Class imbalances *versus* class overlapping: an analysis of a learning system behavior

Ronaldo C. Prati[1], Gustavo E. A. P. A. Batista[1], and Maria C. Monard[1]

Laboratory of Computational Intelligence - LABIC
Department of Computer Science and Statistics - SCE
Institute of Mathematics and Computer Science - ICMC
University of São Paulo - Campus of São Carlos
P. O. Box 668, 13560-970, São Carlos, SP, Brazil
Phone: +55-16-273-9692. FAX: +55-16-273-9751.
{prati,gbatista,mcmonard}@icmc.usp.br

Abstract. Several works point out class imbalance as an obstacle on applying machine learning algorithms to real world domains. However, in some cases, learning algorithms perform well on several imbalanced domains. Thus, it does not seem fair to directly correlate class imbalance to the loss of performance of learning algorithms. In this work, we develop a systematic study aiming to question whether class imbalances are truly to blame for the loss of performance of learning systems or whether the class imbalances are not a problem by themselves. Our experiments suggest that the problem is not directly caused by class imbalances, but is also related to the degree of overlapping among the classes.

1 Introduction

Machine learning methods have advanced to the point where they might be applied to real world problems, such as in data mining and knowledge discovery. By being applied on such problems, several new issues that have not been previously considered by machine learning researchers are now coming into light. One of these issues is the class imbalance problem, *i.e.*, the differences in class prior probabilities. In real world machine learning applications, it has often been reported that the class imbalance hinder the performance of some standard classifiers. However, the relationship between class imbalance and learning algorithms is not clear yet, and a good understanding of how each one affects the other is lacking. In spite of a decrease in performance of standard classifiers on many imbalanced domains, this does not mean that the imbalance is the sole responsible for the decrease in performance. Rather, it is quite possible that beyond class imbalances yield certain conditions that hamper classifiers induction.

Our research is motivated by experiments we had performed over some imbalanced datasets, for instance the sick dataset [9], that provided good results (99.65% AUC) even with a high degree of imbalance (only 6.50% of the examples belong to the minority class). In addition, other research works seems to agree with our standpoint [8].

In this work, we develop a systematic study aiming to question whether class imbalances hindrance classifier induction or whether these deficiencies might be explained in other ways. To this end, we develop our study on a series of artificial datasets. The idea behind using artificial datasets is to be able to fully control all the variables we want to analyze. If we were not able to control such variables, the results may be masked or difficult to understand and interpret, under the risk of producing misleading conclusions. Our experiments suggest that the problem is not solely caused by class imbalances, but is also related to the degree of data overlapping among the classes.

This work is organized as follow: Section 2 introduces our hypothesis regarding class imbalances and class overlapping. Section 3 presents some notes related to evaluating classifiers performance in imbalanced domains. Section 4 discusses our results. Finally, Section 5 presents some concluding remarks.

2 The Role of Class Imbalance on Learning

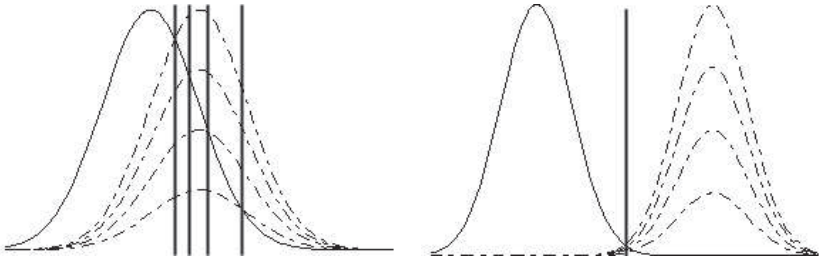
In the last years, several works have been published in the machine learning literature aiming to overcome the class imbalance problem [7, 12]. There were even two international workshops, the former was sponsored by AAAI [5] and the latter was held together with the Twentieth International Conference on Machine Learning [1]. There seems to exist an agreement in the Machine Learning community with the statement that the imbalance between classes is the major obstacle on inducing classifiers in imbalanced domains.

