An Experimental Design to Evaluate Class Imbalance Treatment Methods

Gustavo Batista, Diego Silva
Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo
São Carlos, Brazil
{gbatista, diegofsilva}@icmc.usp.br

Ronaldo Prati
Centro de Matemática, Computação e Cognição
Universidade Federal do ABC
Santo André, Brazil
ronaldo.prati@ufabc.edu.br

Abstract—In the last decade, class imbalance has attracted a huge amount of attention from researchers and practitioners. Class imbalance is ubiquitous in Machine Learning, Data Mining and Pattern Recognition applications; therefore, these research communities have responded to such interest with literally dozens of methods and techniques. Surprisingly, there are still many fundamental open-ended questions such as “Are all learning paradigms equally affected by class imbalance?”, “What is the expected performance loss for different imbalance degrees?” and “How much of the performance losses can be recovered by the treatment methods?”. In this paper, we propose a simple experimental design to assess the performance of class imbalance treatment methods. This experimental setup uses real data sets with artificially modified class distributions to evaluate classifiers in a wide range of class imbalance. We employ such experimental design in a large-scale experimental evaluation with twenty-two data sets and seven learning algorithms from different paradigms. Our results indicate that the expected performance loss, as a percentage of the performance obtained with the balanced distribution, is quite modest (below 5%) for the most balanced distributions up to 10% of minority examples. However, the loss tends to increase quickly for higher degrees of class imbalance, reaching 20% for 1% of minority class examples. Support Vector Machine is the classifier paradigm that is less affected by class imbalance, being almost insensitive to all but the most imbalanced distributions. Finally, we show that the sampling algorithms only partially recover the performance losses. On average, typically about 30% or less of the performance that was lost due to class imbalance was recovered by random oversampling and SMOTE.

Keywords—Class imbalance, experimental setup, sampling methods

I. INTRODUCTION

In the last decade, class imbalance has attracted a huge amount of attention from researchers and practitioners. Class imbalance is ubiquitous in Machine Learning, Data Mining and Pattern Recognition, as it has been extensively documented in the literature with applications such as diagnostics of rare diseases [1], fraud detection [2], identification of oil spills in satellite radar images [3], and many others. Literally hundreds of research papers have reported that imbalanced data lead to performance losses, and some sort of treatment (such as sampling [4], cost-sensitive learning [5], ensembles [6], among others) are able to improve classification.

Although the relationship between class imbalance and performance loss is well documented, we argue that it is under-comprehended. For instance, we have been inquired several times by different researchers about “from what distribution can a data set be considered imbalanced?”. Although it is a quite naïve question, we were never able to fully answer it, since we are not aware of any definitive study relating performance loss, degree of class imbalance and learning systems.

Virtually every paper about class imbalance has the exact same experimental setup. A proposed method is compared against one or two competing methods over a dozen or so data sets. Although this experimental setup is enough to support an argument that the new method is as good as or better than the state-of-the-art, it still leaves many unanswered questions. For sake of clarity, let us use an example: suppose that for a given application, a classifier over imbalanced data results in 80% AUC, and after the application of a certain treatment method, we obtain 90% AUC. It is clearly a significant improvement in classification; however, “were we able to fully recover the losses caused by class imbalance?”

We believe this question is extremely relevant and completely unanswered by the current research in class imbalance. The literature has provided dozens of methods to treat class imbalance, and there is (or will be) no single method able to provide the best performance for all data sets. Therefore, given a problem in which we still need to improve the classification performance after the application of a treatment method A, we would like to know if it is worth seeking for a different method B to apply instead of A, hoping that B will outperform A. A side effect is that it is difficult to compare results across domains as, say, a 5% absolute increment in AUC may have different relative meanings in different domains.

The reader should have anticipated that these two questions are directly related. If we know the relationship between imbalance degree and performance loss, we can evaluate when a treatment method has recovered most of the loss. Unfortunately, no existing analysis is able to provide the answers we are looking for every pair of data set and learning system. However, we can answer these questions in
terms of expected performance.

In this paper, we propose a simple experimental design to assess the performance of class imbalance treatment methods and classifiers. This experimental setup uses real data sets with artificially modified class distributions to evaluate classifiers in a wide range of class imbalance. We divided the evaluation in two parts. The first part consists in inducing a classifier for each class distribution and measuring the performance loss compared to the balanced distribution. We found out that Support Vector Machine is the classifier paradigm that is the least affected by class imbalance, being insensitive to almost all but the most imbalanced distributions.

