Abe, N. (2003). Sampling approaches to learning from imbalanced datasets: Active learning, cost sensitive learning and beyond. *Workshop learning from imbalanced data sets II, in proc. 20th intl. conf. machine learning*. Washington, DC: IEEE Press. Ahachad, A., Álvarez-Pérez, L., & Figueiras-Vidal, A. R. (2017). Boosting ensembles with controlled emphasis intensity. *Pattern Recognition Letters, 88*, 1–5.

Alvear-Sandoval, R., & Figueiras-Vidal, A. R. (2018). On building ensembles of stacked denoising auto-encoding classifiers and their further improvement. *Information Fusion, 39*, 41–52.

Barua, S., Islam, M. M., Yao, X., & Murase, K. (2014). MWMOTE-majority weighted minority oversampling technique for imbalanced data set learning. *IEEE Transactions of Knowledge and Data Engineering, 26*, 405–425.

Batista, G. E., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter, 6*, 20–29.

Batuwita, R., & Palade, V. (2009). Micropred: Effective classification of pre-miRNAs for human miRNA gene prediction. *Bioinformatics, 25*, 989–995.

Batuwita, R., & Palade, V. (2010). FSVM-CIL: Fuzzy support vector machines for class imbalance learning. *IEEE Transactions of Fuzzy Systems, 18*, 558–571.

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning, 2*, 1–127.

Bi, J., & Zhang, C. (2018). An empirical comparison on state-of-the-art multi-class imbalance learning algorithms and a new diversified ensemble learning scheme. *Knowledge-Based Systems, 158*, 81–93.

Bordes, A., Ertekin, S., Weston, J., & Bottou, L. (2005). Fast kernel classifiers with online and active learning. *Journal of Machine Learning Research, 6*, 1579–1619.

Branco, P., Torgo, L., & Ribeiro, R. P. (2016). A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys, 49*. 31:1-31:50

Bregman, L. M. (1967). The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics, 7*, 200–217.

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). *Classification and regression trees*. Wadsworth & Brooks.

Bunkhumpornpat, C., Sinapiromsaran, K., & Lursinsap, C. (2009). Safe-level-SMOTE: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem. In *Proc. pacific-asia conf. advances in knowledge discovery and data mining, Bangkok (thailand), 475–482*. Springer.

Bunkhumpornpat, C., Sinapiromsaran, K., & Lursinsap, C. (2012). DBSMOTE: Density-based synthetic minority over-sampling technique. *Applied Intelligence, 36*, 664–684.

Castro, C. L., & Braga, A. P. (2013). Novel cost-sensitive approach to improve the multilayer perceptron performance on imbalanced data. *IEEE Transactions on Neural Networks and Learning Systems, 24*, 888–899.

Chan, P. K., & Stolfo, S. J. (1998). Toward scalable learning with non-uniform class and cost distributions: A case study in credit card fraud detection. In *Proc. 4th intl. conf. knowledge discovery and data mining, New York, NY, 164–168*. AAAI Press.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321–357.

Chen, X., Fang, T., Huo, H., & Li, D. (2011). Graph-based feature selection for object-oriented classification in VHR airborne imagery. *IEEE Transactions on Geoscience and Remote Sensing, 49*, 353–365.

Cid-Sueiro, J., Arribas, J. I., Urbán-Muñoz, S., & Figueiras-Vidal, A. R. (1999). Cost functions to estimate a posteriori probabilities in multiclass problems. *IEEE Transactions on Neural Networks, 10*, 645–656.

Cid-Sueiro, J., & Figueiras-Vidal, A. R. (2001). On the structure of strict sense bayesian cost functions and its applications. *IEEE Transactions on Neural Net- works, 12*, 445–455.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems, 2*, 303–314.

Dal Pozzolo, A., Caelen, O., & Bontempi, G. (2015). When is undersampling effective in unbalanced classification tasks? *Machine learning and knowledge discovery in databases, 200–215*. Springer.

Deng, L., & Yu, D. (2014). Deep learning: Methods and applications. *Foundations and Trends in Signal Processing, 7*, 197–387.

Domingos, P. (1999). Metacost: A general method for making classifiers cost-sensitive. In *Proc. 5th ACM intl. conf. knowledge discovery and data mining, San Diego, CA, 155–164*. ACM.

