Knowledge-Based Systems 25 (2012) 13–21

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

On the effectiveness of preprocessing methods when dealing with different levels of class imbalance

V. García, J.S. Sánchez *, R.A. Mollineda

Institute of New Imaging Technologies, Dept. Llenguatges i Sistemes Informàtics, Universitat Jaume I, Av. Sos Baynat s/n, 12071 Castelló de la Plana, Spain

ARTICLE INFO

Article history:
Available online 26 June 2011

Keywords:
Imbalance
Resampling
Classification
Performance measures
Multi-dimensional scaling

ABSTRACT

The present paper investigates the influence of both the imbalance ratio and the classifier on the performance of several resampling strategies to deal with imbalanced data sets. The study focuses on evaluating how learning is affected when different resampling algorithms transform the originally imbalanced data into artificially balanced class distributions. Experiments over 17 real data sets using eight different classifiers, four resampling algorithms and four performance evaluation measures show that over-sampling the minority class consistently outperforms under-sampling the majority class when data sets are strongly imbalanced, whereas there are not significant differences for databases with a low imbalance. Results also indicate that the classifier has a very poor influence on the effectiveness of the resampling strategies.

1. Introduction

Class imbalance constitutes one of the problems that has recently received most attention in research areas such as Machine Learning, Pattern Recognition, Data Mining, and Knowledge Discovery. A two-class data set is said to be imbalanced if one of the classes (the minority one) is represented by a very small number of instances in comparison to the other (majority) class [1]. Besides, the minority class is usually the most important one from the point of view of the learning task. It has been observed that class imbalance may cause a significant deterioration in the performance attainable by standard learners because these are often biased towards the majority class [2,3]. These classifiers attempt to reduce global measures such as the error rate, not taking the data distribution into consideration. This issue is especially important in real-world applications where it is often costly to misclassify examples of the minority class, such as diagnosis of infrequent diseases [4], fraud detection in mobile telephone communications [5,6] or credit cards [7], detection of oil spills in satellite radar images [8], text categorization [9,10], credit assessment [11], prediction of customer insolvency [12], and translation initiation site recognition in DNA sequences [13]. Because of examples of the minority and majority classes usually represent the presence and absence of rare cases respectively, they are also known as positive and negative examples.

Main research on this topic can be categorized into three groups. One has primarily focused on the implementation of solutions for handling the imbalance, both at the data and algorithmic levels [14–16]. Another group has addressed the problem of measuring the classifier performance in imbalanced domains [17,18]. The third has been to analyse what data complexity characteristics aggravate the problem and even, to study whether there exist other factors that lead to a loss of classifier performance or it is the imbalance problem per se that causes the performance decrease [19,20].

Among the most investigated issues, one can find both algorithmic and data level solutions. Examples of the former are approaches to internally biasing the discriminating process [14] and multi-experts systems [21], whereas the data level solutions consist of artificially resampling the original data set until the problem classes are approximately equally represented. Conclusions about what is the best data level solution for the class imbalance problem are divergent. In this sense, Hulse and Khoshgoftaar [22] suggest that the utility of each particular resampling technique depends on various factors, including the ratio between positive and negative examples, the characteristics of data, and the nature of the classifier. Other papers [2,23–25] have also studied this dependence during the last decade. Nevertheless, their conclusions should be carefully interpreted because most of them are based on narrow learning frameworks.

In many ways, this paper significantly extends previous works by increasing the scope and detail at which it is studied the influence of the imbalance ratio and the classifier on the effectiveness of the most popular resampling strategies (under and over-sampling). To this end, we will carry out a collection of experiments over 17 real databases with two different levels of imbalance, employing

* Corresponding author.
E-mail addresses: jimenezv@uji.es (V. García), sanchez@uji.es (J.S. Sánchez), mollined@uji.es (R.A. Mollineda).

0950-7051/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.knosys.2011.06.013

eight classifiers, four resampling techniques and four performance metrics.

The rest of the paper is organized as follows. Section 2 reviews several resampling techniques for problems with imbalanced data sets. Section 3 surveys a number of common performance evaluation measures, which can be especially useful for class imbalance. Next, in Section 4 the experimental set-up is described. Section 5 reports the results and discusses the most important findings. Finally, Section 6 remarks our conclusions and outlines possible directions for future research.

2. Data-driven methods for balancing the class distributions

Resampling techniques aim at correcting problems with the distribution of a data set [26]. Weiss and Provost [27] noted that in many real applications the original distribution of samples is not always the best distribution to use for a given classifier, and different resampling approaches try to modify the "natural" distribution to another that is closer to the optimal one. This can be accomplished either by over-sampling the minority class, by under-sampling the majority class, or by combining simple over and under-sampling techniques in a systematic manner [25,28]. All these strategies can be applied to any learning system, since they act as a preprocessing phase, allowing the learning system to receive the training instances as if they belonged to a well-balanced data set. Thus any bias of the system towards the majority class due to the different proportion of examples per class would be expected to be eliminated.

While these methods can result in greatly improved results over the original data set, they have also shown several important drawbacks. Under-sampling techniques may throw out potentially valuable data, whereas over-sampling artificially increases the size of the data set and consequently, worsens the computational burden of the learning algorithm. On the other hand, both under and over-sampling modify the prior probability of classes, and both lead to a decrease in the accuracy of the negative class.

Effectiveness of these resampling approaches has been analysed in previous studies with respect to different sources of data complexity and classification models. However, most of them have focused on some particular learning factors (classifiers, data sets, performance metrics, resampling strategies), but disregarding the effect of others.

- Japkowicz and Stephen [2] discussed the performance of basic resampling methods when using a C5.0 decision tree induction system over a reduced number of artificial and real-world data sets. The error rate on each class was recorded to carry out this study.
- Barandela et al. [24] presented an empirical comparison of several under and over-sampling techniques based on intelligent heuristics. The experiments were constrained to five real data sets using the nearest neighbour rule for classification and the geometric mean as the performance evaluation metric.
- Estabrooks et al. [25] studied the behaviour of random strategies at different resampling rates with C4.5 classifiers. They evaluated the performance on seven artificial and five real data sets by means of the overall error rate and the error on each class.
- Batista et al. [23] conducted a broad experimental analysis with 13 databases and 10 resampling methods, but conclusions were limited to the C4.5 decision tree and the use of the area under the ROC curve for assessing the results.

