Journal of the Operational Research Society (2012) 1–11

www.palgrave-journals.com/jors/

On the suitability of resampling techniques for the class imbalance problem in credit scoring

AI Marqués, V García and JS Sánchez*
Universitat Jaume I, Castellón, Spain

In real-life credit scoring applications, the case in which the class of defaulters is under-represented in comparison with the class of non-defaulters is a very common situation, but it has still received little attention. The present paper investigates the suitability and performance of several resampling techniques when applied in conjunction with statistical and artificial intelligence prediction models over five real-world credit data sets, which have artificially been modified to derive different imbalance ratios (proportion of defaulters and non-defaulters examples). Experimental results demonstrate that the use of resampling methods consistently improves the performance given by the original imbalanced data. Besides, it is also important to note that in general, over-sampling techniques perform better than any under-sampling approach.

Journal of the Operational Research Society advance online publication, 3 October 2012
doi:10.1057/jors.2012.120

Keywords: credit scoring; class imbalance; resampling; logistic regression; support vector machine

1. Introduction

The recent world financial crisis has aroused increasing attention of banks and financial institutions on credit risk assessment, converting this into a key task because of the heavy losses associated with wrong decisions. One major risk comes from the difficulty to distinguish the creditworthy applicants from those who will probably default on repayments. In this context, credit scoring has been identified as a crucial tool to evaluate credit risk, improve cash flow, reduce possible risks and make managerial decisions (Thomas *et al*, 2002; Abrahams and Zhang, 2008), and one of the most popular application fields for both data mining and operational research (Baesens *et al*, 2009).

In practice, the process of credit scoring can be deemed as a prediction problem where a new input sample (the credit applicant) must be categorized into one of the predefined classes (in general, 'good' applicants and 'bad' applicants, depending on how likely they are to default with their repayments) based on a number of observed variables or attributes related to that sample. The input to the model consists of a variety of information that describes socio-demographic characteristics and economic conditions of the applicant, and the prediction method will produce the output in terms of the applicant creditworthiness.

The most classical approaches to credit scoring are based on parametric statistical models, such as discriminant analysis and logistic regression. However, most recent research has been addressed to implement solutions with non-parametric methods and computational intelligence techniques: decision trees, artificial neural networks, support vector machines, evolutionary algorithms, etc.

From the many comparative studies carried out (Baesens *et al*, 2003; Huang *et al*, 2004; Xiao *et al*, 2006; Wang *et al*, 2011), it is not possible to claim the superiority of a method over other competing algorithms regardless of data characteristics. For instance, noisy samples, missing values, skewed class distribution and attribute relevance may significantly affect the success of most prediction models.

This paper focuses on one of the data characteristics that may have most influence on the performance of classification techniques: the *imbalance in class distribution* (Japkowicz and Stephen, 2002; Chawla *et al*, 2004; He and Garcia, 2009). While some complexities have been widely studied in the credit scoring literature (eg, attribute relevance), the class imbalance problem has received relatively little attention so far. Nevertheless, imbalanced class distribution naturally happens in credit scoring where, in general, the number of observations in the class of defaulters is much smaller than the number of cases belonging to the class of non-defaulters (Pluto and Tasche, 2006).

In this paper, we conduct an experimental study over real-life credit scoring data sets using seven resampling algorithms to handle the class imbalance problem and two well-established prediction models (logistic regression and support vector machine). All techniques are evaluated in

*Correspondence: JS Sanchez, Computer Languages and Systems, Universitat Jaume I, Av. Sos Baynat, s/n, Castellon 12071, Spain.
E-mail: sanchez@uji.es

terms of their area under the ROC curve (AUC), and then compared for statistical differences using the Friedman's average rank test and a *post hoc* test. The aim of this study is to determine whether or not the resampling strategies are suitable to deal with the class imbalance problem, and to which extent different levels of imbalance affect the performance of each method.

2. Related works

Class imbalance hinders the performance of most standard classification systems, which assume a relatively well-balanced class distribution and equal misclassification costs (Japkowicz and Stephen, 2002). The class imbalance problem occurs when one class vastly outnumbers the other class, which is usually the most important one and with the highest misclassification costs (Chawla *et al*, 2008). Instances from the minority and majority classes are often referred to as positive and negative, respectively.

2.1. Class imbalance in credit scoring

As already mentioned, imbalanced class distribution happens in many credit scoring applications. For example, it is common to find that defaulters constitute less than 10% of the database. This is the main reason why the class imbalance problem has attracted growing attention in the literature, both to detect fraudulent financial activities and to predict creditworthiness of credit applicants.

In the credit scoring domain, research has mainly focused on analysing the behaviour of prediction models, showing that the performance on the minority class drops down significantly as the imbalance ratio increases (Hand and Vinciotti, 2003; Kennedy *et al*, 2010; Bhattacharyya *et al*, 2011; Brown and Mues, 2012). However, only a few works have been addressed to design solutions for imbalanced credit data sets. For example, Vinciotti and Hand (2003) introduced a modification to straightforward logistic regression by taking into account the misclassification costs when the probability estimates are made. Huang *et al* (2006) proposed two strategies for classification and cleaning of skewed credit data. One method involves randomly selecting instances to balance the proportion of examples in each class, whereas the second method consists of combining the ID3 decision tree and the PRISM filter.

An algorithmic level solution corresponds to the proposal by Yao (2009), who carried out a systematic comparative study on three weighted classifiers: C4.5 decision tree, support vector machine and rough sets. The experiments over two credit scoring data sets showed that the weighted methods outperform those standard classifiers in terms of type-I error. Within the PAKDD'2009 data mining competition, Xie *et al* (2009) proposed an ensemble of logistic regression and AdaBoost with the aim of optimizing the AUC for a highly imbalanced credit data set. In the same direction of combining classifiers, Florez-Lopez (2010) employed several cooperative strategies (simple and weighted voting) based on statistical models and computational intelligence techniques in combination with bootstrapping to handle the imbalance problem.

