Coupling REPMAC with FDA to solve highly imbalanced classification problems

Hernán Ahumada, Guillermo L. Grinblat, Lucas C. Uzal, Alejandro Ceccatto and Pablo M. Granitto

CIFASIS (CONICET-UNR-UPCAM)
Bv. 27 de Febrero 210 bis, 2000 Rosario, Argentina
{ahumada,grinblat,uzal,ceccatto,granitto}@cifasis-conicet.gov.ar

Abstract

In many critical real world classification problems one of the classes has much less samples than the others (class imbalance). In a previous work we introduced the REPMAC algorithm to solve imbalanced problems. Using a clustering method, REPMAC recursively splits the majority class in several subsets, creating a decision tree, until the resulting sub-problems are balanced or easy to solve. In this work we evaluate the use of three different classifiers coupled with REPMAC. We compare the performance of those methods using 7 datasets from the UCI repository spanning a wide range of number of features and imbalance degree. We find that the good performance of REPMAC is almost independent of the classifier coupled to it, which suggest that it success is mostly related to the use of an appropriate strategy to cope with imbalanced problems.

1 Introduction

In many real world classification problems one of the classes is represented by a much lower number of instances than the other classes, usually know as "class imbalance" problem. This setting is of great importance since it usually corresponds to critical applications, such as fraud and fault detection or medical diagnosis [13, 8]. Typical machine learning methods are basically designed to learn fairly balanced datasets. When dealing with the class imbalance problem, those approaches focus mostly on the majority classes, predicting poorly the minority class examples [20]. This problem has been the subject of an increasing number of publications over the last years. Also, several international workshops were dedicated specifically to the imbalance data problem [1, 12, 16].

In a previous work [2] we introduced the REPMAC (REcursive Partitioning of the MAjority Class) method, a new method to solve the imbalance problem, that combines unsupervised and supervised learning. We adopted a "divide and conquer strategy", as is usually done to solve multiclass problems using a combination of binary classifiers. Using a clustering algorithm, the method recursively splits the data of the majority class in two, until the resulting datasets are balanced or can be easily discriminated. The result of the process is a decision tree, which drives each example to an appropriate balanced classifier at a given leaf. We showed in that work that REPMAC has potential advantages over other strategies, constructing small trees that are very efficient classifiers.

In that previous series of experiments we chose to use linear SVMs [10] as classifiers. That election introduced potential limitations to the performance of REPMAC. For example, we lost the potential benefits of using non–linear classifiers associated with REPMAC. Also, SVM uses an approximation to estimate class posterior probabilities, which we need to construct ROC curves. In this work we evaluate the performance of REPMAC coupled with Flexible Discriminant Analysis (FDA) classifiers [14]. As SVMs, FDA can produce diverse discriminant functions, from high margin linear classifiers to very flexible discriminants, also giving accurate estimations of posterior class probabilities in all cases.

The rest of the paper is organized as follows. In the next section we review previous works on imbalanced datasets. In section 3 we review the REPMAC method in detail and FDA classifiers. Then, in section 4 we evaluate the combination of REPMAC with diverse FDA classifiers on several datasets. Finally, in section 5 we discuss the results and future lines of research.

2 Related Work

Several works introduced solutions to the class imbalance problem associated to particular learning methods. Chawla et al. [9] and Drummond et al. [11] described specific methods for decision tree learning. Zhang et al. [24] addressed the problem using k-nn classifiers, showing also an interesting application involving information extraction from the biomedical literature. A number of papers introduced variants of SVMs [10] appropriate for imbalanced problems, following different strategies [17, 6, 3, 21].

The most used strategy to cope with imbalanced datasets, however, is to equalize the composition of the dataset, either by subsampling the numerous classes or by oversampling the minority one. A clear advantage of those methods is that they can be used with any classifier. The two most simple schemes are random minority oversampling (ROS), where instances of the minority class are randomly duplicated, and random majority undersampling (RUS), where instances of the majority classes are randomly discarded from the dataset. More elaborated (sub or over) sampling methods were also developed. For example, Chawla et al. [8] introduced an intelligent oversampling method called Synthetic Minority Oversampling Technique (SMOTE). SMOTE adds new, artificial minority examples by extrapolating between preexisting minority instances rather than simply duplicating original examples. Batista et al. [5] evaluated 10 different (sub and over) sampling strategies finding that the simple ROS oversampling is usually the best choice. They also introduced the use of the area under the ROC curve (AUC from now) as a more powerful metric to compare strategies in these settings. In a recent work, Van Hulse et al. [19] produced a deep experimental comparison including 35 datasets, several over and under sampling strategies and different classifiers. Again, they found that the simple RUS and ROS strategies are difficult to improve on and that the results of different strategies are classifier-dependent.