2.1. Over-sampling

The simplest method to increase the size of the minority class corresponds to random over-sampling, that is, a non-heuristic method that balances the class distribution through the random replication of positive examples. This contributes to balance the class distribution without adding new information to the data set. Nevertheless, since this method replicates existing positive examples, overfitting is more likely to occur.

Instead of simply duplicating original examples, Chawla et al. [15] proposed an over-sampling technique that generates new synthetic minority instances by interpolating between preexisting positive examples that lie close together. This method, called SMOTE (Synthetic Minority Over-sampling TEchnique), allows the classifier to build larger decision regions that contain nearby instances from the minority class.

From the original SMOTE algorithm, several modifications have further been proposed in the literature. For example, SMOTEBoost is an approach introduced by Chawla et al. [29] that combines SMOTE with the standard boosting procedure. García et al. [30] developed three variants based upon the concept of surrounding neighbourhood with the aim of taking both proximity and spatial distribution of the instances into consideration. Han et al. [31] presented the Borderline-SMOTE algorithm, which only creates new minority examples based on existing instances that are near the decision border. On the other hand, MSMOTE [32] not only considers the distribution of minority instances but also rejects latent noise based on the k-nearest neighbour classifier. Hongyu and Herna [33] introduced the DataBoost-IM method, which combines boosting and data generation.

2.2. Under-sampling

Random under-sampling [2,34] aims at balancing the data set through the random removal of negative examples. Despite important information can be lost when examples are discarded at random, it has empirically been shown to be one of the most effective resampling methods.

Unlike the random approach, many other proposals are based on a more intelligent selection of the negative examples to be eliminated. For example, Kubat and Matwin [35] proposed the one-sided selection (OSS) technique, which selectively removes only those negative instances that either are redundant or that border the minority class examples (the authors assume that these bordering cases are noise). The border examples were detected using the concept of Tomek links [36], while the redundant ones were eliminated by means of Hart's condensing [37].

In contrast to the one-sided selection technique, the so-called neighbourhood cleaning rule [38] emphasizes more data cleaning than data reduction. To this end, Wilson's editing [39] is used to identify and remove noisy negative instances. Similarly, Barandela et al. [14] introduced a method that eliminates not only noisy instances of the majority class by means of Wilson's editing (WE), but also redundant examples through the modified selective subset (MSS) condensing algorithm [40].

On the other hand, Yen et al. [41] presented a cluster-based under-sampling algorithm. It first clusters all the original examples into some clusters, and then selects an appropriate number of majority class samples from each cluster by considering the ratio of the number of majority class samples to the number of minority class samples in the cluster. García and Herrera [42] proposed the use of evolutionary computation algorithms to under-sample the majority class. Chen et al. [43] introduced a method based on pruning support vectors of the majority class.

3. Performance metrics for imbalanced class distributions

Evaluation of classification performance plays a critical role in the design of a learning system and therefore, the use of an appropriate measure becomes as important as the selection of a good algorithm to successfully tackle a given problem. Traditionally,

standard performance metrics have been classification accuracy and/or error rates. For a two-class problem, these can be easily derived from a 2×2 confusion matrix as that given in Table 1.

The classification accuracy (Acc) and its counterpart, the error rate ($Err = 1 - Acc$), evaluate the effectiveness of the learner by its percentage of correct (or incorrect) predictions:

$$Acc = \frac{TP + TN}{TP + FN + TN + FP} \tag{1}$$

Empirical and theoretical evidences show that these measures are strongly biased with respect to data imbalance and proportions of correct and incorrect classifications. In a binary decision problem, a learner predicts instances as either positive or negative; if very few examples belong to the positive class, a naive learning system could obtain a very high accuracy by just classifying all instances as negative. However, this is useless in most real domains because the class of interest is generally the positive one. Therefore, evaluators such as accuracy or error rate appear to be inappropriate for class imbalanced data, thus motivating the search for other measures based on some straightforward indexes, which have also been formulated from the 2×2 confusion matrix.

To deal with class imbalance, sensitivity (or recall) and specificity have usually been adopted to monitor the classification performance on each class separately. Note that sensitivity (also called true positive rate, TPrate) is the percentage of positive examples that are correctly classified, while specificity (also referred to as true negative rate, TNrate) is defined as the proportion of negative examples that are correctly classified:

$$TPrate = \frac{TP}{TP + FN} \tag{2}$$

$$TNrate = \frac{TN}{TN + FP} \tag{3}$$

Similarly, the false positive rate (FPrate) represents the percentage of negative examples that are misclassified, whereas the false negative rate (FNrate) measures the proportion of positive examples that are misclassified:

$$FPrate = \frac{FP}{TN + FP} \tag{4}$$

$$FNrate = \frac{FN}{TP + FN} \tag{5}$$

On the other hand, in several problems we are especially interested in obtaining high performance on only one class. For example, in the diagnosis of a rare disease, one of the most important things is to know how reliable is a positive diagnosis. For such problems, the precision (or purity) metric is often adopted, which can be defined as the percentage of examples that are correctly labeled as positive:

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

Apart from these simple metrics, it is possible to encounter several more complex evaluation measures that have been used in different practical domains. One of the most popular techniques for the evaluation of classifiers in imbalanced problems is the Receiver Operating Characteristic (ROC) curve [44], which is a tool for visualizing, organizing and selecting classifiers based on their trade-offs between benefits (true positives) and costs (false positives). A quantitative representation of a ROC curve is the area under it, which is known as AUC [45,17]. When only one run is available from a classifier, the AUC can be computed as the arithmetic mean (macro-average) of TPrate and TNrate [46]:

$$AUC = \frac{TPrate + TNrate}{2} \tag{7}$$

Table 1
Confusion matrix for a two-class decision problem.