Conversely, we believe that class imbalances are not always the problem. In order to illustrate our conjecture, consider the decision problem shown in Figure 1. The problem is related to building a Bayes classifier for a simple single attribute problem that should be classified into two classes, positive and negative. It is assumed perfect knowledge regarding conditional probabilities and priors. The conditional probabilities for the two classes are given by Gaussian functions, with the same standard deviation for each class, but the negative class having mean one standard deviation (Figure 1(a)) and four standard deviations (Figure 1(b)) apart from the positive class mean. The vertical lines represent optimal Bayes splits.

From Figure 1, it is clear that the influence of changing priors on the positive class, as indicated by the dashed lines, is stronger in Figure 1(a) than in Figure 1(b). This indicates that it is not the class probabilities the main responsible for the hinder in the classification performance, but instead the degree of overlapping between the classes. Thus, dealing with class imbalances will not always help classifiers performance improvement.

3 On Evaluating Classifiers in Imbalanced Domains

The most straightforward way to evaluate classifiers performance is based on the confusion matrix analysis. Table 1 illustrates a confusion matrix for a two class problem having class values **positive** and **negative**.



(a) High overlaid instances (b) Low overlaid instances

Fig. 1. A Simple Decision Problem

	Positive Prediction	*Negative Prediction*
Positive Class	True Positive (TP)	False Negative (FN)
Negative Class	False Positive (FP)	True Negative (TN)

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

From such matrix it is possible to extract a number of widely used metrics for measuring learning systems performance, such as Classification Error Rate, defined as $Err = \frac{FP+FN}{TP+FN+FP+TN}$, or, equivalently, Accuracy, defined as $Acc = \frac{TP+TN}{TP+FP+FN+TN} = 1 - Err$.

However, when the prior classes probabilities are highly different, the use of such measures might produce misleading conclusions. Error rate and accuracy are particularly suspect as performance measures when studying the effect of class distribution on learning since they are strongly biased to favor the majority class. For instance, it is straightforward to create a classifier having an accuracy of 99% (or an error rate of 1%) in a domain where the majority class proportion correspond to 99% of the instances, by simply forecasting every new example as belonging to the majority class.

Other fact against the use of accuracy (or error rate) is that these metrics consider different classification errors as equally important. However, highly imbalanced problems generally have highly non-uniform error costs that favor the minority class, which is often the class of primary interest. For instance, a sick patience diagnosed as healthy might be a fatal error while a healthy patience diagnosed as sick is considered a much less serious error since this mistake can be corrected in future exams.

Finally, another point that should be considered when studying the effect of class distribution on learning systems is that the class distribution may change. Consider the confusion matrix shown in Table 1. Note that the class distribution (the proportion of positive to negative instances) is the relationship between the first and the second lines. Any performance metric that uses values from both columns will be inherently sensitive to class skews. Metrics such as accuracy and error rate use values from both lines of the confusion matrix. As class distribution

changes these measures will change as well, even if the fundamental classifier performance does not.

All things considered, it would be more interesting if we use a performance metric that disassociates the errors (or hits) that occurred in each class. From Table 1 it is possible to derive four performance metrics that directly measure the classification performance on positive and negative classes independently:

- **False negative rate**: $FN_{rate} = \frac{FN}{TP+FN}$ is the percentage of positive cases misclassified as belonging to the negative class;
- **False positive rate**: $FP_{rate} = \frac{FP}{FP+TN}$ is the percentage of negative cases misclassified as belonging to the positive class;
- **True negative rate**: $TN_{rate} = \frac{TN}{FP+TN}$ is the percentage of negative cases correctly classified as belonging to the negative class;
- **True positive rate**: $TP_{rate} = \frac{TP}{TP+FN}$ is the percentage of positive cases correctly classified as belonging to the positive class;

These four performance measures have the advantage of being independent of class costs and prior probabilities. The aim of a classifier is to minimize the false positive and negative rates or, similarly, to maximize the true negative and positive rates. Unfortunately, for most real world applications, there is a tradeoff between FN_{rate} and FP_{rate} and, similarly, between TN_{rate} and TP_{rate}. The ROC[1] graphs [10] can be used to analyze the relationship between FN_{rate} and FP_{rate} (or TN_{rate} and TP_{rate}) for a classifier.