The second part consists of applying a treatment method and inducing a classifier for each class distribution. This time we measured the percentage of the performance loss that was recovered by the treatment method. In other words, 100% represents that the classifier induced after the application of the treatment method obtained the same performance of the classifier induced over (original) balanced data. We used two well-known over-sampling methods, random over-sampling and SMOTE. We show that the expected performance recovery for both methods is typically about 30% or less for the most imbalanced distributions.

We believe that papers that propose new treatment methods for class imbalance should adopt the proposed experimental setup. The proposed setup allows not just comparing a new method to competing methods, but also to compare the new method to a reference performance provided by the balanced data set.

This work is organised as follows: Section II describes the experimental design proposed in this work; Section III analyses how learning systems belonging to different paradigms perform under different class imbalance degrees; Section IV analyses the ability of two sampling techniques to recover the performance lost due to class imbalance; Section V discusses possible limitations of our approach; Section VI compares the proposed approach to other experimental setups in the literature; Section VII briefly describes the inducers and their parameters; finally, Section VIII presents our conclusion and suggestions for future work.

II. EXPERIMENTAL DESIGN

Due to lack of space we assume the reader is familiar with the class imbalance problem and methods. In the case this assumption does not hold, the literature has several comprehensible surveys, we recommend [7]. For the same reason, we are not able to include here all numerical results obtained in our experiments. Therefore, we decided to create a paper website [8] that has detailed results, including tables and data; however, we note this paper is totally self-contained.

Our experimental design is inspired by the design used in [9]. The central idea is to generate several training set distributions with increasing degrees of class imbalance while maintaining the training sets with a fixed number of examples. Our training set distributions range from the balanced distribution (denoted as 50/50) to the imbalanced distributions of 1/99 and 99/1. In order to generate these class distributions, we restricted the training sets to have a total number of instances equals to the number of instances of the minority class. We avoid using extremely imbalanced data sets; in particular, the combination of a small data set with large class imbalance would result in a training set with too few examples. We return to this discussion in Section V, where we comment possible limitations of this work.

The test sets have the naturally occurring class distributions. We reserved 25% of the original data as test set using a stratified sample, and did not touch this portion of the data until the final evaluation; the remaining 75% was used as training set, with class distributions manipulated as previously described. In order to reduce variance and increase statistical significance, we repeated the whole process 100 times using different train and test sample partitions.

For this specific study we assembled a database with twenty data sets. Since we want to promote reproducibility, most of them are public domain benchmark data sets available in repositories such as UCI Machine Learning Repository [10] or used in projects like Statlog [11]. A few data sets are not public domain, but we decided to include them since they are frequently used in studies involving class imbalance. These data sets include tumour identification in mammography images [12], [13]. We also included data sets obtained in past research, and we make them publicly available for the first time in the paper website.

We use the area under the ROC curve (AUC) [14] as the main measure to assess our results. Since we chose AUC and also because we want to control the degree of class imbalance, we converted all non-binary class data sets into binary class, associating a positive label to one of the classes (usually the minority one) and aggregating all remaining classes into a negative class. Table I presents a summarised description of the data sets included in our study. The table lists the data sets full names, as well as a short identifier used in this paper, the number of instances and attributes, labels associated with the positive and negative classes and attribute class distribution. The data sets are listed in increasing order of class imbalance.

Two data sets resulted in two entries each in Table I, because different classes were used as positive class. For the Letter dataset, Letter-a is the variation in which the positive class is the original letter “a” class; and Letter-vowel has the positive class composed by all classes that represent vowels. For the Splice-junction Gene Sequences dataset, one entry has the intron-exon (“ie”) boundaries as positive class.

1We use the notation $X/Y$, with $X + Y = 100$ to denote that for a set of 100 instances, $X$ belongs to the positive class and $Y$ belongs to the negative class.

96
and the other has the exon-intron ("ei") boundaries. The final number of data sets is twenty-two, considering the four entries generated from these two data sets.

We use this experimental design to answer the questions raised in abstract and introduction. We start discussing the expected performance loss for different learning paradigms in the next section.