Kennedy *et al* (2010) explored the suitability and performance of various one-class classifiers for several imbalanced credit scoring problems with varying levels of imbalance. The experimental results suggest that the one-class classifiers perform especially well when the minority class constitutes 2% or less of the data, whereas the two-class classifiers are preferred when the minority class represents at least 15% of the data. Tian *et al* (2010) proposed a new method based on the support vector domain description model, showing that this can be effective in ranking and classifying imbalanced credit data.

An exhaustive comparative study of various classification techniques when applied to skewed credit data sets was carried out by Brown and Mues (2012). They progressively increased the levels of class imbalance in each of five real-life data sets by randomly under-sampling the minority class of defaulters, so as to identify to what extent the predictive power of each technique was adversely affected. The results showed that traditional models, such as logistic regression and linear discriminant analysis, are fairly robust to imbalanced class sizes.

3. Resampling methods

Much work has been done to deal with the class imbalance problem, at both data and algorithmic levels. At the data level, the most popular strategies consist of applying different forms of resampling to change the class distribution of the data. This can be done by either over-sampling the minority class or under-sampling the majority class until both classes are approximately equally represented.

Both data level solutions present several drawbacks because they artificially alter the original class distribution. While under-sampling may result in throwing away potentially useful information about the majority class, over-sampling worsens the computational burden of some learning algorithms and creates noise that could result in a loss of performance (Barandela *et al*, 2003).

At the algorithmic level, solutions include internally biasing the discrimination-based process, assigning distinct costs to the classification errors and learning from one class. Conclusions about what is the best solution for the class imbalance problem are divergent. However, the data level methods are the most investigated because they are independent of the underlying classifier and can be easily implemented for any problem. Hence, the present study will concentrate on a number of resampling strategies.

3.1. Over-sampling

The simplest strategy to expand 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. Nevertheless, this method may increase the likelihood of overfitting since it makes exact copies of the minority class instances.

In order to avoid overfitting, Chawla *et al* (2002) proposed a technique, called Synthetic Minority Over-sampling TEchnique (SMOTE), to up-size the minority class. Instead of merely replicating cases belonging to the minority class, this algorithm generates artificial examples from the minority class by interpolating existing instances that lie close together. It first finds the k nearest neighbours belonging to the minority class for each positive example and then, the synthetic examples are generated in the direction of some or all of those nearest neighbours. SMOTE allows the classifier to build larger decision regions that contain nearby instances from the minority class. Depending upon the amount of over-sampling required, a number of neighbours from the k nearest neighbours are randomly chosen (in the experiments reported in the original paper, k was set to 5). When, for example, the amount of over-sampling needed is 200%, only two neighbours from the k nearest neighbours are chosen and then one synthetic prototype is generated in the direction of each of these two neighbours.

Although SMOTE has proved to be an effective tool for handling the class imbalance problem, it may over-generalize the minority class as it does not take care of the distribution of majority class neighbours. As a result, SMOTE generation of synthetic examples may increase the overlapping between classes (Maciejewski and Stefanowski, 2011). Numerous modifications to the original SMOTE have been proposed in the literature, most of them pursuing to determine the region in which the positive examples should be generated. Thus, the Safe-Level SMOTE (SL-SMOTE) algorithm (Bunkhumpornpat *et al*, 2009) calculates a 'safe level' coefficient (sl) for each example from the minority class, which is defined as the number of other minority class instances among its k neighbours. If the coefficient sl is equal or close to 0, such an example is considered as noise; if sl is close to k, then this example may be located in a safe region of the minority class. The idea is to direct the generation of new synthetic examples close to safe regions.

On the other hand, Batista *et al* (2004) proposed a methodology that combines SMOTE and data cleaning, with the aim of reducing the possible overlapping introduced when the synthetic examples from the minority class are generated. In order to create well-defined classes, after over-sampling the minority class by means of SMOTE, the Wilson's editing algorithm (Wilson, 1972) is applied to remove any example (either positive or negative) that is misclassified by its three nearest neighbours. This method is here called SMOTE + WE.

3.2. Under-sampling

Random under-sampling aims at balancing the data set through the random removal of examples from the majority class. Despite its simplicity, it has empirically been shown to be one of the most effective resampling methods. However, the major problem of this technique is that it may discard data potentially important for the prediction process. In order to overcome this limitation, other methods have been designed to provide a more intelligent selection strategy. For example, Kubat and Matwin (1997) proposed the One-Sided Selection technique (OSS), which selectively removes only those negative instances that are redundant or noisy (majority class examples that border the minority class). The border examples are detected by using the concept of Tomek links (Tomek, 1976), whereas the redundant cases (those that are distant from the decision boundary) are discovered by means of Hart's condensing (Hart, 1968).

Laurikkala (2001) introduced a new algorithm called Neighbourhood CLeaning rule (NCL) that operates in a similar fashion as OSS. In this case, Wilson's editing is used to remove majority class examples whose class label differs from the class of at least two of its three nearest neighbours. Besides, if a positive instance is misclassified by its three nearest neighbours, then the algorithm also eliminates the neighbours that belong to the majority class.

A quite different alternative corresponds to under-Sampling Based on Clustering (SBC) (Yen and Lee, 2006), which rests on the idea that there may exist different clusters in a given data set, and each cluster may have distinct characteristics depending on the ratio of the number of minority class examples to the number of majority class examples in the cluster. Thus the SBC algorithm first gathers all examples in the data set into some clusters, and then determines the number of majority class examples that will be randomly picked up. Finally, it combines the selected majority class instances and all the minority class examples to obtain a resampled data set.

4. Experiments

The aim of the experiments here carried out is to evaluate the performance of different under- and over-sampling algorithms and investigate to what extent the behaviour of each technique is affected by different levels of imbalance. On the other hand, we also analyse the suitability of each resampling method in function of the type of classifier when addressing the class imbalance problem. To this end, both statistical and artificial intelligence prediction models will be compared.