A ROC graph characterizes the performance of a binary classification model across all possible trade-offs between the classifier sensitivity (TP_{rate}) and false alarm (FP_{rate}). ROC graphs are consistent for a given problem even if the distribution of positive and negative instances is highly skewed. A ROC analysis also allows the performance of multiple classification functions to be visualized and compared simultaneously. A standard classifier corresponds to a single point in the ROC space. Point (0, 0) represents classifying all instances as negative, while point (0, 1) represents classifying all instances as positive. The upper left point (0, 1) represents a perfect classifier. One point in a ROC diagram dominates another if it is above and to the left. If point A dominates point B, A outperforms B for all possible class distributions and misclassification costs [2].

Some classifiers, such as the Naïve Bayes classifier or some Neural Networks, yield a score that represents the degree to which an instance is a member of a class. Such ranking can be used to produce several classifiers, by varying the threshold of an instance pertaining to a class. Each threshold value produces a different point in the ROC space. These points are linked by tracing straight lines through two consecutive points to produce a ROC curve[2]. For Decision Trees, we could use the class distributions at each leaf as score or, as proposed

[1] ROC is an acronym for *Receiver Operating Characteristic*, a term used in signal detection to characterize the tradeoff between hit rate and false alarm rate over a noisy channel.

[2] Conceptually, we may imagine varying a threshold from $-\infty$ to $+\infty$ and tracing a curve through the ROC space

in [3], by ordering the leaves by its positive class accuracy and producing several trees by re-labelling the leaves, once at time, from all forecasting negative class to all forecasting positive class in the positive accuracy order.

The area under the ROC curve (AUC) represents the expected performance as a single scalar. The AUC has a known statistical meaning: it is equivalent to the Wilconxon test of ranks, and is equivalent to several other statistical measures for evaluating classification and ranking models [4]. In this work, we use the AUC as the main method for assessing our experiments. The results of these experiments are shown in the next section.

4 Experiments

As the purpose of our study is to understand when class imbalances influence the degradation of performance on learning algorithms, we run our experiments on a series of artificial datasets whose characteristics we are able to control, thus allowing us to fully interpret the results. This is not the case when real datasets are used, as we stated before.

The artificial datasets employed in the experiments have two major controlled parameters. The first one is the distance between the centroids of the two clusters, and the second one is the grade of imbalance. The distance between centroids let us control the "level of difficulty" of correctly classifying the two classes. The grade of imbalance let us analyze if imbalance is a factor for degrading performance by itself.

The main idea behind our experiments is to analyze if class imbalance, by itself, can degrade the performance of learning systems. In order to perform this analysis, we created several datasets. These datasets are composed by two clusters: one representing the majority class and the other one representing the minority class. Figure 2 presents a pictorial representation of four possible instances of these datasets in a two-dimensional space.

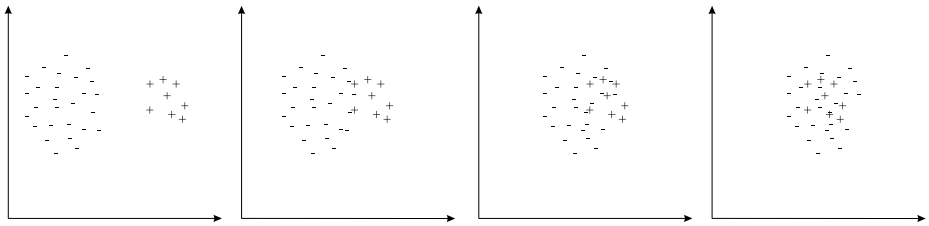


Fig. 2. Pictorial representation of some instances of the artificial datasets employed in the experiments.

We aim to answer several question analyzing the performance obtained on these datasets. The main questions are:

– Is class imbalance a problem for learning systems as it is being stated in several research works? In other words, will a learning system present low performance with a highly imbalanced dataset even when the classes are far apart?
– The distance between the class clusters is a factor that contributes to the poor performance of learning systems in an imbalanced dataset?
– Supposing that the distance between clusters matters in learning with imbalanced datasets, how class imbalance can influence the learning performance for a given distance between the two cluster?

The following section provides a more in deep description of the approach we used to generate the artificial datasets used in the experiments.

4.1 Experiments setup

To evaluate our hypothesis, we generated 10 artificial domains. Each artificial domain is described by 5 attributes, and each attribute value is generated at random, using a Gaussian distribution, with standard deviation 1. Jointly, each domain has 2 classes: positive and negative. For the first domain, the mean of the Gaussian function for both classes is the same. For the following domains, we stepwise add 1 standard deviation to the mean of the positive class, up to 9 standard deviations. For each domain, we generated 12 datasets. Each dataset has 10.000 instances, but having different proportions of instances belonging to each class, considering 1%, 2.5%, 5%, 10%, 15%, 20%, 25%, 30%, 35%, 40%, 45% and 50% of the instances in the positive class, and the remainder in the negative class.