III. CLASS IMBALANCE, PERFORMANCE LOSS AND LEARNING PARADIGMS

We included at least one representative of each major learning paradigm. We selected C4.5 (decision trees), C4.5Rules (rules extracted from decision trees), CN2 and RIPPER (decision rules), Back-propagation Neural Network (connectionism), Na"ive Bayes (Probabilistic) and Support Vector Machines (Statistical Learning).

The results are summarised in Table II. Due to lack of space, we only present the average results for all data sets. The interested reader can find detailed results in the paper website [8]. Each reported value is a performance loss relative to the balanced distribution, as defined by Equation 1:

\[
L = \frac{B - I}{B}
\]

where \(B\) stands for the performance obtained with the balanced distribution and \(I\) for the performance obtained with an imbalanced distribution. As previously described, \(B\) and \(I\) are measured in this work as the area under the ROC curve (AUC).

At this point we can review the question "from what distribution can a data set be considered imbalanced?" Technically, the answer is every non-balanced distribution, since most learning systems (except SVM) show some degree of performance loss for every non-balanced distribution. Obviously, some practitioners might consider small losses insignificant; in that case the distributions in the range 20/80 to 80/20 usually present small losses, on average bellow 2%. Imbalanced distributions above and including 10/90 (and 90/10) tend to have more expressive losses, above 5%; and the most imbalanced distributions in our study (1/99 and 99/1) had losses around 20% on average. From the practical standpoint, we can answer the previous question by saying that for most learning systems the losses due to class imbalance start to be significant when the minority class represents 10% of the data set or less.

We also posed the following question: ‘are all learning paradigms equally affected by class imbalance?’ As we can see from the results in Table II the answer is clearly no. RIPPER is the learning algorithm that was the most affected by class imbalance. In contrast, SVM is very little affected by all but the most imbalanced distributions, obtaining even negative performance losses, i.e., small performance gains compared to the balanced distribution. A possible explanation is that SVM frequently uses few support vectors to determine the separation between classes, as previously observed by [15]. However, SVM still suffer the consequences of severe class imbalance, as noted by [16]. The authors conducted an experiment using two data sets with class ratios of 10:1 and 10000:1. In the second data set, the edge separation between classes tended to become more located over the space belonging to the minority class in comparison with the first data set. Therefore, the SVM inducer tended to classify more test cases as belonging to the majority class. In extreme situations, when the number of training examples from the minority class is not enough to characterise the decision space of this class, SVM might classify every instance as belonging to the majority class.

In the next section we use the same experimental design to measure the capacity of treatment methods to recover the performance losses.

and the other has the exon-intron ("ei") boundaries. The final number of data sets is twenty-two, considering the four entries generated from these two data sets.

We use this experimental design to answer the questions raised in abstract and introduction. We start discussing the expected performance loss for different learning paradigms in the next section.

III. CLASS IMBALANCE, PERFORMANCE LOSS AND LEARNING PARADIGMS

We included at least one representative of each major learning paradigm. We selected C4.5 (decision trees), C4.5Rules (rules extracted from decision trees), CN2 and RIPPER (decision rules), Back-propagation Neural Network (connectionism), Na"ive Bayes (Probabilistic) and Support Vector Machines (Statistical Learning).

The results are summarised in Table II. Due to lack of space, we only present the average results for all data sets. The interested reader can find detailed results in the paper website [8]. Each reported value is a performance loss relative to the balanced distribution, as defined by Equation 1:

\[
L = \frac{B - I}{B}
\]

where \(B\) stands for the performance obtained with the balanced distribution and \(I\) for the performance obtained with an imbalanced distribution. As previously described, \(B\) and \(I\) are measured in this work as the area under the ROC curve (AUC).

At this point we can review the question “from what distribution can a data set be considered imbalanced?” Technically, the answer is every non-balanced distribution, since most learning systems (except SVM) show some degree of performance loss for every non-balanced distribution. Obviously, some practitioners might consider small losses insignificant; in that case the distributions in the range 20/80
IV. CLASS IMBALANCE, PERFORMANCE RECOVERY AND
TREATMENT METHODS

So far we were able to characterise the expected performance losses for different class distributions and learning systems. These results lead to another question: "how much of the performance losses can be recovered by the treatment methods?" In this section we use the same experimental design to answer this question.