The resampling algorithms used in the experiments are the over-sampling and under-sampling techniques previously described in Section 2, that is, random over-sampling (ROS), SMOTE, SL-SMOTE, SMOTE + WE, random under-sampling (RUS), OSS, NCL and SBC. The classification methods correspond to two well-known models suitable for credit scoring: logistic regression (logR) and support vector machine (SVM) with a linear kernel. All resampling techniques and both prediction models have been implemented with the KEEL software (Alcalá-Fdez *et al*, 2009), using their default parameters settings.

4.1. Description of the experimental databases

Five real-world credit data sets have been taken to test the performance of the strategies investigated in the present paper. The widely used Australian, German and Japanese data sets are from the UCI Machine Learning Database Repository (http://archive.ics.uci.edu/ml/). The UCSD data set corresponds to a reduced version of a database used in the 2007 Data Mining Contest organized by the University of California San Diego and Fair Isaac Corporation. The Iranian data set (Sabzevari *et al*, 2007) comes from a modification to a corporate client database of a small private bank in Iran.

As we are interested in analysing the impact of different levels of class imbalance on resampling and classification algorithms, each original set has been altered by randomly under-sampling the minority class in order to construct six data sets with varying imbalance ratios (the ratio of the number of minority class examples to the number of majority class examples), $iRatio = \{1{:}4, 1{:}6, 1{:}8, 1{:}10, 1{:}12, 1{:}14\}$. Table 1 reports a summary of the main characteristics of the benchmarking data sets. As can be seen, the Iranian data set has not been modified because of its extremely high imbalance ratio, and it may be interesting to study the behaviour of the resampling techniques under this hard condition. Therefore, we have obtained a total number of 25 data sets for the experiments.

Table 1 Some characteristics of the data sets used in the experiments

Data set (iRatio)	*# Attributes*	*# Good*	*# Bad*
Australian (1:4)	14	307	77
(1:6)	—	—	51
(1:8)	—	—	38
(1:10)	—	—	31
(1:12)	—	—	26
(1:14)	—	—	22
German (1:4)	24	700	175
(1:6)	—	—	117
(1:8)	—	—	88
(1:10)	—	—	70
(1:12)	—	—	58
(1:14)	—	—	50
Japanese (1:4)	15	296	74
(1:6)	—	—	49
(1:8)	—	—	37
(1:10)	—	—	30
(1:12)	—	—	25
(1:14)	—	—	21
UCSD (1:4)	38	1836	459
(1:6)	—	—	306
(1:8)	—	—	230
(1:10)	—	—	184
(1:12)	—	—	153
(1:14)	—	—	131
Iranian (1:19)	27	950	50

4.2. Experimental Protocol

The standard way to assess credit scoring systems is to use a holdout sample since large sets of past applicants are usually available. However, there are situations in which data are too limited to build an accurate scorecard and therefore, other strategies have to be used in order to obtain a good estimate of the classification performance. The most common way around this corresponds to cross-validation (Thomas *et al*, 2002, Ch. 7).

Accordingly, a five-fold cross-validation method has been adopted for the present experiments: each data set in Table 1 has been randomly divided into five stratified parts of equal (or approximately equal) size. For each fold, four blocks have been pooled as the training data, and the remaining part has been employed as an independent test set. Ten repetitions have been run for each trial, giving a total of 50 pairs of training and test sets. Each resampling technique has been applied to each training set, thus obtaining the resampled data sets that have then been used to build the prediction models (logR and SVM). The non-preprocessed training sets have also been employed for model construction. The results from classifying the test samples have been averaged across the 50 runs.

4.3. Evaluation criteria

Standard performance evaluation criteria in the fields of credit scoring include accuracy, error rate, Gini coefficient, Kolmogorov-Smirnov statistic, mean squared error, area under the ROC curve, type-I error and type-II error (Thomas *et al*, 2002; Yang *et al*, 2004; Hand, 2005; Abdou and Pointon, 2011). For a two-class problem, most of these metrics can be easily derived from a 2×2 confusion matrix as that given in Table 2, where each entry (i,j) contains the number of correct/incorrect predictions. For consistency with previous works in the topic of performance measures, the positive and negative classes

Table 2 Confusion matrix for a two-class problem

	Predicted positive	*Predicted negative*
Positive class	True positive (TP)	False negative (FN)
Negative class	False positive (FP)	True negative (TN)

correspond to bad and good applicants (or credit risk), respectively.

Most credit scoring applications often employ the accuracy (or the error rate) as the criterion for performance evaluation. It represents the proportion of the correctly (or wrongly) classified cases (good and bad) on a particular data set. However, empirical and theoretical evidences show that this measure is strongly biased with respect to data imbalance and proportions of correct and incorrect predictions (Provost and Fawcett, 1997). Besides, the accuracy ignores the cost of different error types (bad applicants being predicted as good, or vice versa).

To deal with the class imbalance problem in credit scoring applications, the area under the ROC curve (AUC) has been suggested as an appropriate performance evaluator without regard to class distribution or misclassification costs (Baesens *et al*, 2003) and correspondingly, this has been the evaluation measure adopted for the experiments. For a binary problem, the AUC criterion defined by a single point on the ROC curve is also referred to as balanced accuracy (Sokolova and Lapalme 2009):

$$AUC = \frac{sensitivity + specificity}{2} \tag{1}$$

where $sensitivity = TP/(TP+FN)$ measures the percentage of positive examples that have been predicted correctly, whereas $specificity = TN/(TN+FP)$ corresponds to the percentage of negative instances predicted as negative.

4.4. Statistical significance tests over multiple data sets

Probably, the most common way to compare two or more classifiers over various data sets is the Student's paired *t*-test, which checks whether the average difference in their performance over the data sets is significantly different from zero. However, this appears to be conceptually inappropriate and statistically unsafe because parametric tests are based on the usual assumptions of independence, normality and homogeneity of variance, which are often violated due to the nature of the problems (Demšar, 2006; Zar, 2009; García *et al*, 2010).