	Predicted positive	Predicted negative
Actual positive	True positive (TP)	False negative (FN)
Actual negative	False positive (FP)	True negative (TN)

The *F*-measure [47] is used to integrate precision and TPrate into a single metric, representing a weighted harmonic mean between these two metrics:

$$F = \frac{(1 + \beta^2) \cdot (Precision \cdot TPrate)}{\beta^2 \cdot Precision + TPrate} \tag{8}$$

Here, the non-negative constant β is a parameter to control the influence of TPrate and precision separately. It can be demonstrated that when $\beta = 0$, then the *F*-measure reduces to precision and conversely, it approaches TPrate when $\beta \to \infty$. Generally, β is set to 1, what turns the *F*-measure into the harmonic mean of precision and TPrate, thus obtaining the following expression (usually known as the F_1-measure):

$$F_1 = \frac{2 \cdot Precision \cdot TPrate}{Precision + TPrate} \tag{9}$$

To evaluate the performance in class imbalance problems, Kubat and Matwin [35] use the geometric mean of accuracies measured separately on each class:

$$Gmean = \sqrt{TPrate \cdot TNrate} \tag{10}$$

This metric is associated to a point on the ROC curve, and the idea is to maximize the accuracies of both classes while keeping them balanced. It can be seen as a kind of good trade-off between both rates because a high value occurs when they both are also high, whereas a low value is related to at least one low rate.

Although AUC and Gmean minimize the negative influence of skewed class distributions, they cannot distinguish between the contribution of each class to the overall performance, nor which is the dominant class. This means that different combinations of TPrate and TNrate may produce the same result for those two metrics.

As an attempt to remedy this deficiency, Ranawana and Palade [18] introduced the optimized precision:

$$OP = Acc - \frac{|TNrate - TPrate|}{TNrate + TPrate} \tag{11}$$

This represents the difference between the global accuracy and a second term that computes how balanced both class accuracies are. High OP values require high overall accuracy and well-balanced class accuracies. However, OP can be strongly affected by the biased influence of the overall accuracy.

Recently, García et al. [48] proposed a new measure called Generalized Index of Balanced Accuracy, which can be defined for any performance metric $\mathcal{M}$ as:

$$IBA_\alpha(\mathcal{M}) = (1 + \alpha \cdot Dom) \cdot \mathcal{M} \tag{12}$$

where *Dom*, called *dominance*, is defined as $Dom = TPrate - TNrate$ within the range $[-1,+1]$, and it is weighted by $\alpha \geqslant 0$ to reduce its influence on the result of the particular metric $\mathcal{M}$.

The dominance is here used to estimate the relationship between TPrate and TNrate. The closer the dominance is to 0, the more balanced both individual class accuracies are. The weighting factor $(1 + \alpha \cdot Dom)$ in Eq. 12 is within the range $[1 - \alpha, 1 + \alpha]$. Note that if $\alpha = 0$ or $TPrate = TNrate$, the IBA_α turns into the measure $\mathcal{M}$. In practice, one should select a value of α depending on the metric used. In the present paper, we will utilize $\mathcal{M} = Gmean^2$ and $\alpha = 0.1$.

As can be seen, the IBA metric quantifies a certain trade-off between a measure of overall accuracy (here, we will use Gmean) and

Table 2
Data sets used in the experiments.

Data set	Positive examples	Negative examples	Classes	Majority class	Source
Breast	81	196	2	1	UCI[a]
Ecoli	35	301	8	1–3,5–8	UCI
German	300	700	2	1	UCI
Glass	17	197	9	1,2,4–9	UCI
Haberman	81	225	2	1	UCI
Laryngeal2	53	639	2	1	Library[b]
Letter-A	789	19,211	26	2–26	UCI
Phoneme	1586	3818	2	1	UCI
Optidigts	554	5066	10	1–8,10	UCI
Pendigits	1055	9937	10	1–5,7–10	UCI
Pima	268	500	2	1	UCI
Satimage	626	5809	7	1–3,5–7	UCI
Scrapie	531	2582	2	1	Library
Segmentation	330	1980	6	1–4,6	UCI
Spambase	1813	2788	2	1	UCI
Vehicle	212	634	4	2,3,4	UCI
Yeast	429	1055	10	1,3–10	UCI

[a] UCI Machine Learning Database Repository http://archive.ics.uci.edu/ml/.
[b] Library http://www.vision.uji.es/~sanchez/Databases/.

an index of how balanced the two class accuracies are (the dominance index). Unlike most performance metrics, the IBA function not only takes care of the overall accuracy but also intends to favor classifiers with better results on the positive class (generally, the most important class).

4. Experimental set-up

The empirical study was directed to determine the influence of the imbalance ratio on the performance of under and over-sampling techniques using a variety of learning methods and several

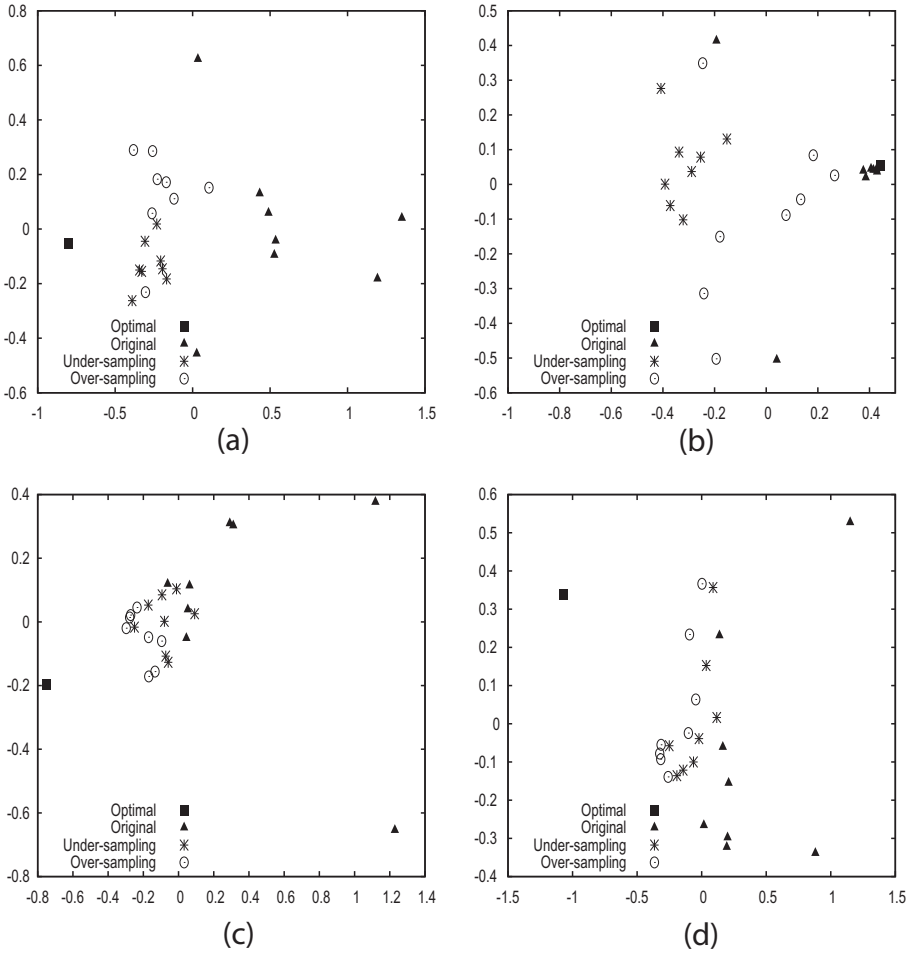

Fig. 1. MDS plots for the strongly/moderately imbalanced data sets when evaluated with (a) TPrate, (b) TNrate, (c) Gmean and, (d) IBA.