Elkan, C. (2001). The foundations of cost-sensitive learning. In *Proc. 7th intl. conf. machine learning, Stanford, CA, 973–978*. Morgan Kaufmann.

Ertekin, S., Huang, J., Bottou, L., & Giles, C. L. (2007). Learning on the border: Active learning in imbalanced data classification. In *Proc. 16th ACM conf. information and knowledge management, 127–136, Lisbon (Portugal)*. ACM Press.

Estabrooks, A., Jo, T., & Japkowicz, N. (2004). A multiple resampling method for learning from imbalanced data sets. *Computational Intelligence, 20*, 18–36.

Fan, W., Stolfo, S. J., Zhang, J., & Chan, P. K. (1999). Adacost: Misclassification cost-sensitive boosting. In *Proc. 16th intl. conf. machine learning, 97–105, Bled (Slovenia)*.

Fernández, A., López, V., Galar, M., Del Jesus, M. J., & Herrera, F. (2013). Analysing the classification of imbalanced data-sets with multiple classes: Binarization techniques and ad-hoc approaches. *Knowledge-Based Systems, 42*, 97–110.

Freitas, A. (2011). Building cost-sensitive decision trees for medical applications. *AI Communications, 24*, 285–287.

Fung, G. M., & Mangasarian, O. L. (2005). Multicategory proximal support vector machine classifiers. *Machine Learning, 59*, 77–97.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2012). A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Pt. C, 42*, 463–484.

Gómez-Verdejo, V., Arenas-García, J., & Figueiras-Vidal, A. R. (2008). A dynamically adjusted mixed emphasis method for building boosting ensembles. *IEEE Trans- actions on Neural Networks, 19*, 3–17.

Gómez-Verdejo, V., Ortega-Moral, M., Arenas-García, J., & Figueiras-Vidal, A. R. (2006). Boosting by weighting critical and erroneous samples. *Neurocomputing, 69*, 679–685.

González, P., Álvarez, E., Barranquero, J., Díez, J., González-Quirós, R., Nogueira, E., et al. (2013). Multiclass support vector machines with example-dependent costs applied to plankton biomass estimation. *IEEE Transactions on Neural Networks and Learning Systems, 24*, 1901–1905.

Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., & Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications, 73*, 220–239.

Han, H., Wang, W. Y., & Mao, B. H. (2005). Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In *Proc. intl. conf. intelligent computing, Hefei (China), 878–887*. Springer.

Harries, M., Gama, J., & Bifet, A. (2009). Electricity data-set. [dataset] OpenML Data Set Repository. https://www.openml.org/d/151.

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *Proc. intl. joint conf. neural networks, Hong Kong (China), 1322–1328*. IEEE.

He, H., & García, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering, 21*, 1263–1284.

He, H., & Ma, Y. (Eds.) (2013). Imbalanced learning: Foundations, algorithms, and applications, hoboken, NJ, IEEE-Wiley.

Hido, S., Kashima, H., & Takahashi, Y. (2009). Roughly balanced bagging for imbalanced data. *Statistical Analysis and Data Mining, 2*, 412–426.

Holmes, G., Pfahringer, B., van Rijn, J., & Vanschoren, J. (2014). BNG (page-blocks) data-set. [dataset] OpenML Data Set Repository. https://www.openml.org/d/259.

Hong, X., Chen, S., & Harris, C. J. (2007). A kernel-based two-class classifier for imbalanced data sets. *IEEE Transactions on Neural Networks, 18*, 28–41.

Hornik, K., Stinchcombe, M., & White, H., Jr. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks, 2*, 359–366.

Hu, S., Liang, Y., Ma, L., & He, Y. (2009). MSMOTE: Improving classification performance when training data is imbalanced. In *Proc. 2nd intl. workshop comp. sci. eng., Quingdao (China): vol. 2* (pp. 13–17). IEEE.

Imam, T., Ting, K., & Kamruzzaman, J. (2006). z-SVM: An SVM for improved classification of imbalanced data. In *Proc. 19th australian joint conf. artificial intelligence, 264–273, Hobart (Australia)*.

Japkowicz, N. (2000). Learning from imbalanced data sets: A comparison of various strategies. In *Proc. AAAI workshop in learning from imbalanced data sets, Austin, TX, 10–15*. AAAI Press.

Japkowicz, N., & Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis, 6*, 429–449.