In general, the non-parametric tests should be preferred over the parametric ones because they do not assume normal distributions or homogeneity of variance. In this work, we have adopted the Friedman test to determine whether there exist significant differences among the strategies. The process starts by ranking the algorithms for each data set independently according to the AUC results: as there are nine competing strategies, the ranks for each data set will be from 1 (best) to 9 (worst). Then the average rank of each algorithm across all data sets is computed. Under the null hypothesis, which states that all strategies are equivalent and so their average ranks should be equal, the Friedman statistic is distributed according to the χ^2_F distribution with $K-1$ degrees of freedom, K being the number of algorithms.

The Friedman test only can detect significant differences over the whole set of comparisons. For this reason, if the null hypothesis of equivalence of average ranks is rejected, we can then proceed with a *post hoc* test. In particular, the Nemenyi test, which is analogous to the Tukey test for ANOVA, states that the performances of two or more algorithms are significantly different if their average ranks are at least as great as their critical difference (CD) with a given level of significance (α):

$$CD = q_\alpha \sqrt{\frac{K(K+1)}{6N}} \tag{2}$$

where N denotes the number of data sets and q_α is a critical value based on the Studentized range statistic divided by $\sqrt{2}$ (Hochberg and Tamhane, 1987; Demšar, 2006).

5. Results and discussion

To better understand the effect of the class imbalance ratio on the performance of the eight resampling algorithms using the logR and SVM models, we have divided the data sets into two groups: strongly imbalanced databases ($iRatio \geqslant 10$) and those with a low/moderate imbalance ($iRatio < 10$).

5.1. Low/moderate imbalance ratio

Tables 3 and 4 report the AUC values for the data sets with a low/moderate imbalance ratio when using the resampling techniques with the logistic regression and SVM classification models, respectively. The Friedman average ranking of the algorithms ($K=9$) over the data sets ($N=12$) at a significance level of $\alpha=0.05$ is also provided for each classifier, showing that the prediction results using the resampled sets are better than those with the original imbalanced data (except for the SBC algorithm, which achieves the worst AUC values independently of the classifier used).

In general, the over-sampling algorithms outperform the under-sampling techniques, what can be seen by either analysing the average rankings or comparing the AUC of each algorithm over each data set. The best resampling methods correspond to SMOTE + WE, ROS and SMOTE, both with logistic regression and with SVM classifiers. It is also interesting to note that these algorithms usually

Table 3 AUC values over the data sets with low/moderate imbalance using logR

Data set	*Imbalanced*	*RUS*	*OSS*	*NCL*	*SBC*	*ROS*	*SMOTE*	*SMOTE+WE*	*SL-SMOTE*
Australian									
(1:4)	0.877	0.860	0.831	0.882	0.559	0.882	0.883	0.885	0.877
(1:6)	0.796	0.817	0.820	0.869	0.566	0.863	0.861	0.871	0.871
(1:8)	0.845	0.752	0.771	0.848	0.500	0.860	0.856	0.870	0.855
German									
(1:4)	0.611	0.709	0.717	0.710	0.676	0.735	0.723	0.729	0.716
(1:6)	0.608	0.688	0.667	0.713	0.662	0.705	0.706	0.721	0.700
(1:8)	0.554	0.686	0.705	0.657	0.643	0.719	0.733	0.753	0.718
Japanese									
(1:4)	0.869	0.864	0.810	0.859	0.793	0.871	0.871	0.876	0.878
(1:6)	0.825	0.818	0.827	0.855	0.679	0.874	0.877	0.852	0.864
(1:8)	0.764	0.816	0.804	0.842	0.815	0.826	0.873	0.867	0.860
UCSD									
(1:4)	0.728	0.800	0.793	0.807	0.700	0.810	0.802	0.806	0.794
(1:6)	0.670	0.826	0.807	0.802	0.669	0.812	0.800	0.799	0.785
(1:8)	0.615	0.787	0.754	0.774	0.657	0.814	0.805	0.805	0.788
Average ranking	7.708	6.000	6.333	4.458	8.500	2.667	2.917	2.333	4.083

Table 4 AUC values over the data sets with low/moderate imbalance using SVM

Data set	*Imbalanced*	*RUS*	*OSS*	*NCL*	*SBC*	*ROS*	*SMOTE*	*SMOTE+WE*	*SL-SMOTE*
Australian									
(1:4)	0.891	0.891	0.891	0.891	0.843	0.891	0.891	0.891	0.891
(1:6)	0.852	0.869	0.872	0.872	0.833	0.871	0.872	0.872	0.872
(1:8)	0.863	0.852	0.845	0.846	0.819	0.889	0.854	0.851	0.856
German									
(1:4)	0.510	0.712	0.706	0.716	0.695	0.726	0.724	0.716	0.718
(1:6)	0.500	0.701	0.690	0.701	0.650	0.688	0.707	0.702	0.706
(1:8)	0.500	0.739	0.641	0.624	0.682	0.725	0.729	0.727	0.728
Japanese									
(1:4)	0.888	0.888	0.888	0.888	0.580	0.888	0.888	0.888	0.888
(1:6)	0.866	0.866	0.865	0.866	0.560	0.866	0.866	0.866	0.866
(1:8)	0.862	0.867	0.875	0.875	0.766	0.874	0.870	0.875	0.875
UCSD									
(1:4)	0.666	0.771	0.797	0.786	0.668	0.790	0.780	0.784	0.766
(1:6)	0.500	0.774	0.770	0.759	0.635	0.779	0.770	0.778	0.753
(1:8)	0.500	0.775	0.739	0.730	0.636	0.797	0.772	0.780	0.758
Average ranking	7.083	4.625	5.167	4.875	8.333	3.500	3.708	3.667	4.042

perform better than the original imbalanced data (without resampling) even with a higher imbalance ratio; for example, in Table 3 the AUC using SMOTE over German (1:8) is 0.733, whereas the AUC over the original German (1:4) is 0.611. One can see that in many cases, this effect also happens when comparing over-sampling and under-sampling.