Recently, Yen and Lee [22] introduced a new intelligent under-sampling technique based on clustering the majority class and then under-sampling it in a cluster–balanced way. A similar strategy was proposed by Yu et al. [23], who under-sampled the majority class using the prototypes of a Kohonen network, but they limited the application to SVM classifiers. In a recent technical report [7], Y.C. Chang proposed to split the majority class using k-means clustering, in order to obtain a given number of clusters, each one of them with a number of samples similar to the minority class. A linear SVM was then fitted to each balanced problem, and all classifiers were then joined using a logistic regression. Even if the method has potential advantages over ROS (it uses all the information in the dataset, for example), its results are

The REPMAC Method

Inputs:
D^+ : The majority class dataset
D^- : The minority class dataset
$Cl()$: A clustering method
$DF()$: A decision function (i.e. a classifier)

Function REPMAC(D^+, D^-, Cl, DF):
1. Apply Cl to D^+ to create D_1^+ and D_2^+
2. For $i = 1$ to 2:
   IF Stopping–Criteria(D_i^+, D^-) is met THEN
      Build a classifier $DF(D_i^+, D^-)$
   ELSE
      CALL $REPMAC(D_i^+, D^-, Cl, DF)$:

Figure 1: The REPMAC algorithm.

usually no better than the simpler ROS.

3 The REPMAC Method

REPMAC is based on a simple idea: To divide the majority class intelligently into several clusters in order to transform the imbalanced problem into a set of balanced problems. In Figure 1 we show a scheme of REPMAC. The method follows the typical classification tree strategy: First, it splits the majority class in two clusters using any appropriate method (k-means clustering [18] in this case). Then it analyzes each of the clusters to check if they meet any of a set of stopping criteria. If they do not, it goes back to the first step (creating thus a recursive process) and applies again the clustering method. When one of the clusters meets the criteria, the method fits a classifier to the resulting dataset (i.e. the cluster plus the minority class). The result of the process is a decision tree, with a clustering solution at each node and a classifier at each leaf.

Once we have the full decision tree, a new example can be classified according to the following procedure: At each level of the tree, starting from the root, the example is assigned to one of the branches, according to the rules of the clustering method (for example, for k-means, looking for the nearest centroid). The procedure is iterated until a leaf is reached, where the example is classified using the discriminant function associated to that leaf.

The selection of an appropriate stopping criteria is one of the keys of the method. The simplest control is to check the degree of imbalance of the resulting sub–problems after the clustering procedure. If we define the Imbalance Level (IL) of a potential leaf as the ratio between the number of samples of the majority and the minority class, we can stop the recursive splitting if $IL \leq S_{IL}$, for a given value of S_{IL}. Of course, higher values of S_{IL} will produce smaller trees, but also the resulting sub-problems will be more imbalanced.

In our previous work [2] we also introduced a more elaborated stopping criteria based on the Performance Level (PL) of each sub-problem. To estimate PL we used an internal cross-validation (CV) procedure. We showed that this criteria produces smaller trees but with some decrease in accuracy. We leave the evaluation of this criteria coupled with FDA methods to a future work.

Table 1: Details of the 7 datasets used in this work. The "p" column shows the number of inputs, "n" the total number of samples and "ratio" the fraction of samples in the minority class.

Dataset	p	n	ratio (%)
nursery (3)	8	13460	2.53
letter (A)	16	20000	3.95
car (3)	6	1728	3.99
glass (3)	9	214	7.94
pendigits (5)	16	10792	9.60
satimage (4)	36	6435	9.73
optdigits (8)	64	5620	9.86

3.1 Classifiers

In our first study of REPMAC we used a linear SVM [10] to build the different classifiers. In this work we will also use Flexible Discriminant Analysis for that task. FDA was introduced by Hastie et al. [14] as an improved version of classical Fisher's LDA [15]. LDA is a standard tool for classification and dimension reduction. Roughly, it seeks a linear combination of the features, which maximizes the ratio of its between-class variance to its within-class variance. After that, classes are typically assigned according to Mahalanobis distances to class centroids in this transformed space. FDA is a regularized version of LDA, more appropriate for noisy, high–dimensional situations. The method is based on recasting the LDA problem as a regularized linear regression one, and then to apply any of the many well-known techniques available for this task. We selected here two versions of FDA. First, we use standard Ridge Regression (GenRidge) [15], which has only one free parameter, the ridge constant λ that penalizes high values of the fitted variables. λ plays a similar role to the C parameter in SVMs, regulating the margin of the solutions. We also use regression splines (MARS) [15] as regression method for FDA. Mars allows FDA to produce efficient non–linear classifiers (equivalent to using kernels with SVMs). In both cases, GenRidge and Mars, the method produces accurate estimations of the class posterior probabilities.