A myriad of methods has been proposed to treat class imbalance. Unfortunately, due to lack of space, we can only include the results of a couple of them in this paper. We chose to analyse Random Over-sampling and SMOTE [12] for the following reasons: first, our previous experience with sampling methods [17] shows that these two methods outperform several other (over and under) sampling methods; second, SMOTE has become one of the most popular sampling methods for class imbalance, and it is frequently used as counterpart in empirical evaluations and has several proposed extensions and variations (for instance [18], [19]).

These two sampling methods were applied in the same experimental setup used in the previous section. We saved all data partitions and applied the treatment methods in the exact same data used to measure the performance loss. The sampling methods were applied to all non-balanced data sets and new minority class examples were created until the training set became perfectly balanced. Once again, the test sets were not touched, and they keep the naturally occurring class distributions.

Our assessment measure is the performance recovery, measured as percentage of the performance loss defined by Equation 1. The idea is to calculate the performance loss for the treated data set in the same way we did in the previous section for the imbalanced data. We include the equation here for sake of clarity:

\[ L_T = \frac{B - T}{B} \]  

(2)

where \( B \) stands for the performance obtained with the balanced distribution and \( T \) for the performance obtained with the treated data, both measured in AUC.

In a second step, we calculate the performance recovery as a fraction of the performance loss for treated data over the performance loss for imbalanced data:

\[ R = \frac{L - L_T}{L} \]  

(3)

A simple way to understand Equation 3 is to think that \( R = 100\% \) when \( L_T = 0\% \), or in other words, the performance recovery will reach 100% when there is no performance loss for treated data. In that case, the classifier performance for treated data is the same as the performance for the balanced distribution.

Table III presents the expected performance recovery for random over-sampling. Notice the large negative values for the distribution near the balanced one. These results mean that random over-sampling was not able to recover the losses associated to these distributions. Instead, random over-sampling resulted in classifiers with worse performance than the classifiers directly induced over imbalanced data. Of course, these numbers should be analysed with a word of caution: in general the losses associated with these distributions are quite small, and the random over-sampling losses in term of absolute values are less impressive than the percentage values.

Certainly, the most interesting values are the performance recovery associated with large class imbalance. For the class distributions of 1/99, 5/95 and 10/90 (and their negative class counterparts) we noticed that although some scattered number present recoveries above 50%, most results are below 30%. In our opinion, these numbers represent a rather modest performance recovery, especially if we consider that random over-sampling frequently provides competitive results with other sampling methods.

The results also show not all learning systems are equally benefited by the replication of minority class examples. For example, the performance of Naïve Bayes was not improved for any class distribution. Apparently, the simple replication of examples does not help to correct the conditional probability estimates made by Naïve Bayes. These results contribute to the common criticism against the early class imbalance research that (inadvertently) considered fair to extrapolate to other inducers the results obtained with C4.5.

Table IV presents the expected performance recovery for SMOTE. This method does not replicate examples from the minority class, but interpolates cases of this class in order to expand its decision space. A first observation about SMOTE