When comparing the results given by logR in Table 3 with those of SVM in Table 4, it seems that the logistic regression model consistently performs better than the SVM approach, independently of the imbalance ratio. This finding is in agreement with the conclusions drawn in some previous studies (Baesens *et al*, 2003; Xiao *et al*, 2006; Kennedy *et al*, 2010).

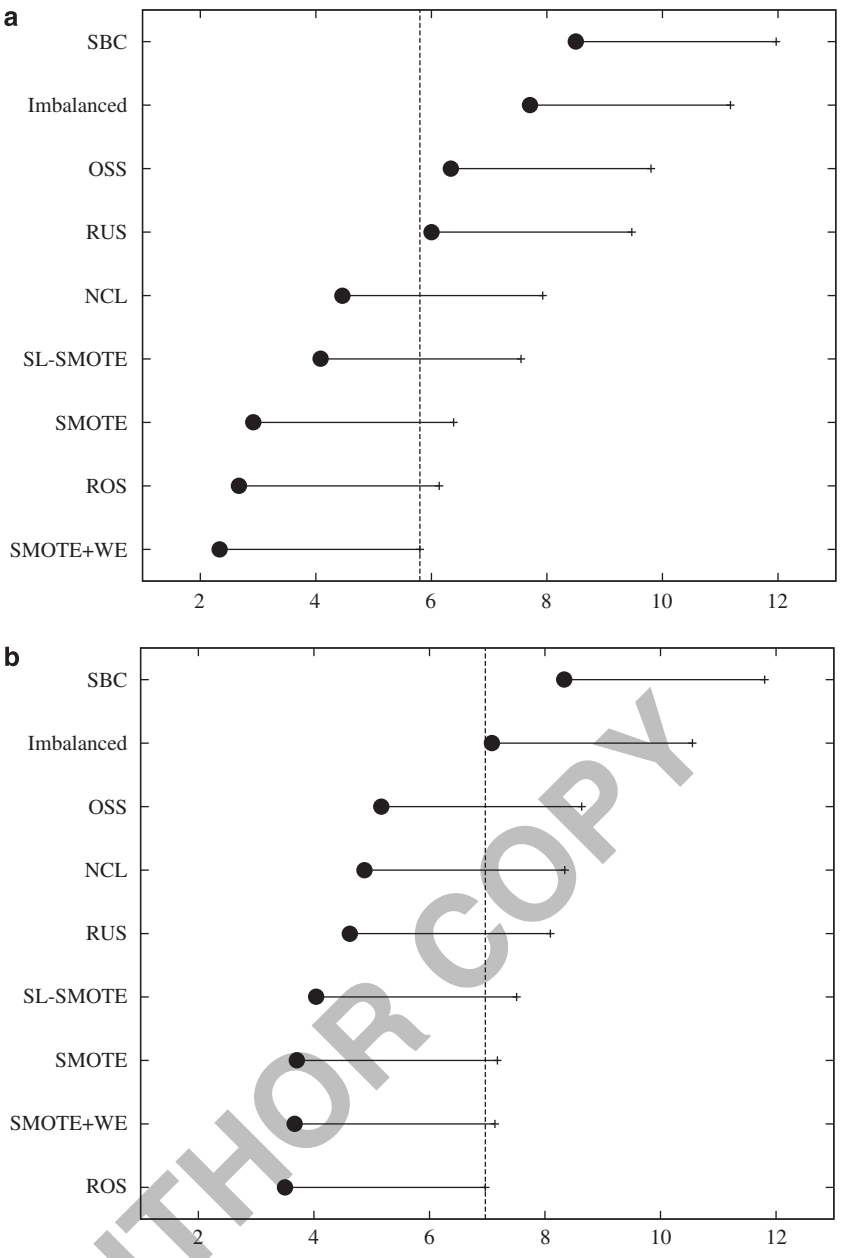

Figure 1 Significance diagrams ($CD = 3.468$) for the data sets with a low/moderate imbalance ratio: (a) Logistic regression; (b) SVM.

A Nemenyi *post hoc* test ($\alpha = 0.05$) has also been applied to report any significant differences between all pairs of algorithms. The results of this test are then depicted by significance diagrams (Lessmann *et al*, 2008), plotting the Friedman average ranks and the critical difference tail. The diagram plots resampling algorithms against average rankings, whereby all methods are sorted according to their ranks. The line segment to the right of each algorithm represents the critical difference (in this case, $CD = 3.468$). The vertical dotted line indicates the end of the best performing method. Therefore, all algorithms right to this line perform significantly worse than the best method.

Figure 1(a) displays the significance diagram for the logR model, where the best resampling technique has been SMOTE + WE with an average rank of 2.333. As can be seen, this method is significantly better than using the original imbalanced data set or any under-sampling algorithm; only the NCL under-sampling approach is not significantly worse than the best performing technique. Note that even the random over-sampling algorithm with an average rank value of 2.667 is significantly better than the imbalanced data set, OSS and SBC.

In the case of the SVM, Figure 1(b) clearly shows that only the results of using the imbalanced data and the

SBC method are significantly worse than those given by the best performing algorithm (random over-sampling with an average rank of 3.500). From this, it seems that the use of a linear kernel SVM produces non-significant differences in performance among most resampling techniques.