Table 3
Euclidean distances to the optimal point in the MDS space for data sets with a severe/moderate imbalance.

	TPrate			TNrate			Gmean			IBA		
	Under	Over	Original	Under	Over	Original	Under	Over	Original	Under	Over	Original
1-NN	0.4603	0.5265	0.9201	0.8796	0.7500	0.7304	0.6277	0.5665	0.7552	0.9953	0.9376	1.2408
7-NN	0.4828	0.6685	1.3279	0.7821	0.3255	0.0394	0.7119	0.5235	1.1556	1.0316	0.8651	1.4239
13-NN	0.4669	0.6185	1.3359	0.8375	0.3943	0.0319	0.7953	0.5154	1.1706	1.0952	0.8503	1.4179
MLP	0.4929	0.6993	1.2464	0.6009	0.2632	0.0699	0.5294	0.4842	0.8676	0.9068	0.8488	1.2936
SVM	0.5972	0.5406	1.9972	0.7815	0.8471	0.0240	0.8684	0.6661	1.9523	1.2268	1.0570	2.0619
NBC	0.5731	0.6388	1.0732	0.8237	0.7786	0.6905	0.6926	0.6158	0.8057	1.1554	1.0698	1.2096
J48	0.6453	0.9284	1.2948	0.6983	0.1813	0.0685	0.6966	0.5965	0.8344	1.1118	1.0296	1.3676
RBF	0.6123	0.5489	2.1522	0.7327	0.6564	0.0216	0.6812	0.5792	2.0290	1.1177	0.9781	2.2251
Average	0.5414	0.6462	1.4185	0.7670	0.5246	0.2095	0.7004	0.5684	1.1963	1.0801	0.9545	1.5301

Table 4
Index of performance using Gmean for strongly/moderately imbalanced data sets.

	Wins	Losses	Index	Wins	Losses	Index	Wins	Losses	Index	Wins	Losses	Index
	1-NN			7-NN			13-NN			J48		
SMOTE	22	0	22	23	1	22	24	4	20	18	7	11
gg-SMOTE	20	3	17	21	3	18	24	4	20	23	3	20
RUS	8	17	−9	11	15	−4	12	17	−5	18	6	12
WE+MSS	7	16	−9	6	21	−15	8	24	−16	5	25	−20
Original	4	25	−21	5	26	−21	8	27	−19	4	27	−23
	MLP			NBC			RBF			SVM		
SMOTE	24	1	23	14	7	7	20	7	13	20	0	20
gg-SMOTE	20	4	16	13	6	7	19	5	14	20	1	19
RUS	16	8	8	12	8	4	20	7	13	14	10	4
WE+MSS	5	25	−20	3	23	−20	12	19	−7	9	17	−8
Original	3	30	−27	13	11	2	1	34	−33	0	35	−35

Table 5
Index of performance using IBA for strongly/moderately imbalanced data sets.

	Wins	Losses	Index	Wins	Losses	Index	Wins	Losses	Index	Wins	Losses	Index
	1-NN			7-NN			13-NN			J48		
SMOTE	22	0	22	23	1	22	24	3	21	7	8	9
gg-SMOTE	20	3	17	22	3	19	23	4	19	21	4	17
RUS	8	17	−9	14	16	−2	11	15	−4	19	3	16
WE+MSS	7	16	−9	7	22	−15	8	23	−15	6	25	−19
Original	4	25	−21	5	29	−24	6	27	−21	4	27	−23
	MLP			NBC			RBF			SVM		
SMOTE	22	2	20	14	6	8	22	2	20	21	2	19
gg-SMOTE	21	3	18	11	7	4	21	3	18	20	2	18
RUS	16	6	10	15	5	10	16	6	10	15	8	7
WE+MSS	5	24	−19	1	25	−24	5	24	−19	8	19	−11
Original	1	30	−29	12	10	2	1	34	−33	0	35	−35

metrics. According to this aim, experiments were conducted as follows:

Data sets: 17 real data sets (summary of whom is given in Table 2) were employed in the experiment. All data sets were transformed into two-class problems by keeping one original class and joining the objects of the remaining classes. The fifth column in Table 2 indicates the original classes that have been joined to shape the majority class. For example, in Vehicle database the objects of classes 2, 3, and 4 were combined to form a unique majority class and the original class 1 was left as the minority class.
Partitions: For each database, 5 independent runs of a stratified 10-fold cross validation were performed. Therefore, using 5 runs of 10-fold cross validation and 17 databases, a total of 850 training sets were used in our experiments.
Resampling strategies: two under-sampling algorithms, random (RUS) and the combination of Wilson's editing with MSS condensing over the negative instances (WE+MSS) [14], and two over-sampling techniques, SMOTE and the Gabriel-graph-based SMOTE (gg-SMOTE) [30], were employed.
Classifiers: the *k*-nearest neighbours (1,7,13-NN) rule, a multilayer perceptron (MLP), a support vector machine (SVM), the Naive Bayes classifier (NBC), a decision tree (J48), and a radial basis function network (RBF) were applied, all of them taken from the Weka toolkit [49] using the default parameter settings. In order to run the NBC on the data sets here considered, the numeric attributes were modeled by a normal distribution.
Performance metrics: TPrate, TNrate, IBA and Gmean were calculated to measure the classification performance.

Classifiers were applied to sets that were preprocessed by the different resampling strategies and also to each original training set, providing a baseline for comparison. Since we were more interested in comparing under and over-sampling than in devising the best algorithm, the results of the two under-sampling and the two over-sampling techniques were averaged. For each database we had 24 models, which came from 3 strategies (original, under

and over) and 8 classifiers, and for each model we obtained its score on each of the four performance metrics. For comparison purposes, we also included the optimal value of each measure, which represents the perfect classification over each database, thus giving a total number of 25 models.