4 Experiments

4.1 Datasets

We used 7 different datasets to perform our experiments, all obtained from the UCI repository [4]. In Table 1 we show their main characteristics. All 7 are multiclass problems, which we converted to highly imbalanced binary problems by selecting a given class as the minority one and combining all other classes to form the majority class. In Table 1 we show between parenthesis the class selected as minority. To have a fair comparison, the selected datasets span a large variety of situations, from small to very large datasets, from imbalance ratios of 1/40 to 1/10 or from 6 to 64 variables.

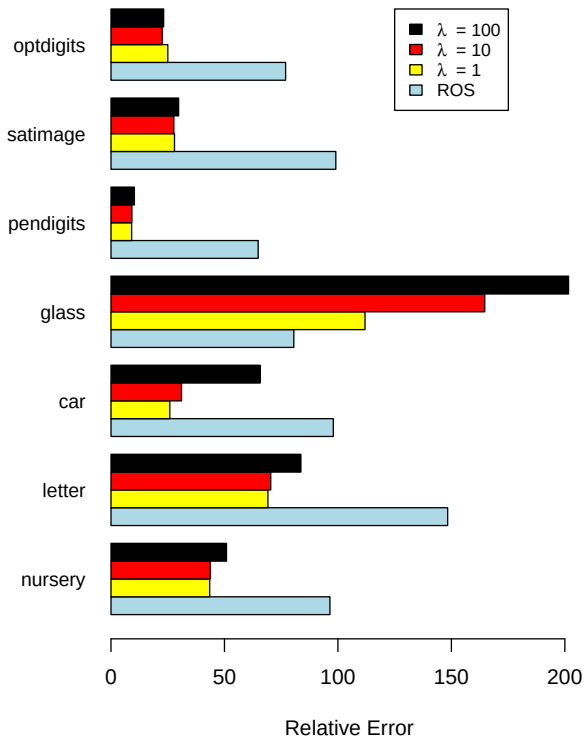

Figure 2: Error levels (1-AUC) for REPMAC with FDA–GenRidge using different values of λ. Units are relative to the Plain method (which is set to 100). We include ROS results as reference.

4.2 Experimental setup

In this work we evaluate the performance of REPMAC coupled with different classifiers: linear SVM, FDA–GenRidge and FDA–Mars. We selected for comparison 3 different methods for imbalanced problems: as base (plain) method, we use directly each of the classifiers with the original datasets. We also selected standard ROS (with the right oversampling rate as to balance the dataset) and the REPMAC method. In all cases (when it is not explained otherwise) we optimized free parameters, like C or λ, using internal cross-validation over the training set.

In all experiments we used a 5–fold Cross–Validation setting, which was repeated 4 times as to average 20 runs of each method. In all cases the same splits in train and test sets were used for all methods and classifiers.

As performance measure we used the AUC, as suggested by Batista et al.[5]. The AUC is appropriate for imbalanced problems, evaluating at the same time the behavior of the methods at different operating points. When we use figures to show the results, we preferred to show an error measure, (1 - AUC), for each method. The difference between error levels are easier to visualize. In all figures we use units relative to the performance of a given base method (i.e., we plotted $100 \times Error(method)/Error(base)$).

Table 2: Area under the ROC curve (AUC) using linear SVM classifiers. The result correspond to the 3 methods under comparison, evaluated over the 7 datasets used in this work.

Dataset	Plain	ROS	REPMAC
nursery	0.9862	0.9863	**0.9955**
letter	0.9809	0.9876	**0.9942**
car	0.8923	0.9614	**0.9920**
glass	0.6899	**0.8748**	0.7748
pendigits	0.9723	0.9769	**0.9982**
satimage	0.7137	0.7587	**0.9338**
optdigits	0.9827	0.9821	**0.9951**

4.3 Comparative results

As a first experiment, we evaluated the influence of the regularization of the classifiers over the performance of REPMAC. We used FDA–GenRidge in this case, with three very different values of λ. Figure 2 show the corresponding results. Only in 2 datasets (glass and car) the results are highly dependent on λ. It is also interesting that in 6 out of the 7 datasets the best result corresponds to the lowest regularization ($\lambda = 1$). This is probably related to a reduction in the risk of overfitting when discriminating the small sub–problems produced by REPMAC.