Table 5 AUC values over the highly imbalanced data sets using logR

Data set	*Imbalanced*	*RUS*	*OSS*	*NCL*	*SBC*	*ROS*	*SMOTE*	*SMOTE+WE*	*SL-SMOTE*
Australian									
(1:10)	0.660	0.749	0.773	0.864	0.539	0.854	0.866	0.866	0.836
(1:12)	0.762	0.698	0.741	0.869	0.500	0.843	0.861	0.865	0.860
(1:14)	0.675	0.661	0.624	0.789	0.500	0.790	0.850	0.851	0.845
German									
(1:10)	0.528	0.664	0.655	0.630	0.611	0.697	0.708	0.724	0.696
(1:12)	0.511	0.627	0.595	0.576	0.645	0.618	0.683	0.664	0.647
(1:14)	0.535	0.699	0.616	0.570	0.599	0.657	0.674	0.670	0.641
Japanese									
(1:10)	0.663	0.770	0.705	0.853	0.550	0.851	0.841	0.878	0.834
(1:12)	0.681	0.708	0.709	0.756	0.822	0.778	0.832	0.874	0.849
(1:14)	0.610	0.720	0.670	0.718	0.705	0.763	0.792	0.791	0.750
UCSD									
(1:10)	0.605	0.800	0.766	0.768	0.559	0.802	0.799	0.800	0.769
(1:12)	0.583	0.792	0.759	0.762	0.543	0.826	0.816	0.819	0.800
(1:14)	0.600	0.781	0.718	0.756	0.500	0.832	0.821	0.828	0.788
Iranian									
(1:19)	0.505	0.628	0.619	0.594	0.500	0.664	0.701	0.699	0.717
Average ranking	8.077	5.346	6.846	5.462	7.846	3.385	2.423	1.846	3.769

Table 6 AUC values over the highly imbalanced data sets using SVM

Data set	*Imbalanced*	*RUS*	*OSS*	*NCL*	*SBC*	*ROS*	*SMOTE*	*SMOTE+WE*	*SL-SMOTE*
Australian									
(1:10)	0.610	0.867	0.867	0.867	0.830	0.859	0.867	0.864	0.867
(1:12)	0.682	0.883	0.883	0.883	0.711	0.883	0.883	0.883	0.883
(1:14)	0.568	0.873	0.849	0.873	0.736	0.871	0.873	0.871	0.873
German									
(1:10)	0.500	0.696	0.635	0.585	0.640	0.694	0.709	0.717	0.679
(1:12)	0.500	0.652	0.525	0.500	0.569	0.672	0.702	0.685	0.652
(1:14)	0.500	0.696	0.565	0.496	0.618	0.674	0.671	0.700	0.649
Japanese									
(1:10)	0.886	0.886	0.886	0.886	0.653	0.868	0.878	0.876	0.881
(1:12)	0.568	0.825	0.763	0.830	0.573	0.878	0.863	0.861	0.829
(1:14)	0.500	0.816	0.730	0.746	0.500	0.779	0.771	0.762	0.727
UCSD									
(1:10)	0.500	0.777	0.711	0.711	0.500	0.796	0.773	0.776	0.752
(1:12)	0.500	0.770	0.702	0.673	0.686	0.797	0.786	0.795	0.775
(1:14)	0.500	0.793	0.639	0.628	0.500	0.813	0.792	0.794	0.778
Iranian									
(1:19)	0.500	0.673	0.498	0.498	0.500	0.718	0.732	0.712	0.719
Average ranking	8.077	3.346	5.962	5.885	7.539	3.346	3.039	3.423	4.385

5.2. High imbalance ratio

Tables 5 and 6 provide the AUC values for the highly imbalanced data sets when applying the logR and SVM prediction models, respectively. The Friedman average ranking of the strategies ($K = 9$) over the data sets ($N = 13$) has also been included for each classifier. As can been seen, both under-sampling and over-sampling methods outperform the original imbalanced data set independently of the classifier used.

In the case of the logistic regression model, all over-sampling algorithms perform better than the under-sampling techniques. The best performing approach corresponds to SMOTE + WE with an average rank of 1.846, followed by SMOTE with 2.423 and ROS with 3.385. Although the under-sampling methods perform worse than any over-sampling algorithm, it is worth pointing out that they still improve the AUC values achieved when classifying with the original imbalanced data set (this is the strategy with the highest average rank).

Focusing on the results of the SVM classifier in Table 6, one can observe that SMOTE, ROS and SMOTE + WE are the best approaches with average ranks of 3.039, 3.346 and 3.423, respectively. In this case, the random