We used multidimensional scaling (MDS) to interpret the results [50]. In a similar way to the work by Caruana and Niculescu-Mizil [51], for each metric, we built a $25 \times D$ table, where D denotes the number of databases. Each entry (i,j) in the table represents the score of the model i on the database j. We calculated the Euclidean distance between each pair of rows, and then performed multidimensional scaling on the matrix of these pairwise distances between models in order to obtain a projection onto a 2-dimensional space.

A second analysis of the results aimed at evaluating the performance of each individual strategy by pairwise comparisons for each classifier. Results obtained in terms of IBA and Gmean were evaluated by a paired t-test between each pair of strategies, for each data set. Based on these values, we computed an index of performance as the difference between *wins* and *losses*, where *wins* is the total number of times that a technique A has been significantly better than another and *losses* is the total number of times that A has been significantly worse than another strategy, with a confidence interval of 95%.

5. Experimental results

Databases were divided into two collections according to the imbalance ratio. A first group will be deemed as strongly/moderately imbalanced, including Ecoli, Glass, Laryngeal2, Letter-A, Optdigits, Pendigits, Satimage, Scrapie, and Segmentation. The second group will consist of those databases whose imbalance could be considered as low: Breast, German, Haberman, Phoneme, Pima, Spambase, Vehicle, and Yeast. After ordering all databases by the imbalance ratio, the cut-off (between both groups) was chosen where the maximum difference between two consecutive values occurred.

We performed both the MDS analysis and the significance test separately for the databases with a severe or moderate imbalance and those with a low imbalance since the difficulty of the learning process increases with the imbalance ratio.

5.1. Results on data sets with a severe/moderate imbalance

Fig. 1 illustrates the MDS results for those data sets that present high imbalance levels. As expected, Fig. 1(a) shows that the TPrate given by the resampling strategies are closer to the optimal value than that of the original set, irrespectively of the classifier used. Conversely, the TNrate in Fig. 1(b) presents the opposite behaviour, that is, the original training set is nearer the optimal point (that represents the perfect classification) than any resampling technique. These results are even clearer in Table 3, where the Euclidean distance of each strategy to the optimal point in the MDS space is reported.

Focusing on the overall performance metrics, Gmean in Fig. 1(c) and IBA in Fig. 1(d), one can observe that resampling is better than using the original training set. When comparing over and under-sampling, it seems that the generation of artificial positive instances excels the removal of negative examples. The distances reported in Table 3 corroborate that over-sampling behaves better than under-sampling for all classifiers when databases with a significant imbalance are considered. As can be seen, the most important differences appear when using the 7-NN, 13-NN and SVM classifiers.

As a further confirmation of the findings using the MDS analysis, Tables 4 and 5 report the index of performance calculated as described in Section 4 for the Gmean and IBA metrics respectively, for each combination of resampling method and classifier. These results demonstrate that in most cases, both over-sampling techniques here used (SMOTE and gg-SMOTE) are significantly better than the two under-sampling algorithms (RUS and WE+MSS) in terms of the index of performance. Also, it seems clear that the use of the original training set (without any preprocessing) corresponds to the worst option to deal with strongly/moderately imbalanced data.

From the results in Tables 4 and 5, it is also possible to remark that differences between SMOTE and gg-SMOTE are marginal, in the sense that both algorithms generally achieve similar indices of performance, especially when using the Gmean measure. Nevertheless, with the IBA metric, the index of performance for SMOTE is higher than that of gg-SMOTE over all classifiers, except the J48 decision tree. Finally, it is interesting to point out that in most cases, the random under-sampling scheme outperforms the "intelligent" WE+MSS algorithm, since this produces a drastic reduction on the majority class.

Results on the data sets with high imbalance have shown that over-sampling is more suitable for handling important differences between the majority and minority classes. Under-sampling tries to balance the size of both classes by removing negative examples, what in this context may give rise to a huge loss of potentially discriminant data.

5.2. Results on data sets with a low imbalance

Fig. 2 provides the MDS plots for the data sets with a low imbalance. As can be seen, the results for TPrate and TNrate are very similar to the case of the strongly/moderately imbalanced data sets: the resampling techniques are closer to the optimal TPrate value than the original training set, whereas this is nearer the optimal TNrate than the under and over-sampling strategies.

When analyzing the overall performance metrics Gmean in Fig. 2(c) and IBA in Fig. 2(d), both under and over-sampling outperform the original training set, but it is difficult to suggest the best strategy. However, a more detailed analysis on Table 6 reveals a slightly better behaviour of the under-sampling techniques: 7 out of 8 classifiers in the case of Gmean and 5 out of 8 for IBA agree with this assertion.

Tables 7 and 8 provide the indices of performance for the data sets with a low imbalance level. As can be seen, both under and over-sampling obtain similar results, especially in terms of *wins* (the number of times that a technique has overcome another), but always better than those of using the original training set. This suggests that in this case, effectiveness of a particular resampling method depends equally on the class imbalance as well as other data characteristics (overlapping, dimensionality, density, small disjuncts, etc.) [19,20,52–55].

Therefore, instead of focusing only on class imbalance, a broader analysis of the data complexity should be done to explain why some strategy is better for a particular data set [56]. The study could be addressed to clear up, given a characterization of the original data distribution, how the quality of the learning process could be affected by the potential overfitting that over-sampling may cause or by the loss of information produced by under-sampling.