<table>
<thead>
<tr>
<th>Inducer</th>
<th>1/99</th>
<th>5/95</th>
<th>10/90</th>
<th>20/80</th>
<th>30/70</th>
<th>40/60</th>
<th>50/50</th>
<th>60/40</th>
<th>70/30</th>
<th>80/20</th>
<th>90/10</th>
<th>95/5</th>
<th>99/1</th>
</tr>
</thead>
<tbody>
<tr>
<td>C4.5RULES</td>
<td>3.50</td>
<td>11.01</td>
<td>6.66</td>
<td>2.86</td>
<td>1.27</td>
<td>0.47</td>
<td>0.00</td>
<td>0.58</td>
<td>1.30</td>
<td>2.85</td>
<td>6.62</td>
<td>11.05</td>
<td>24.81</td>
</tr>
<tr>
<td>CN2</td>
<td>18.07</td>
<td>6.09</td>
<td>3.00</td>
<td>1.15</td>
<td>0.27</td>
<td>0.09</td>
<td>0.00</td>
<td>0.38</td>
<td>0.74</td>
<td>1.32</td>
<td>3.06</td>
<td>5.90</td>
<td>15.34</td>
</tr>
<tr>
<td>C4.5</td>
<td>22.39</td>
<td>9.15</td>
<td>4.68</td>
<td>1.68</td>
<td>0.56</td>
<td>0.32</td>
<td>0.00</td>
<td>0.20</td>
<td>0.81</td>
<td>1.78</td>
<td>4.64</td>
<td>8.98</td>
<td>22.94</td>
</tr>
<tr>
<td>NEURAL NETWORK</td>
<td>25.00</td>
<td>7.62</td>
<td>4.61</td>
<td>1.96</td>
<td>0.89</td>
<td>0.34</td>
<td>0.00</td>
<td>0.07</td>
<td>0.15</td>
<td>0.90</td>
<td>2.81</td>
<td>5.60</td>
<td>22.32</td>
</tr>
<tr>
<td>Ripper</td>
<td>31.75</td>
<td>19.81</td>
<td>13.18</td>
<td>6.44</td>
<td>2.96</td>
<td>1.25</td>
<td>0.00</td>
<td>0.27</td>
<td>2.01</td>
<td>5.09</td>
<td>10.44</td>
<td>15.62</td>
<td>28.11</td>
</tr>
<tr>
<td>SVM</td>
<td>12.41</td>
<td>1.93</td>
<td>-1.27</td>
<td>-4.21</td>
<td>-3.92</td>
<td>-2.41</td>
<td>-0.00</td>
<td>-2.54</td>
<td>-2.47</td>
<td>-2.15</td>
<td>-0.95</td>
<td>2.21</td>
<td>12.88</td>
</tr>
<tr>
<td>Average</td>
<td>23.50</td>
<td>9.08</td>
<td>5.13</td>
<td>1.69</td>
<td>0.40</td>
<td>0.04</td>
<td>0.00</td>
<td>-0.13</td>
<td>0.41</td>
<td>1.53</td>
<td>4.27</td>
<td>7.90</td>
<td>20.09</td>
</tr>
</tbody>
</table>
is that it does not outperform random over-sampling by a large margin for all inducers. However, it clearly performs better for Naïve Bayes inducer, providing some expressive performance recoveries where random over-sampling could not recover any loss at all. We should note that SMOTE is considerably more computationally expensive than random over-sampling. Therefore, this performance gain obtained with some inducers comes with the cost of additional computational time.

In general, we can say that the performance recoveries for both methods were rather modest. Although some scattered results present gains above 50%, most of the performance recoveries are around or bellow 30%.

V. LIMITATIONS

We reserve this section to discuss some limitations of the proposed experimental setup. The most obvious limitation is that this experimental setup is restricted to binary-class problems. Although it is possible to extend the approach to more than two classes, the number of results to be analysed would increase exponentially. In contrast, binary-class problems are quite common in imbalanced domains, in which frequently the positive class is assigned to a class of interest and the negative class is associated to all remaining objects.

A less obvious limitation is the size of the training sets. Since we generated several distributions, from the positive class being a 1% minority class up to the positive class being a 99% majority class, we had to restrict the size of the training set as the size of the minority class. In order to partially overcome this limitation, we looked for more balanced data sets or imbalanced data sets with larger number of instances. However, this limitation prevented us to use in our experimental evaluation some imbalanced data sets frequently present in imbalanced data papers.

A possible criticism is that the restricted sizes of the training sets might be biasing the results in some way. For instance, some inducers that better deal with smaller data sets might be favoured. However, notice that we do not report any absolute performance results; all results are relative to the performance obtained with the balanced distribution. In addition, the training sets have a constant number of instances for all class distributions.

We should note, however, that the restricted size of the training sets may lead to very few minority class examples for the most imbalanced distributions. It is not uncommon to find data sets in which the minority class is represented by a dozen or so examples under these class distributions. We must note that this not a very uncommon situation, and it known as absolute rarity [20]. The literature has several examples of application domains that had class imbalance of the order of 1:1000, 1:10000, or superior. In these domains, the learning systems almost invariably have to deal with similar absolute number of minority class examples.

VI. RELATED WORK

The literature on class imbalance has several experimental papers with large-scale comparisons of learning algorithms or treatment methods. In this section, we compare the experimental setups of these papers to the one used here.

As we previously observed, our experimental design is inspired by the setup used in [9]. Weiss & Provost varied the training set class distribution in order to identify which distribution should be used when a limited number of examples is available. They conclude that, when AUC measure is used, although no single class distribution is able to provide the best-performing classifier, the balanced distribution performs well.