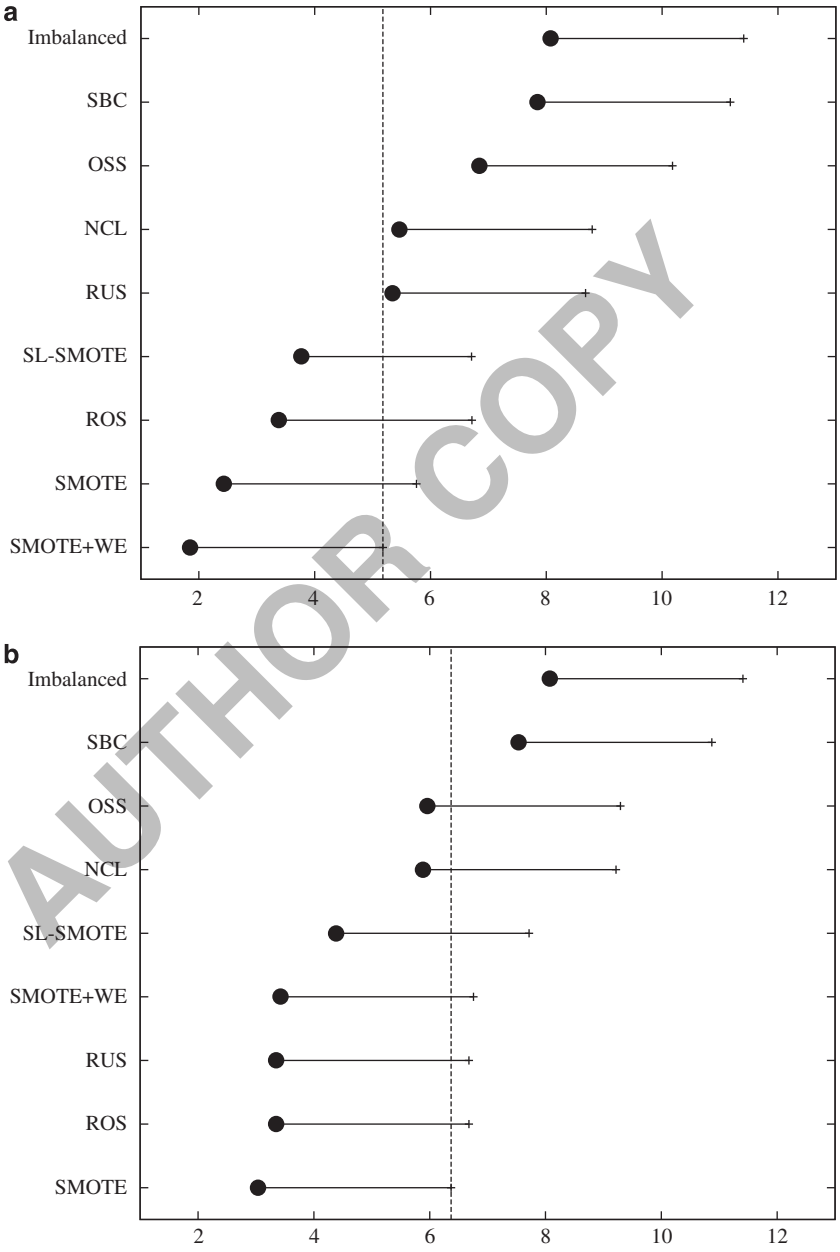

Figure 2 Significance diagrams ($CD = 3.332$) for the highly imbalanced data sets: (a) Logistic regression; (b) SVM.

under-sampling algorithm appears to be as good as those over-sampling strategies, with an average rank of 3.346. Once again, the SBC technique and the use of the imbalanced data set without any preprocessing correspond to the options with the highest average ranks (7.539 and 8.077, respectively).

If we now analyse the results obtained for the highly imbalanced data sets and those of a low/moderate imbalance ratio in Section 5.2, it is possible to notice that the best solution to the class imbalance problem consistently corresponds to over-sampling, independently of employing a statistical model or an artificial intelligence technique.

As in the case of the results for the data sets with a low/moderate ratio, a Nemenyi *post hoc* test ($\alpha = 0.05$) has also been applied to report any significant differences between all pairs of algorithms and then depicted by significance diagrams with a critical difference value of 3.332.

Figure 2(a) shows the significance diagram for the logistic regression model, where the SMOTE + WE technique proves to be significantly better than using any under-sampling algorithm or the original imbalanced data set. The rest of over-sampling algorithms are also significantly better than OSS, SBC and the imbalanced sets.

For the SVM, Figure 2(b) allows to observe that differences among the resampling strategies are less significant than in the case of using logR. Nonetheless, one can see that five methods (SMOTE, ROS, RUS, SMOTE + WE, SL-SMOTE) perform significantly better than SBC and the original imbalanced sets.

6. Conclusions

This paper has studied a number of resampling techniques for statistical and computational intelligence prediction models when addressing the class imbalance problem. The performance of these methods has been assessed by means of the AUC (balanced accuracy) measure, and then the Friedman statistic and the Nemenyi *post hoc* test have been applied to determine whether the differences between the average ranked performances were statistically significant. In order to better illustrate these statistical differences, the significance diagram for each classifier has been analysed.

The experiments carried out over real-world data sets with varying imbalance ratios have demonstrated that resampling can be an appropriate solution to the class imbalance problem in credit scoring. Also, the results have allowed to see that over-sampling outperforms under-sampling in most cases, especially with the logistic regression prediction model where the Nemenyi test has shown more significant differences. Another interesting finding refers to the fact that the resampling approaches have produced similar gains in performance without regard to the imbalance ratio.

In credit scoring applications, a small increase in performance may result in significant future savings and have important commercial implications (Henley and Hand, 1997). Taking this into account, the improvement in performance achieved by the resampling strategies may become of great importance for banks and financial institutions. Therefore, it seems strongly advisable to face down the imbalance problem (probably by means of an over-sampling technique) before building the prediction model.

Acknowledgements—This work has partially been supported by the Spanish Ministry of Education and Science under grant TIN2009–14205 and the Generalitat Valenciana under grant PROMETEO/2010/028.

References

Abdou HA and Pointon J (2011). Credit scoring, statistical techniques and evaluation criteria: A review of the literature. *Intelligent Systems in Accounting, Finance & Management*: 18(2–3): 59–88.

Abrahams CR and Zhang M (2008). *Fair Lending Compliance: Intelligence and Implications for Credit Risk Management*. Wiley: Hoboken, NJ.

Alcalá-Fdez J *et al* (2009). KEEL: A software tool to assess evolutionary algorithms for data mining problems. *Soft Computing* **13**(3): 307–318.

Baesens B, van Gestel T, Viaene S, Stepanova M, Suykens J and Vanthienen J (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society* **54**(6): 627–635.

Baesens B, Mues C, Martens D and Vanthienen J (2009). 50 years of data mining and OR: Upcoming trends and challenges. *Journal of the Operational Research Society* **60**(S1): 816–823.

Barandela R, Sánchez JS, García V and Rangel E (2003). Strategies for learning in class imbalance problems. *Pattern Recognition* **36**(3): 849–851.

Batista GEAPA, Prati RC and Monard MC (2004). A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explorations Newsletter* **6**(1): 20–29.

Bhattacharyya S, Jha S, Tharakunnel K and Westland JC (2011). Data mining for credit card fraud: A comparative study. *Decision Support Systems* **50**(3): 602–613.

Brown I and Mues C (2012). An experimental comparison of classification algorithms for imbalanced credit scoring data sets. *Expert Systems with Applications* **39**(3): 3446–3453.

Bunkhumpornpat C, Sinapiromsaran K and Lursinsap C (2009). Safe-level-SMOTE: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem. In: *Proceedings of the 13th Pacific Asia Conference on Knowledge Discovery and Data Mining*, Bangkok, Thailand, pp 475–482.

Chawla NV, Bowyer KW, Hall LO and Kegelmeyer WP (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research* **16**: 321–357.

Chawla NV, Japkowicz N and Kotcz A (2004). Editorial: Special issue on learning from imbalanced data sets. *SIGKDD Explorations Newsletter* **6**(1): 1–6.