We then compared all classifiers and methods. For REPMAC we used in this case a simple stopping rule: $IL \leq 1.15$. In Tables 2, 3 and 4 we show the AUC results for SVM, FDA–GenRidge and FDA–Mars, respectively. In each Table we highlighted the best result for each dataset. The SVM results were taken from our previous work [2]. There are only minor differences among the three tables. In all cases ROS produces only a small improvement over the Plain method, except for the Glass dataset, where it shows the best performance. In 6 out of the 7 evaluations REPMAC showed the best performance for linear SVMs (Table 2). A similar behaviour can be observed for the two new classifiers, which indicates that the good results of REPMAC are almost independent of the classifier being used.

The satimage dataset is an interesting example where there is a big difference between FDA–Mars and the two linear classifiers. This indicates that the problem has a non–linear decision boundary between the classes. For REPMAC, however, the difference is considerably reduced, probably because the method has divided the non–linear boundary into several almost linear pieces.

In Figure 3 we show a more direct comparison among the three classifiers for the REPMAC method. In two datasets SVM works best, in other two FDA–Mars is better, and both classifiers have the same error in other two datasets. In only one dataset FDA–GenRidge performs best, but in that case REPMAC is worse than simple ROS. Overall, there is an equivalence between linear SVM and FDA–Mars.

Table 3: Area under the ROC curve (AUC) using FDA-GenRidge classifiers. The result correspond to the 3 methods under comparison, evaluated over the 7 datasets used in this work.

Dataset	Plain	ROS	REPMAC
nursery	0.9824	0.9830	**0.9923**
letter	0.9880	0.9821	**0.9917**
car	0.9614	0.9622	**0.9900**
glass	0.8231	**0.8575**	0.8020
pendigits	0.9617	0.9751	**0.9965**
satimage	0.7519	0.7542	**0.9306**
optdigits	0.9770	0.9823	**0.9943**

Table 4: Area under the ROC curve (AUC) using FDA–MARS classifiers. The result correspond to the 3 methods under comparison, evaluated over the 7 datasets used in this work.

Dataset	Plain	ROS	REPMAC
nursery	0.9840	0.9797	**0.9912**
letter	0.9766	0.9886	**0.9942**
car	0.9635	0.7850	**0.9887**
glass	0.7126	**0.8470**	0.7402
pendigits	0.9879	0.9533	**0.9985**
satimage	0.9221	0.9396	**0.9396**
optdigits	0.9850	0.9745	**0.9951**

4.4 Stopping criteria

As we showed in our previous work, the complexity of REPMAC (the depth of the tree and the number of leaves produced) can be regulated by changing the stopping criteria. The simpler control is to check the degree of imbalance IL of the sub–problem under evaluation. For Tables 2 to 4 we have used $S_{IL} = 1.15$, i.e., we stopped the splitting when the ratio between majority and minority class is less than 1.15. To evaluate the influence of S_{IL} in the performance of REPMAC, we repeated the experiments described before using also $S_{IL} = 1.50$ and $S_{IL} = 2.00$ for FDA–GenRidge. Figure 4, top panel, shows the corresponding error levels in units relative to the performance of the ROS method with the same classifier. Figure 4, mid panel, shows the average number of nodes (over the 20 runs of the method) created by REPMAC with different S_{IL} values, and the bottom panel of Figure 4 shows the average depth of the created trees. As we had observed in our previous work for SVM and a subset of these datasets, in most cases there is a positive correlation between error levels and S_{IL} values. All the error levels show

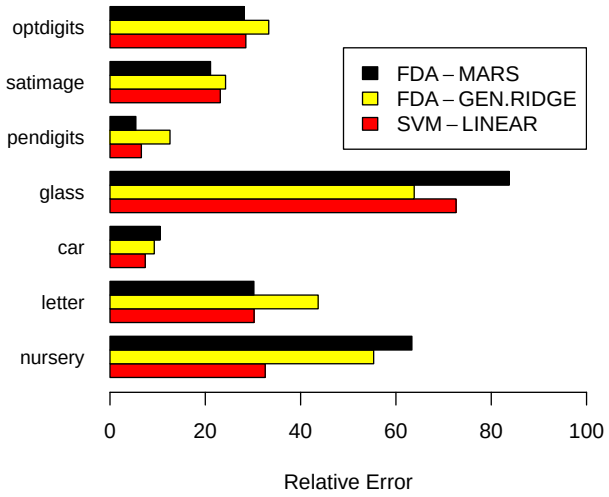

Figure 3: Direct comparison of REPMAC results for three different classifiers. The figure shows error levels (1-AUC) in units relative to the Plain–SVM method (which is set to 100).