6. Conclusions and further extensions

This paper has presented a thorough empirical analysis of the effect of the imbalance ratio (the ratio between minority and majority classes) and the classifier on the effectiveness of resampling strategies when dealing with imbalanced data. The

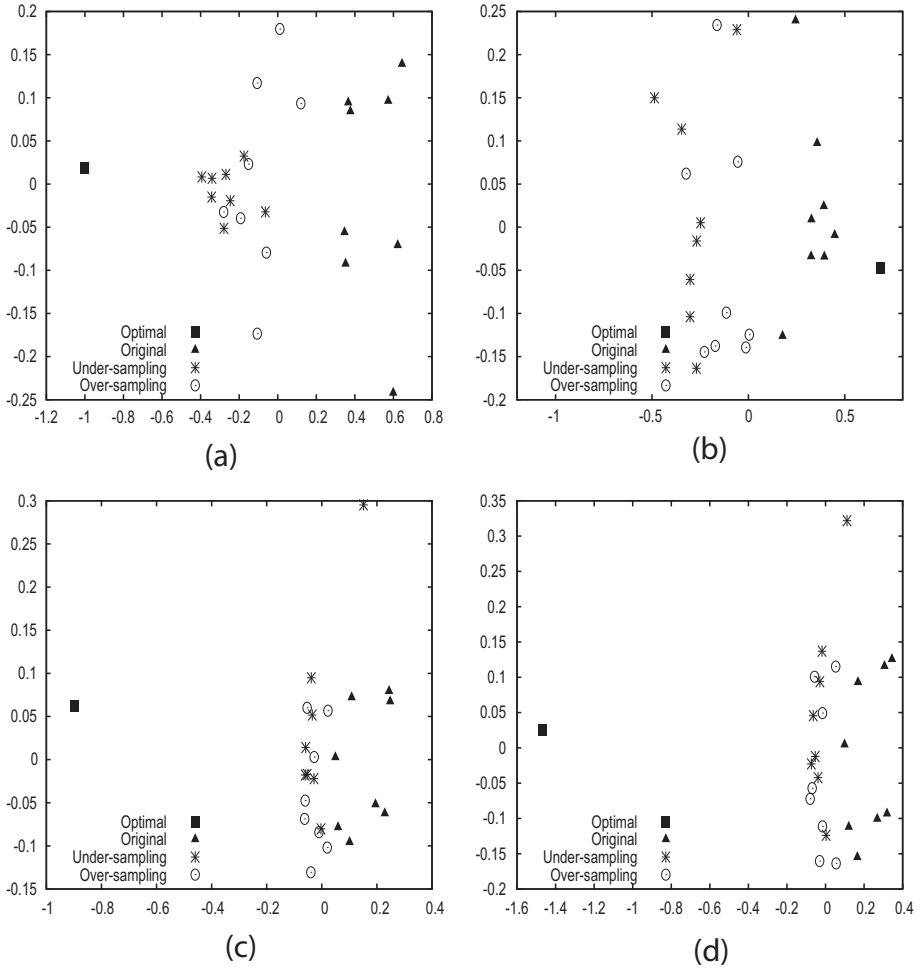

Fig. 2. MDS plots for data sets with a low imbalance when evaluated with (a) TPrate, (b) TNrate, (c) Gmean and, (d) IBA.

Table 6
Distances to the optimal point in the MDS space for data sets with a low imbalance.

	TPrate			TNrate			Gmean			IBA		
	Under	Over	Original	Under	Over	Original	Under	Over	Original	Under	Over	Original
1-NN	0.8268	1.1240	1.3695	0.9361	0.6861	0.5153	0.9045	0.9305	1.0095	1.4772	1.5330	1.6411
7-NN	0.7263	0.8502	1.5765	0.9905	0.9195	0.2928	0.8721	0.8979	1.0970	1.4291	1.4570	1.7392
13-NN	0.6604	0.8114	1.6508	0.9887	0.8629	0.2415	0.8397	0.8428	1.1318	1.3935	1.3898	1.7892
MLP	0.7329	0.9003	1.3502	0.9547	0.8022	0.3637	0.8390	0.8440	0.9473	1.4027	1.3994	1.5650
SVM	0.6080	0.9160	1.6220	1.1896	0.7516	0.3037	1.0733	0.8430	1.1444	1.6054	1.4113	1.7744
NBC	0.9388	0.9479	1.3581	0.7951	0.8945	0.5265	0.8585	0.9185	1.0046	1.4528	1.5219	1.6363
J48	0.7566	1.0252	1.3801	0.9354	0.7058	0.3615	0.8465	0.8779	0.9654	1.4134	1.4479	1.5929
RBF	0.6608	0.7230	1.6264	1.0446	1.0151	0.3601	0.8616	0.8704	1.1405	1.4390	1.4506	1.8137
Average	0.7388	0.9123	1.4917	0.9793	0.8297	0.3706	0.8869	0.8781	1.0551	1.4516	1.4514	1.6940

experiments have used 17 data sets (8 strongly or moderately imbalanced and 9 with a low imbalance), 8 classifiers and 4 resampling methods. Besides, the performance has been evaluated by means of four different metrics.

Experimental results have shown that, in general, over-sampling performs better than under-sampling for data sets with a severe class imbalance. This result can be explained by the fact that under-sampling may throw out too many negative examples in order to balance the size of both classes, thus producing a very considerable loss of potentially important information for the learner. When imbalance is low, the results have suggested that both over and under-sampling give similar performance, thus making necessary a further data complexity analysis to choose a suitable resampling technique for a particular imbalanced data set. Besides, the use of the original training sets without any preprocessing has been demonstrated to be clearly worse than resampling.

On the other hand, experiments have revealed that the characteristics of the classifier have little influence on the effectiveness of the different resampling strategies. In this sense, it seems to be more important the way of preprocessing the data than the learner used for classification.

The present work has open some interesting research avenues with regards to the resampling strategies for imbalanced data sets, such as: (i) To analyse the sets by means of data complexity (or problem difficulty) measures, thus obtaining a better description of data and allowing a more accurate application of specific

Table 7
Index of performance using Gmean for data sets with a low imbalance.

	Wins	Losses	Index	Wins	Losses	Index	Wins	Losses	Index	Wins	Losses	Index
	1-NN			7-NN			13-NN			J48		
SMOTE	12	5	7	16	3	13	13	3	10	13	3	10
gg-SMOTE	14	2	12	20	0	20	20	1	19	14	1	13
RUS	8	7	1	11	7	4	14	7	7	15	4	11
WE+MSS	11	6	5	4	15	−11	5	17	−12	5	19	−14
Original	2	27	−25	1	27	−26	1	29	−28	2	23	−21
	MLP			NBC			RBF			SVM		
SMOTE	11	1	10	11	5	6	11	3	8	18	0	18
gg-SMOTE	14	0	14	8	7	1	12	3	9	18	0	18
RUS	9	4	5	11	4	7	14	1	13	13	5	8
WE+MSS	4	14	−10	13	9	4	12	12	0	6	23	−17
Original	3	21	−18	5	23	−18	0	30	−30	2	29	−27

Table 8
Index of performance using IBA for data sets with a low imbalance.