Jo, T., & Japkowicz, N. (2004). Class imbalances versus small disjuncts. *ACM SIGKDD Explorations Newsletter, 6*, 40–49.

Khoshgoftaar, T. M., Seiffert, C., Van Hulse, J., Napolitano, A., & Folleco, A. (2007). Learning with limited minority class data. In *Proc. 6th intl. conf. machine learning and applications, 348–353, Cincinnati, OH*. IEEE.

Kowalczyk, A., & Raskutti, B. (2002). One class SVM for yeast regulation prediction. *SIGKDD Explorations Newsletter, 4*, 99–100.

Krawczyk, B. (2016). Learning from imbalanced data: Open challenges and future directions. *Progress in Artificial Intelligence, 5*, 221–232.

Kubat, M., & Matwin, S. (1997). Addressing the curse of imbalanced training sets: One-sided selection. In *Proc. 14th intl. conf. machine learning, Nashville, TN, 179–186*. Morgan Kaufmann.

Kukar, M., & Kononenko, I. (1998). Cost-sensitive learning with neural networks. In *Proc. european conf. 13th artificial intelligence, Brighton (UK), 4 45–4 49*. Wiley.

Kwak, N. (2008). Feature extraction for classification problems and its application to face recognition. *Pattern Recognition, 41*, 1701–1717.

Laurikkala, J. (2001). Improving identification of difficult small classes by balancing class distribution. *Artificial intelligence in medicine, 63–66*. Springer.

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

Lee, S. S. (20 0 0). Noisy replication in skewed binary classification. *Computational Statistics and Data Analysis, 34*, 165–191.

Liao, T. W. (2008). Classification of weld flaws with imbalanced class data. *Expert Systems with Applications, 35*, 1041–1052.

Ling, C. X., & Li, C. (1998). Data mining for direct marketing: Problems and solutions. In *Proc. 4th int. conf. knowledge discovery and data mining, New York, NY, 73–79*. AAAI Press.

Liu, B., Hsu, W., & Ma, Y. (1999). Mining association rules with multiple minimum supports. In *Proc. 5th ACM intl. conf. knowledge discovery and data mining, San Diego, CA, 337–341*. ACM.

López, V., Fernández, A., García, S., Palade, V., & Herrera, F. (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences, 250*, 113–141.

Manevitz, L. M., & Yousef, M. (2001). One-class SVMs for document classification. *Journal of Machine Learning Research, 2*, 139–154.

Masnadi-Shirazi, H., & Vasconcelos, N. (2010). Risk minimization, probability elicitation, and cost-sensitive SVMs. In *Proc. 27th intl. conf. machine learning, 759–766, Haifa (Israel)*.

Masnadi-Shirazi, H., & Vasconcelos, N. (2011). Cost-sensitive boosting. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 33*, 294–309.

Mazurowski, M. A., Habas, P. A., Zurada, J. M., Lo, J. Y., Baker, J. A., & Tourassi, G. D. (2008). Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. *Neural Networks, 21*, 427–436.

Mehrotra, H., Singh, R., Vatsa, M., & Majhi, B. (2016). Incremental granular relevance vector machine: A case study in multimodal biometrics. *Pattern Recognition, 56*, 63–76.

Mena, L., & González, J. A. (2009). Symbolic one-class learning from imbalanced datasets: Application in medical diagnosis. *International Journal on Artificial Intelligence Tools, 18*, 273–309.

Nahar, J., Imam, T., Tickle, K. S., & Chen, Y. P. P. (2013). Computational intelligence for heart disease diagnosis: A medical knowledge driven approach. *Expert Systems with Applications, 40*, 96–104.

Park, B. J., Oh, S. K., & Pedrycz, W. (2013). The design of polynomial function-based neural network predictors for detection of software defects. *Information Sciences, 229*, 40–57.

Parzen, E. (1962). On estimation of a probability density function and mode. *Annals of Mathematical Statistics, 33*, 1065–1076.

Phua, C., Alahakoon, D., & Lee, V. (2004). Minority report in fraud detection: Classification of skewed data. *ACM SIGKDD Explorations Newsletter, 6*, 50–59.

Provost, F., & Fawcett, T. (2001). Robust classification for imprecise environments. *Machine Learning, 42*, 203–231.