We used the results of Weiss & Provost to elect the balanced distribution as the reference distribution in our experiments. Although the results in [9] were obtained with the C4.5 decision tree inducer only, our results conform with their results. As the reader can observe in Table II, the average results for all inducers present positive losses for all non-balanced distributions. The exception is SVM, in which the best distribution was 20/80 with an average improvement of 4.21% over the balanced distribution. These are average results over all data sets, we invite the interested reader to check the paper website for detailed (per data set) results.

An interesting counterpart to the experimental design of this paper is present in [21]. In that work, Cieslak & Chawla investigate classifier performance when testing class distributions change substantially. For instance, a disease outbreak may change the class prior probability considerably, making it differ substantially from the class distribution used in training. Therefore, in their experimental setup a classifier
is compared against several test sets with different class distributions.

Another experimental work is presented in [22]. Khoshgoftaar et al. are interested in studying classification performance when one class is rare. In their experimental setup, the positive class is represented by 5, 10, 20 or 40 instances, and the negative class is set up in such a way that the positive class represents from 65% to 1% of the total number of examples. For instance, when there are 5 positive examples, the negative examples can be as low as 3 examples (65% positive class) up to 495 (1% positive class). Khoshgoftaar et al. conclude that the balanced distribution is outperformed by the distributions 2:1 (negative:positive) for 10, 20 and 40 positive examples, and by the distribution 3:1 for 5 positive examples.

An interesting feature of the experimental design used in [22] is to allow the analysis of classification performance factorized by class rarity and class distribution. Our experimental design does not allow the same analysis since, for instance, a 1% class distribution may represent different absolute number of positive examples in different data sets. In contrast, our experimental design leads to training sets with fixed number of examples and, therefore, the results are not influenced by the training sets sizes. The Khoshgoftaar et al. design cannot completely factor out the influence of training set sizes. This influence might even help to explain why the most rare configuration (with 5 positive examples) required a 3:1 class distribution when the other less rare distributions required only 2:1. We note that in this design, the most balanced distributions also have smaller training sets sizes.

All these papers, however, report results in terms of absolute performance values. As we argued in the Introduction, this approach cannot answer relevant question about the class imbalance problem.

### VII. Inducers and Parameters

In this section we briefly describe de inducers and their parameters. Whenever possible, we used the original implementations of the inducers, that is the case for C4.5, C4.5Rules, CN2 and Ripper.

C4.5 and C4.5Rules are the original implementation provided by Quinlan [23]. We used the default parameters, with exception of pruning. Several research papers have pointed out that C4.5 pruning is hardly beneficial in imbalanced domains. Therefore, we induced unpruned trees. CN2 is the original implementation provided by Clark & Niblett [24], we used it with default parameters. The same occurred to Ripper, which is the implementation provided by Cohen [25].

For Naive Bayes and Neural Networks, we used the implementation provided by Borgelt [26]. In order to estimate conditional probabilities, Naive Bayes uses a frequency table for symbolic attributes and normal distribution for continuous attributes. For Neural Networks, most domains required a simple configuration with no hidden layer, and the learning lasted for 1000 epochs. For SVM we used LIBSVM [27] with radial basis kernel with degree 3.

### VIII. Conclusion and Future Work

This paper proposes an experimental design to evaluate the influence of class imbalance in classifiers performance. This experimental design allows to evaluate the performance loss caused by different degrees of class imbalance, as well as to measure the performance recovery obtained by treatment methods.

We conducted an experiment to answer some open-ended questions about class imbalance, and concluded that all evaluated classifiers are affected by the class imbalance problem. All classifiers, except SVM, present some loss of performance for all class distributions. This loss tends to be more expressive as the classes become more imbalanced. Additionally, SVM seems to be only susceptible to absolute rarity. For SVM, performance loss occurred for the proportions 1/99 and 99/1. In these proportions the data sets have only 1% minority class cases and, considering the restricted size of the training sets, the absolute number of minority class examples is very limited, characterising absolute rarity.

We also made experiments with treated data by sampling methods. We measured how much class imbalance performance loss was recovered by artificially rebalancing the data. The over-sampling methods evaluated were able to occasionally recover a significant proportion (between 50% and 60%) of the performance lost. However, for the majority of the executions, the performance recovery was below 30%, which can be considered a quite modest recovery rate.