Although the class complexity is quite simple (we generate datasets with only two classes, and each class is grouped in only one cluster), this situation is often faced by machine learning algorithms since most of them, for classification problems, follow the so-called separate-and-conquer strategy, which recursively divides and solves smaller problems in order to induce the whole concept. Furthermore, Gaussian distribution might be used as an approximation of several statistical distributions.

To run the experiments, we chose the algorithm for inducing decision trees C4.5 [11]. C4.5 was chosen because it is quickly becoming the community standard algorithm when evaluating learning algorithms in imbalanced domains. All the experiments were evaluated using 10-fold cross validation. As discussed in Section 3, we used the area under the ROC curve (AUC) as a quality measure. We also implemented the method proposed in [3] to obtain the ROC curves and the corresponding AUCs from the standard classifiers induced by C4.5.

4.2 Results

The results obtained by applying C4.5 in the artificially generated datasets are summarized in Table 2, which shows the mean AUC value and the respective standard deviation in parenthesis, of the classifiers induced by C4.5 for all the

datasets having different class priors and different distances between the positive and negative class centroids. We omitted the values of AUC for the datasets having a distance of class centroids greater or equal than 4 standard deviations since the results are quite similar to the datasets having a distance of 3 standard deviations. Furthermore, for those datasets the difference of AUC are statistically insignificant, with 95% of confidence level, for any proportion of instances in each class. The results with the dataset having class centroids 9 standard deviations apart is included in order to illustrate the small variation between them and the previous column.

Positive	Distance of Class Centroids				
instances	0	1	2	3	9
1%	50.00% (0.00%)	64.95% (9.13%)	90.87% (6.65%)	98.45% (2.44%)	99.99% (0.02%)
2.5%	50.00% (0.00%)	76.01% (6.41%)	95.82% (3.11%)	97.95% (2.12%)	99.99% (0.02%)
5%	50.00% (0.00%)	81.00% (2.86%)	98.25% (1.45%)	98.95% (1.11%)	100.00% (0.00%)
10%	50.00% (0.00%)	86.69% (2.11%)	98.22% (1.14%)	99.61% (0.55%)	99.99% (0.02%)
15%	50.00% (0.00%)	88.41% (2.37%)	98.92% (0.75%)	99.68% (0.49%)	99.99% (0.02%)
20%	50.00% (0.00%)	90.62% (1.44%)	99.08% (0.42%)	99.90% (0.21%)	99.99% (0.02%)
25%	50.00% (0.00%)	90.88% (1.18%)	99.33% (0.32%)	99.90% (0.14%)	99.98% (0.03%)
30%	50.00% (0.00%)	90.75% (0.81%)	99.24% (0.29%)	99.86% (0.14%)	99.99% (0.02%)
35%	50.00% (0.00%)	91.19% (0.94%)	99.36% (0.43%)	99.91% (0.08%)	99.99% (0.02%)
40%	50.00% (0.00%)	90.91% (0.99%)	99.46% (0.10%)	99.90% (0.13%)	99.99% (0.03%)
45%	50.00% (0.00%)	91.73% (0.79%)	99.44% (0.22%)	99.90% (0.09%)	99.98% (0.04%)
50%	50.00% (0.00%)	91.32% (0.68%)	99.33% (0.19%)	99.87% (0.13%)	99.99% (0.03%)

Table 2. AUC obtained from classifiers induced by C4.5 varying class priors and class overlapping

As expected, if both positive and negative classes have the same centroids, we have a constant AUC value of 50%, independently of class imbalance. This AUC value means that all examples are classified as belonging to the majority class.