	Wins	Losses	Index	Wins	Losses	Index	Wins	Losses	Index	Wins	Losses	Index
	1-NN			7-NN			13-NN			J48		
SMOTE	13	6	7	16	1	15	17	2	15	14	3	11
gg-SMOTE	14	4	10	18	0	18	19	0	19	14	2	12
RUS	8	8	0	10	7	3	12	7	5	14	4	10
WE+MSS	15	5	10	7	14	−7	7	16	−9	7	14	−7
Original	2	27	−25	1	30	−29	1	31	−30	1	26	−25
	MLP			NBC			RBF			SVM		
SMOTE	10	1	9	11	5	6	11	3	8	16	0	16
gg-SMOTE	12	0	12	9	7	2	11	3	8	17	0	17
RUS	9	4	5	10	4	6	13	1	12	13	5	8
WE+MSS	7	12	−5	13	8	5	12	9	3	7	20	−13
Original	2	23	−21	5	24	−19	0	31	−31	2	30	−28

techniques to tackle the class imbalance problem; (ii) To extend this study to data sets with multiple minority classes; and (iii) To take cost-sensitive learning into consideration within the present analysis.

Acknowledgments

This work has partially been supported by the Spanish Ministry of Education and Science under Grants CSD2007-00018 and TIN2009-14205, and by Fundació Caixa Castelló – Bancaixa under Grant P1-1B2009-04.

References

[1] H. He, E.A. Garcia, Learning from imbalanced data, IEEE Transactions on Knowledge and Data Engineering 21 (9) (2009) 1263–1284.
[2] N. Japkowicz, S. Stephen, The class imbalance problem: a systematic study, Intelligent Data Analysis 6 (5) (2002) 429–449.
[3] J. Liu, Q. Hu, D. Yu, A comparative study on rough set based class imbalance learning, Knowledge-Based Systems 21 (8) (2008) 753–763.
[4] G. Cohen, M. Hilario, H. Sax, S. Hugonnet, A. Geissbuhler, Learning from imbalanced data in surveillance of nosocomial infection, Artificial Intelligence in Medicine 37 (1) (2006) 7–18.
[5] T. Fawcett, F. Provost, Adaptive fraud detection, Data Mininng and Knowledge Discovery 1 (3) (1997) 291–316.
[6] C.S. Hilas, P.A. Mastorocostas, An application of supervised and unsupervised learning approaches to telecommunications fraud detection, Knowledge-Based Systems 21 (7) (2008) 721–726.
[7] P.K. Chan, F. Wei, A. Prodromidis, S.J. Stolfo, Distributed data mining in credit card fraud detection, IEEE Intelligent Systems 14 (6) (1999) 67–74.
[8] M. Kubat, R.C. Holte, S. Matwin, Machine learning for the detection of oil spills in satellite radar images, Machine Learning 30 (2–3) (1998) 195–215.
[9] S. Tan, Neighbor-weighted K-nearest neighbor for unbalanced text corpus, Expert Systems with Applications 28 (4) (2005) 667–671.
[10] Z. Zheng, X. Wu, R. Srihari, Feature selection for text categorization on imbalanced data, SIGKDD Explorations Newsletter 6 (1) (2004) 80–89.
[11] Y.-M. Huang, C.-M. Hung, H.C. Jiau, Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem, Nonlinear Analysis: Real World Applications 7 (4) (2006) 720–757.
[12] S. Daskalaki, I. Kopanas, N. Avouris, Evaluation of classifiers for an uneven class distribution problem, Applied Artificial Intelligence 20 (5) (2006) 381–417.
[13] N. García-Pedrajas, J. Pérez-Rodríguez, M. García-Pedrajas, D. Ortiz-Boyer, C. Fyfe, Class imbalance methods for translation initiation site recognition in DNA sequences, Knowledge-Based Systems 25 (1) (2012) 22–34.
[14] R. Barandela, J.S. Sánchez, V. García, E. Rangel, Strategies for learning in class imbalance problems, Pattern Recognition 36 (3) (2003) 849–851.
[15] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, Journal of Artificial Intelligence Research 16 (2002) 321–357.
[16] S. García, J. Derrac, I. Triguero, C.J. Carmona, F. Herrera, Evolutionary-based selection of generalized instances for imbalanced classification, Knowledge-Based Systems 25 (1) (2012) 3–12.
[17] H. Jin, C.X. Ling, Using AUC and accuracy in evaluating learning algorithms, IEEE Transactions on Knowledge and Data Engineering 17 (3) (2005) 299–310.
[18] R. Ranawana, V. Palade, Optimized precision – a new measure for classifier performance evaluation, in: Proceedings of the IEEE Congress on Computational Intelligence, Vancouver, Canada, 2006, pp. 2254–2261.
[19] T.K. Jo, N. Japkowicz, Class imbalances versus small disjuncts, SIGKDD Explorations Newsletter 6 (1) (2004) 40–49.
[20] R.C. Prati, G.E.A.P.A. Batista, M.C. Monard, Learning with class skews and small disjuncts, in: Proceedings of the 17th Brazilian Symposium on Artificial Intelligence, Sao Luiz, Brazil, 2004, pp. 296–306.
[21] X.-Y. Liu, J. Wu, Z.-H. Zhou, Exploratory undersampling for class-imbalance learning, IEEE Transactions on Systems, Man, and Cybernetics – Part B 39 (2) (2009) 539–550.
[22] J.V. Hulse, T.M. Khoshgoftaar, A. Napolitano, Experimental perspectives on learning from imbalanced data, in: Proceedings of the 24th International Conference on Machine Learning, Corvalis, Oregon, 2007, pp. 935–942.
[23] G.E.A.P.A. Batista, R.C. Prati, M.C. Monard, A study of the behavior of several methods for balancing machine learning training data, ACM SIGKDD Explorations Newsletter 6 (1) (2004) 20–29.
[24] R. Barandela, R.M. Valdovinos, J.S. Sánchez, F.J. Ferri, The imbalance training sample problem: under or over sampling, in: Structural, Syntactic, and Statistical Pattern Recognition, Springer-Verlag, 2004, pp. 806–814.
[25] A. Estabrooks, T. Jo, N. Japkowicz, A multiple resampling method for learning from imbalanced data sets, Computational Intelligence 20 (1) (2004) 18–36.