Radivojac, P., Chawla, N. V., Dunker, A. K., & Obradovic, Z. (2004). Classification and knowledge discovery in protein databases. *Journal of Biomedical Informatics, 37*, 224–239.

Ramentol, E., Caballero, Y., Bello, R., & Herrera, F. (2012). 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*, 245–265.

Rao, R. B., Krishnan, S., & Niculescu, R. S. (2006). Data mining for improved cardiac care. *ACM SIGKDD Explorations Newsletter, 8*, 3–10.

Sáez, J. A., Krawczyk, B., & Woźniak, M. (2016). Analyzing the oversampling of different classes and types of examples in multi-class imbalanced datasets. *Pattern Recognition, 57*, 164–178.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks, 61*, 85–117.

Seiffert, C., Khoshgoftaar, T. M., Van Hulse, J., & Folleco, A. (2014). An empirical study of the classification performance of learners on imbalanced and noisy software quality data. *Information Sciences, 259*, 571–595.

Settles, B. (2010). Active Learning Literature Survey. *Tech. Report 1648*. Computer Sci. Dept., Univ. Wisconsin-Madison.

Stefanowski, J. (2016). Dealing with data difficulty factors while learning from imbalanced data. *Challenges in computational statistics and data mining, 333–363*. Springer.

Sun, Y., Kamel, M. S., Wong, A. K. C., & Wang, Y. (2007). Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognition, 40*, 3358–3378.

Sun, Y., Wong, A. K. C., & Kamel, M. S. (2009). Classification of imbalanced data: A review. *International Journal on Pattern Recognition and Artificial Intelligence, 23*, 687–719.

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

Tavallaee, M., Stakhanova, N., & Ghorbani, A. A. (2010). Toward credible evaluation of anomaly-based intrusion-detection methods. *IEEE Transactions on Systems, Man, and Cybernetics, Pt. C, 40*, 516–524.

Ting, K. M. (2000). A comparative study of cost-sensitive boosting algorithms. In *Proc. 17th intl. conf. machine learning, 983–990, Stanford, CA*.

Tong, S., & Koller, D. (2001). Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research, 2*, 45–66.

De la Torre, M., Granger, E., Sabourin, R., & Gorodnichy, D. O. (2015). Adaptive skew-sensitive ensembles for face recognition in video surveillance. *Pattern Recognition, 48*, 3385–3406.

Triguero, I., del Río, S., López, V., Bacardit, J., Benítez, J. M., & Herrera, F. (2015). ROSEFW-RF: The winner algorithm for the ECBDL'14 big data competition: An extremely imbalanced big data bioinformatics problem. *Knowledge-Based Systems, 87*, 69–79.

Tsai, C. H., Chang, L. C., & Chiang, H. C. (2009). Forecasting of ozone episode days by cost-sensitive neural network methods. *Science of the Total Environment, 407*, 2124–2135.

Van Trees, H. L. (1968). *Detection, estimation, and modulation theory, part i*. Wiley.

Veropoulos, K., Campbell, C., & Cristianini, N. (1999). Controlling the sensitivity of support vector machines. In *Proc. 20th intl. joint conf. artificial intelligence, 55–60, Stockholm (Sweden)*.

Viola, P., & Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision, 57*, 137–154.

Wallace, B. C., Small, K., Brodley, C. E., & Trikalinos, T. A. (2011). Class imbalance, redux. In *Proc. 11th intl. conf. data mining, 754–763*. IEEE Press.

Wang, S., & Yao, X. (2012). Multiclass imbalance problems: Analysis and potential solutions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernet- ics), 42*, 1119–1130.

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

Yang, C. Y., Yang, J. S., & Wang, J. J. (2009). Margin calibration in SVM class-imbalanced learning. *Neurocomputing, 73*, 397–411.

Yu, H., Ni, J., & Zhao, J. (2013). ACOSampling: An ant colony optimization-based undersampling method for classifying imbalanced DNA microarray data. *Neurocomputing, 101*, 309–318.

Zadrozny, B., Langford, J., & Abe, N. (2003). Cost-sensitive learning by cost-proportionate example weighting. In *Proc. intl. conf. data mining, Melbourne, FL, 435–442, IEEE comp. soc.*

Zhou, L. (2013). Performance of corporate bankruptcy prediction models on imbalanced dataset: The effect of sampling methods. *Knowledge-Based Systems, 41*, 16–25.