Consider the column where the centroids of each class are 1 standard deviation appart. If this column is analyzed solely, someone may infer that the degree of class imbalance on its own is the main factor that influences the learning process. The AUC has an upward trend, increasing from nearly 65% when the proportion of instances of positive class is 1% to more than 90% when the proportion of positive and negative instances are the same. However, when the class centroids distance goes up to 2 standard deviations, we can see that the influence of the class priors becomes weaker. For instance, the value of AUC for the classifiers induced with the dataset having 1% and 2.5% of instances in the positive class and the centroid of this class 2 standard deviations apart the centroid of the negative class is still worst than the classifiers induced changing the class distribution and the same centroids, but the values of AUC are closer than the values with the same proportion and the difference of the centroids is 1 standard deviation.

For classifiers induced with datasets having 3 or more standard deviations apart, the problem becomes quite trivial, and the AUC values are nearly 100% regardless of the class distribution.

For a better visualization of the overall trends, these results are shown graphically in Figure 3 and 4. These graphs show the behavior of the C4.5 algorithm assessed by the AUC metric in both class imbalance and class overlapping.

Figure 3 plots the percentage of positive instances in the datasets *versus* the AUC of the classifiers induced by C4.5 for different centroids of positive class (in standard deviations) from the negative class. The curves with centroids of positive class 3 to 8 standard deviations apart are omitted for a better visualization, but the curves are quite similar to the curve with centroid 9 standard deviations apart the negative class. Consider the curves of positive class where the class centroids are 2 and 3 standard deviations apart. Both classifiers have good performances, with AUC higher than 90%, even if the proportion of positive class is barely 1%. Particularly, the curve where the positive class centroid is 9 standard deviations from the negative class centroid represents almost a perfect classifier, independently of the class distribution.

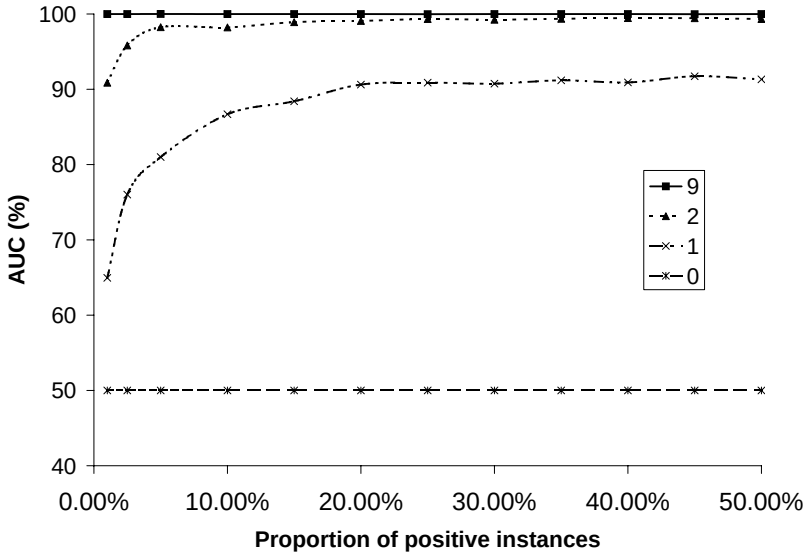


Fig. 3. Variation in the proportion of positive instances *versus* AUC

Figure 4 plots the variation of centroids distances *versus* the AUC of the classifiers induced by C4.5 for different class imbalances. The curves that represent the proportion of positive instances between 20% and 45% are omitted for visualization purposes since they are quite similar to the curve that represents equal proportion of instances in each class. In this graph, we can see that the main degradation in the classifiers performances occurs mainly when the difference between the centre of the positive and negative class is 1 standard deviation. In this case, the degradation is significantly higher for highly imbalanced

datasets, but decreases when the distance between the centre of the positive and negative class increases. The differences in performance of classifiers are statistically insignificant when the difference between the centers goes up 4 standard deviations, independently on how many instances belongs to the positive class.