[26] J. Yang, X. Yu, Z.-Q. Xie, J.-P. Zhang, A novel virtual sample generation method based on Gaussian distribution, Knowledge-Based Systems 24 (6) (2011) 740–748.
[27] G.M. Weiss, F.J. Provost, Learning when training data are costly: the effect of class distribution on tree induction, Journal of Artificial Intelligence Research 19 (2003) 315–354.
[28] Y. Liu, A. An, X. Huang, Boosting prediction accuracy on imbalanced datasets with SVM ensembles, in: Proceedings of the 10th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Singapore, 2006, pp. 107–118.
[29] N.V. Chawla, A. Lazarevic, L.O. Hall, K.W. Bowyer, SMOTEBoost: improving prediction of the minority class in boosting, in: Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases, Dubrovnik, Croatia, 2003, pp. 107–119.
[30] V. García, J.S. Sánchez, R.A. Mollineda, On the use of surrounding neighbors for synthetic over-sampling of the minority class, in: Proceedings of the 8th WSEAS International Conference on Simulation, Modelling and Optimization, Santander, Spain, 2008, pp. 389–394.
[31] H. Han, W.-Y. Wang, B.-H. Mao, Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning, in: Proceedings of the 1th International Conference on Intelligent Computing, Hefei, China, 2005, pp. 878–887.
[32] S. Hu, Y. Liang, L. Ma, Y. He, MSMOTE: improving classification performance when training data is imbalanced, in: Proceedings of the 2nd International Workshop on Computer Science and Engineering, Qingdao, China, 2009, pp. 13–17.
[33] G. Hongyu, V.L. Herna, Learning from imbalanced data sets with boosting and data generation: the DataBoost-IM approach, SIGKDD Explorations Newsletter 6 (1) (2004) 30–39.
[34] J. Zhang, I. Mani, kNN approach to unbalanced data distributions: a case study involving information extraction, in: Proceedings of the Workshop on Learning from Imbalanced Datasets, Washington DC, USA, 2003.
[35] M. Kubat, S. Matwin, Addressing the curse of imbalanced training sets: one-sided selection, in: Proceedings of the 14th International Conference on Machine Learning, Nashville, USA, 1997, p. 179–186.
[36] I. Tomek, Two modifications of CNN, IEEE Transactions on Systems, Man and Cybernetics 6 (11) (1976) 769–772.
[37] P.E. Hart, The condensed nearest negihbor rule, IEEE Transactions on Information Theory 14 (1968) 515–516.
[38] J. Laurikkala, Improving identification of difficult small classes by balancing class distribution, in: Proceedings of the 8th Conference on Artificial Intelligence in Medicine, Cascais, Portugal, 2001, pp. 63–66.
[39] D.L. Wilson, Asymptotic properties of nearest neighbour rules using edited data, IEEE Transactions on Systems, Man and Cybernetics 2 (1972) 408–421.
[40] R. Barandela, F.J. Ferri, J.S. Sánchez, Decision boundary preserving prototype selection for nearest neighbor classification, International Journal of Pattern Recognition and Artificial Intelligence 19 (6) (2005) 787–806.
[41] S.-J. Yen, Y.-S. Lee, C.-H. Lin, J.-C. Ying, Investigating the effect of sampling methods for imbalanced data distributions, in: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, vol. 5, Taipei, Taiwan, 2006, pp. 4163–4168.
[42] S. García, F. Herrera, Evolutionary undersampling for classification with imbalanced datasets: proposals and taxonomy, Evolutionary Computation 17 (3) (2009) 275–306.
[43] X. Chen, B. Gerlach, D. Casasent, Pruning support vectors for imbalanced data classification, in: Proceedings of the International Joint Conference on Neural Networks, Montreal, Canada, 2005, pp. 1883–1888.
[44] F. Provost, T. Fawcett, Robust classification for imprecise environments, Machine Learning 42 (3) (2001) 203–231.
[45] A.P. Bradley, The use of the area under the ROC curve in the evaluation of machine learning algorithms, Pattern Recognition 30 (7) (1997) 1145–1159.
[46] M. Sokolova, N. Japkowicz, S. Szpakowicz, Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation, in: Proceedings of the 19th ACS Australian Joint Conference on Artificial Intelligence, Hobart, Australia, 2006, pp. 1015–1021.
[47] C.J.V. Rijsbergen, Information Retrieval, Butterworths, London, UK, 1979.
[48] V. García, R.A. Mollineda, J.S. Sánchez, Theoretical analysis of a performance measure for imbalanced data, in: Proceedings of the 20th International Conference on Pattern Recognition, Istanbul, Turkey, 2010, pp. 617–620.
[49] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I.H. Witten, The WEKA data mining software: an update, SIGKDD Explorations Newsletter 11 (2009) 10–18.
[50] R. Alaiz-Rodríguez, N. Japkowicz, P. Tischer, A visualization-based exploratory technique for classifier comparison with respect to multiple metrics and multiple domains, in: Proceedings of the 15th European Conference on Machine Learning, Antwerp, Belgium, 2008, pp. 660–665.
[51] R. Caruana, A. Niculescu-Mizil, Data mining in metric space: an empirical analysis of supervised learning performance criteria, in: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, USA, 2004, pp. 69–78.
[52] V. García, R.A. Mollineda, J.S. Sánchez, On the k-NN performance in a challenging scenario of imbalance and overlapping, Pattern Analysis and Applications 11 (3) (2008) 269–280.
[53] J.V. Hulse, T. Khoshgoftaar, Knowledge discovery from imbalanced and noisy data, Data & Knowledge Engineering 68 (12) (2009) 1513–1542.
[54] T.M. Khoshgoftaar, N. Seliya, D.J. Drown, Evolutionary data analysis for the class imbalance problem, Intelligent Data Analysis 14 (1) (2010) 69–88.
[55] G.M. Weiss, The impact of small disjuncts on classifier learning, in: R. Stahlbock, S.F. Crone, S. Lessmann (Eds.), Data Mining, Annals of Information Systems, vol. 8, Springer, US, 2010, pp. 193–226 (chapter 9).
[56] J. Luengo, A. Fernández, S. García, F. Herrera, Addressing data complexity for imbalanced data sets: analysis of SMOTE-based oversampling and evolutionary undersampling, Soft Computing – A Fusion of Foundations, Methodologies and Applications, in press, doi:10.1007/s00500-010-0625-8.