Chawla NV, Cieslak DA, Hall LO and Joshi A (2008). Automatically countering imbalance and its empirical relationship to cost. *Data Mining and Knowledge Discovery* **17**(2): 225–252.

Demšar J (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research* **7**(1): 1–30.

Florez-Lopez R (2010). Credit risk management for low default portfolios. Forecasting defaults through cooperative models and

boostrapping strategies. In: *Proceedings of the 4th European Risk Conference—Perspectives in Risk Management: Accounting, Governance and Internal Control*, Nottingham, UK, pp 1–27.

García S, Fernández A, Luengo J and Herrera F (2010). Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Information Sciences* **180**(10): 2044–2064.

Hand DJ (2005). Good practice in retail credit scorecard assessment. *Journal of the Operational Research Society* **56**(9): 1109–1117.

Hand DJ and Vinciotti V (2003). Choosing k for two-class nearest neighbour classifiers with unbalanced classes. *Pattern Recognition Letters* **24**(9–10): 1555–1562.

Hart PE (1968). The condensed nearest neighbor rule. *IEEE Transactions on Information Theory* **14**(3): 505–516.

He H and Garcia EA (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering* **21**(9): 1263–1284.

Henley WE and Hand DJ (1997). Construction of a k-nearest-neighbour credit-scoring system. *IMA Journal of Management Mathematics* **8**(4): 305–321.

Hochberg Y and Tamhane AC (1987). *Multiple Comparison Procedures*. John Wiley & Sons: New York, NY.

Huang Y-M, Hung C-M and Jiau HC (2006). Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem. *Nonlinear Analysis: Real World Applications* **7**(4): 720–747.

Huang Z, Chen H, Hsu C-J, Chen W-H and Wu S (2004). Credit rating analysis with support vector machines and neural networks: A market comparative study. *Decision Support Systems* **37**(4): 543–558.

Japkowicz N and Stephen S (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis* **6**(5): 429–449.

Kennedy K, Mac Namee B and Delany SJ (2010). Learning without default: A study of one-class classification and the low-default portfolio problem. In: *Proceedings of the 20th Irish Conference on Artificial Intelligence and Cognitive Science*, Dublin, Ireland, pp 174–187.

Kubat M and Matwin S (1997). Addressing the curse of imbalanced training sets: One-sided selection. In: *Proceedings of the 14th International Conference on Machine Learning*, Nashville, TN, pp 179–186.

Laurikkala J (2001). Improving identification of difficult small classes by balancing class distribution. In: *Proceedings of the 8th Conference on Artificial Intelligence in Medicine in Europe*, Cascais, Portugal, pp 63–66.

Lessmann S, Baesens B, Mues C and Pietsch S (2008). Benchmarking classification models for software defect prediction: A proposed framework and novel findings. *IEEE Transactions on Software Engineering* **34**(4): 485–496.

Maciejewski T and Stefanowski J (2011). Local neighbourhood extension of SMOTE for mining imbalanced data. In: *Proceedings of the IEEE Symposium on Computational Intelligence and Data Mining*, Paris, France, pp 104–111.

Pluto K and Tasche D (2006). Estimating probabilities of default for low default portfolios. In: Engelmann B and Rauhmeier R (eds). *The Basel II Risk Parameters: Estimation, Validation, and Stress Testing*. Springer: Berlin, pp 75–101.

Provost F and Fawcett T (1997). Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In: *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, Newport Beach, CA, pp 43–48.

Sabzevari H, Soleymani M and Noorbakhsh E (2007). A comparison between statistical and data mining methods for credit scoring in case of limited available data. In: *Proceedings of the 3rd CRC Credit Scoring Conference*, Edinburgh, UK.

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

Thomas LC, Edelman DB and Crook JN (2002). *Credit Scoring and Its Applications*. SIAM: Philadelphia, PA.

Tian B, Nan L, Zheng Q and Yang L (2010). Customer credit scoring method based on the SVDD classification model with imbalanced dataset. In: *Proccedings of the International Conference on E-business Technology and Strategy*, Ottawa, Canada, pp 46–60.

Tomek I (1976). Two modifications of CNN. *IEEE Transactions on Systems, Man and Cybernetics* **6**(11): 769–772.

Vinciotti V and Hand DJ (2003). Scorecard construction with unbalanced class sizes. *Journal of the Iranian Statistical Society* **2**(2): 189–205.

Wang G, Hao J, Ma J and Jiang H (2011). A comparative assessment of ensemble learning for credit scoring. *Expert Systems with Applications* **38**(1): 223–230.

Wilson DL (1972). Asymptotic properties of nearest neighbour rules using edited data. *IEEE Transactions on Systems, Man and Cybernetics* **2**(3): 408–421.

Xiao W, Zhao Q and Fei Q (2006). A comparative study of data mining methods in consumer loans credit scoring management. *Journal of Systems Science and Systems Engineering* **15**(4): 419–435.

Xie H, Han S, Shu X, Yang X, Qu X and Zheng S (2009). Solving credit scoring problem with ensemble learning: A case study. In: *Proceedings of the 2nd International Symposium on Knowledge Acquisition and Modeling*, Vol. 1, Wuhan, China, pp 51–54.

Yang Z, Wang Y, Bai Y and Zhang X (2004). Measuring scorecard performance. In: *Proceedings of 4th International Conference on Computational Science*, Krakow, Poland, pp 900–906.

Yao P (2009). Comparative study on class imbalance learning for credit scoring. In: *Proceedings of the 9th International Conference on Hybrid Intelligent Systems*, vol. 2, Shenyang, China, pp 105–107.

Yen S-J and Lee Y-S (2006). Under-sampling approaches for improving prediction of the minority class in an imbalanced dataset. In: Thoma M and Morari M (eds). *Intelligent Control and Automation, Lecture Notes in Control and Information Sciences*. Vol. **344** Springer: Berlin, pp 731–740.

Zar JH (2009). *Biostatistical Analysis*. Pearson: Upper Saddle River, NJ.

Received November 2011;
accepted August 2012 after one revision