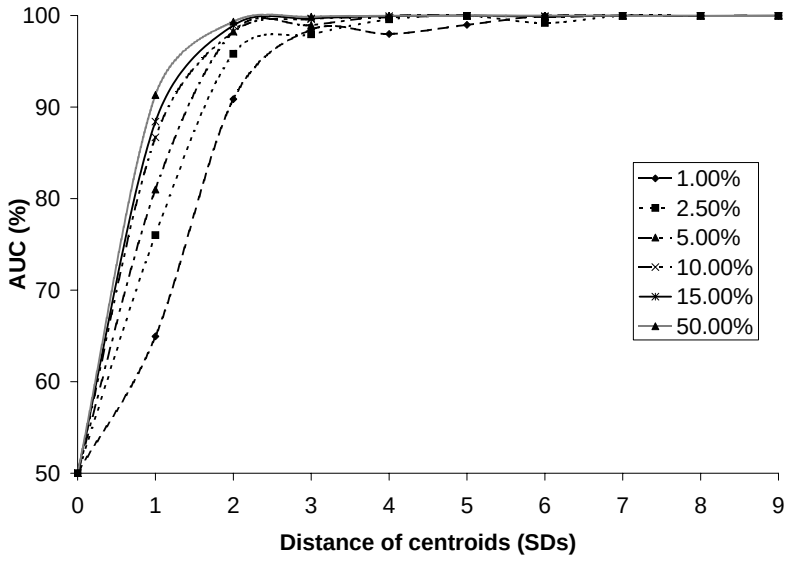


Fig. 4. Variation in the centre of positive class *versus* AUC

Analyzing the results, it is possible to see that class overlapping have an important role in the concept induction, even stronger than class imbalance. Those trends seem to validate our formerly hypothesis, presented in Section 2.

5 Conclusion and future work

Class imbalance is often reported as an obstacle to the induction of good classifiers by machine learning algorithms. However, for some domains, machine learning algorithms are able to achieve meaningful results even in the presence of highly imbalanced datasets.

In this work, we develop a systematic study using a set of artificially generated datasets aiming to show that the degree of class overlapping has a strong correlation with class imbalance. This correlation, to the best of our knowledge, has not been previously analyzed elsewhere in the machine learning literature. A good understanding of this correlation would be useful in the analysis and development of tools to treat imbalanced data or in the (re)design of learning algorithms for practical applications.

In order to study this question in more depth, several further approaches can be taken. For instance, it would be interesting to vary the standard deviations of the Gaussian functions that generate the artificial datasets. It is also worthwhile to consider the generation of datasets where the distribution of instances of the minority class is separated in several small clusters. This approach can lead the study of the class imbalance problem together with the small disjunct problem, as proposed in [6]. Another point to explore is to analyze the ROC curves obtained from the classifiers. This approach might produce some useful insights in order to develop or analyze methods for dealing with class imbalance. Last but not least, experiments should also be conducted on real-world datasets in order to verify that the hypothesis presented in this work does apply to them.

Acknowledgements This research is partially supported by Brazilian Research Councils CAPES and FAPESP.

References

1. N. Chawla, N. Japkowicz, and A. Kolcz, editors. *ICML'2003 Workshop on Learning from Imbalanced Data Sets (II)*, 2003. Proceedings available at `http://www.site.uottawa.ca/~nat/Workshop2003/workshop2003.html`.
2. C. Drummond and R. C. Holt. Explicity representing expected cost: An alternative to roc representation. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 198–207, 2000.
3. C. Ferri, P. Flach, and J. Hernndez-Orallo. Learning decision trees using the area under the ROC curve. In C. S. A. Hoffman, editor, *Nineteenth International Conference on Machine Learning (ICML-2002)*, pages 139–146. Morgan Kaufmann Publishers, 2002.
4. D. J. Hand. *Construction and Assessment of Classification Rules*. John Wiley and Sons, 1997.
5. N. Japkowicz, editor. *AAAI Workshop on Learning from Imbalanced Data Sets*, Menlo Park, CA, 2003. AAAI Press. Techical report WS-00-05.
6. N. Japkowicz. Class imbalances: Are we focusing on the right issue? In *Proc. of the ICML'2003 Workshop on Learning from Imbalanced Data Sets (II)*, 2003.
7. N. Japkowicz and S. Stephen. The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6(5):429–450, 2002.
8. J. Laurikkala. Improving Identification of Difficult Small Classes by Balancing Class Distributions. Technical Report A-2001-2, University of Tampere, Finland, 2001.
9. C. J. Merz and P. M. Murphy. UCI Repository of Machine Learning Datasets, 1998. `http://www.ics.uci.edu/~mlearn/MLRepository.html`.
10. F. J. Provost and T. Fawcett. Analysis and Visualization of Classifier Performance: Comparison under Imprecise Class and Cost Distributions. In *Knowledge Discovery and Data Mining*, pages 43–48, 1997.
11. J. R. Quinlan. *C4.5 Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1993.
12. 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.