an increase, but in all cases (with the exception of the anomalous Glass dataset) they remain clearly better than the ROS method (all values are clearly lower than 100). The reduction in complexity produced by the use of higher S_{IL} values is remarkable, in particular in the number of nodes.

5 Conclusions

In this work we evaluated the performance of REPMAC coupled with different classifiers: linear SVM, FDA–GenRidge and FDA–Mars. When dealing with imbalanced problems, REPMAC recursively split the majority class in several subsets, using a clustering method, producing separate balanced sub-problems that can be more easily discriminated. To evaluate the three classifiers we used 7 datasets (with different characteristics) from the UCI repository.

In several experiments we evaluated diverse aspects of the method, as the dependence with the regularization of each classifier, the non–linear quality of them, or the use of different stopping criteria.

The more important finding of this work is that all those experiments showed that the performance of REPMAC is similar in all cases, being only slightly lower when using FDA–GenRidge classifiers. This suggests that REPMAC's success is independent of the classifier selection, being more related to the use of an efficient strategy to solve imbalanced problems.

As future work we plan to evaluate the use of other clustering methods, appropriate for interesting domains like genomics, proteomics or text mining, and the application of post–pruning techniques to the trees developed by REPMAC. Also, we are evaluating the possibility of using REPMAC in multiclass imbalanced problems.

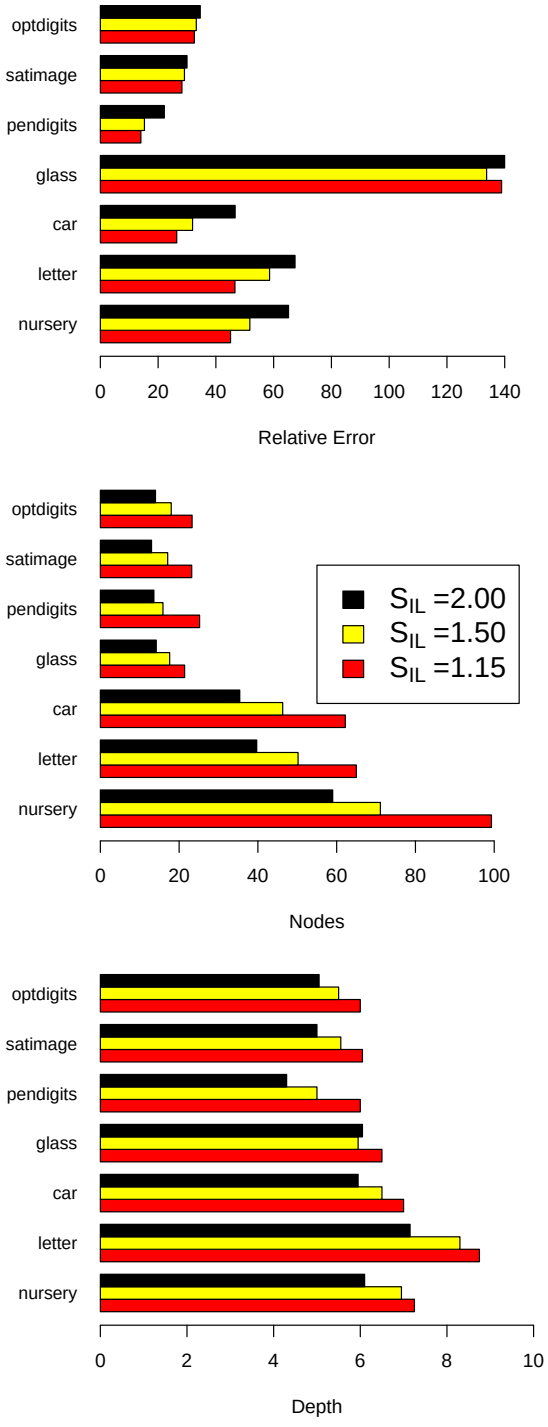

Figure 4: Evaluation of the performance of REPMAC (FDA–GenRidge) for three different S_{IL} values. From top to bottom, relative error (taking the error level of ROS -not plotted- as equal to 100), average number of nodes in the trees and average depth of the trees.