### Table IV

<table>
<thead>
<tr>
<th>Inducer</th>
<th>Class Distribution (positive/negative)</th>
<th>1/99</th>
<th>5/95</th>
<th>10/90</th>
<th>20/80</th>
<th>30/70</th>
<th>40/60</th>
<th>50/50</th>
<th>60/40</th>
<th>70/30</th>
<th>80/20</th>
<th>90/10</th>
<th>95/5</th>
<th>99/1</th>
</tr>
</thead>
<tbody>
<tr>
<td>C4.5RULES</td>
<td></td>
<td>14.06</td>
<td>26.54</td>
<td>17.38</td>
<td>11.38</td>
<td>27.46</td>
<td>-21.59</td>
<td>-86.50</td>
<td>15.28</td>
<td>42.43</td>
<td>47.75</td>
<td>42.96</td>
<td>36.35</td>
<td></td>
</tr>
<tr>
<td>CN2</td>
<td></td>
<td>-1.05</td>
<td>-9.43</td>
<td>-7.19</td>
<td>-74.90</td>
<td>-398.09</td>
<td>-477.27</td>
<td>0.00</td>
<td>-131.46</td>
<td>-47.88</td>
<td>-79.81</td>
<td>-26.50</td>
<td>11.30</td>
<td>-0.43</td>
</tr>
<tr>
<td>C4.5</td>
<td></td>
<td>1.87</td>
<td>19.44</td>
<td>3.91</td>
<td>-23.41</td>
<td>-20.33</td>
<td>-90.32</td>
<td>0.00</td>
<td>-296.90</td>
<td>-11.95</td>
<td>-4.88</td>
<td>27.78</td>
<td>34.04</td>
<td>28.71</td>
</tr>
<tr>
<td>NEURAL NETWORK</td>
<td></td>
<td>-1.50</td>
<td>12.13</td>
<td>15.64</td>
<td>3.62</td>
<td>-93.02</td>
<td>-201.03</td>
<td>0.00</td>
<td>-344.93</td>
<td>-14.17</td>
<td>15.12</td>
<td>5.16</td>
<td>29.83</td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td></td>
<td>33.83</td>
<td>62.77</td>
<td>71.69</td>
<td>81.34</td>
<td>81.01</td>
<td>109.79</td>
<td>0.00</td>
<td>-101.55</td>
<td>83.16</td>
<td>64.17</td>
<td>63.63</td>
<td>56.83</td>
<td>41.61</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>13.43</td>
<td>19.97</td>
<td>35.64</td>
<td>15.51</td>
<td>44.96</td>
<td>-83.13</td>
<td>0.00</td>
<td>-202.59</td>
<td>17.89</td>
<td>22.23</td>
<td>20.00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This paper proposes an experimental design to evaluate the influence of class imbalance in classifiers performance. This experimental design allows to evaluate the performance loss caused by different degrees of class imbalance, as well as to measure the performance recovery obtained by treatment methods. We conducted an experiment to answer some open-ended questions about class imbalance, and concluded that all evaluated classifiers are affected by the class imbalance problem. All classifiers, except SVM, present some loss of performance for all class distributions. This loss tends to be more expressive as the classes become more imbalanced. Additionally, SVM seems to be only susceptible to absolute rarity. For SVM, performance loss occurred for the proportions 1/99 and 99/1. In these proportions the data sets have only 1% minority class cases and, considering the restricted size of the training sets, the absolute number of minority class examples is very limited, characterising absolute rarity. We also made experiments with treated data by sampling methods. We measured how much class imbalance performance loss was recovered by artificially rebalancing the data. The over-sampling methods evaluated were able to occasionally recover a significant proportion (between 50% and 60%) of the performance lost. However, for the majority of the executions, the performance recovery was below 30%, which can be considered a quite modest recovery rate.
As future work we consider to evaluate additional sampling methods as well as other approaches to treat imbalanced data, such as ensembles and cost-sensitive learning.

ACKNOWLEDGMENTS
We thank the anonymous reviewers for their comments on the draft of this paper. We also thank Nitesh Chawla for providing the Microcalcifications in Mammography data sets. This work was funded by FAPESP awards 2011/04054-2 and 2012/07295-3.

REFERENCES