6 Acknowledgments

This work was partially supported by ANPCYT through grants PICT 2226 and 643.

References

[1] AAAI00. *Workshop on Learning from Imbalanced Data Sets.* 2000.

[2] H. Ahumada, G. L. Grinblat, L. C. Uzal, P. M. Granitto, and A. Ceccatto. Repmac: A new hybrid approach to highly imbalanced classification problems. In *HIS'08: Proceedings of the 8th Internation Conference on Hybrid Intelligent Systems*, 2008.

[3] Rehan Akbani, Stephen Kwek, and Nathalie Japkowicz. Applying support vector machines to imbalanced datasets. In *ECML'04: Proceedings of the 15th European Conference in Machine Learning*, pages 39–50, 2004.

[4] A. Asuncion and D.J. Newman. UCI machine learning repository, 2007.

[5] G. E. Batista, R. C. Prati, and M. C. Monard. A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explorations Newsletter*, 6(1):20–29, 2004.

[6] U. Brefeld, P. Geibel, and F. Wysotzki. Support vector machines with example dependent costs. In *ECML'03: Proceedings of the 14th European Conference in Machine Learning*, pages 23–34, 2003.

[7] Y. Chang. *Boosting SVM classifiers with logistic regression.* Technical Report 2003-03, Institute of Statistical Science, Academia Sinica, Taiwan, 2003.

[8] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer. Smote: Synthetic minority oversampling technique. *Journal of Artificial Intelligence Research*, 16:351–357, 2002.

[9] Nitesh Chawla. C4.5 and imbalanced datasets: Investigating the effect of sampling method, probabilistic estimate, and decision tree structure. In *ICML'03: Workshop on Learning from Imbalanced Datasets II*, 2003.

[10] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines.* Cambridge University Press, Cambridge, 2000.

[11] C. Drummond and R. C. Holte. C4.5, class imbalance, and cost sensitivity: Why undersampling beats over-sampling. In *ICML'03: Workshop on Learning from Imbalanced Datasets II*, 2003.

[12] ACM SIGKDD Exploration. *Special Issue on Learning from Imbalanced Data Sets.* 2004.

[13] R. E. Fawcett and F. Provost. Adaptive fraud detection. *Data Mining and Knowledge Discovery*, 3(1):291–316, 1997.

[14] T. Hastie, A. Buja, and R. Tibshirani. Penalized discriminant analysis. *Annals of Statistics*, 23:73–102, 1995.

[15] T. Hastie, R. Tibshirani, and J.H. Friedman. *The elements of statistical learning*. Springer-Verlag, New York, 2001.

[16] ICML03. *Workshop on Learning from Imbalanced Data Sets II*. 2003.

[17] J.-F. Liu and D.-R. Yu. A weighted rough set method to address the class imbalance problem. In *Proceedings of the Sixth International Conference on Machine Learning and Cybernetics*, pages 3693–3698, 2007.

[18] J. McQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematics, Statistics and Probability*, pages 281–297, 1967.

[19] Jason Van Hulse, Taghi M. Khoshgoftaar, and Amri Napolitano. Experimental perspectives on learning from imbalanced data. In *ICML'07: Proceedings of the 24th international conference on Machine learning*, pages 935–942, 2007.

[20] G. M. Weiss and F. Provost. *The Effect of Class Distribution on Classifier Learning: An Empirical Study*. Technical Report ML-TR-44 Rutgers University, Department of Computer Science, 2001.

[21] G. Wu and E. Y. Chang. Class-boundary alignment for imbalanced dataset learning. In *ICML'03: Workshop on Learning from Imbalanced Datasets II*, 2003.

[22] Show-Jane Yen and Yue-Shi Lee. Cluster-based sampling approaches to imbalanced data distributions. In *DaWaK'06: Proceeding of the 8th International Conference in Data Warehousing and Knowledge Discovery*, pages 427–436, 2006.

[23] Ting Yu, John K. Debenham, Tony Jan, and Simeon J. Simoff. Combine vector quantization and support vector machine for imbalanced datasets. In *TFTP International Federation for Information Processing, Volume 217, Artificial Intelligence in Theory and Practice*, pages 81–88, 2006.

[24] J. Zhang and I. Mani. knn approach to unbalanced data distributions: A case study involving information extraction. In *ICML'03: Workshop on Learning from Imbalanced Datasets II*, 2003.