Class Imbalances: Are we Focusing on the Right Issue?

Nathalie Japkowicz NAT@SITE.UOTTAWA.CA
School of Information Technology and Engineering, University of Ottawa, 800 King Edward Avenue, P.O. Box 450 Stn. A, Ottawa, Ontario, Canada K1N 6N5

Abstract

It is often assumed that class imbalances are responsible for significant losses of performance in standard classifiers. The purpose of this paper is to question whether class imbalances are truly responsible for this degradation or whether it can be explained in some other ways. Our experiments suggest that the problem is not directly caused by class imbalances, but rather, that class imbalances may yield small disjuncts which, in turn, will cause this degradation. We argue that it may, then, be more useful to focus on the small disjunct problem directly than it is to focus on the class imbalance problem, and we summarize two of our previous studies that do just that.

1. Introduction

Although class imbalances have been reported to hinder the performance of standard classifiers on many different types of problems[1], no study has made a point of linking the class imbalance problem directly to this loss. As a matter of fact, although the performance of standard classifiers may decrease on many imbalanced domains, that does not prove that it is the imbalance, per se, that causes that decrease. Rather, it is quite possible that class imbalances yield certain conditions that hamper classification, which would suggest 1) that class imbalances are not necessarily always a problem and, perhaps even more importantly, 2) that dealing with class imbalances will not always help improve performance.

The purpose of this paper is to question whether class imbalances are truly to blame for the reported losses of performance or whether these deficiencies can be explained in some other way. We show that class imbalances are, actually, not a problem by themselves, but that, in small and complex data sets, they come accompanied with the problem of small (hidden) disjuncts (Holte et al. 89) which in turn causes a degradation in standard classifiers' performance. We conclude the paper by summarizing two approaches that we have previously designed (and described in greater length, elsewhere) that counter the effect of small hidden disjuncts.

The remainder of the paper is divided into five sections. Section 2 describes the artificial domains on which our study is based. Section 3 shows the results of some experiments conducted on these domains. These results suggest that class imbalances are often problematic, but not always. Section 4 shows the results of further experiments that contrast the class imbalance problem to the problem of small hidden disjunct. These results show that it is the small hidden disjunct problem rather than the class imbalance problem that is to blame for the loss of performance. Section 5 summarizes two approaches that we previously proposed to deal with the class imbalance/small disjunct problem, And, finally, section 6 concludes the paper.

2. Domain Generation

Because the purpose of our study is to understand the nature of the observed degradation in domains that present a class imbalance, rather than conducting our study on real-world domains whose results would be difficult to decipher, we chose to run our experiments on a series of artificial domains whose characteristics we could carefully control. In particular, we created 125 domains with various combinations of three parameters which we deemed significant for our study: *concept complexity*, *training set size*, and *degree of imbalance* . The generation method used was inspired by Schaffer who designed a similar framework for test-

[1] For example, the problem has been reported on cases as diverse as: the detection of oil spills in satellite radar images (Kubat et al., 98), the detection of fraudulent telephone calls (Fawcett and Provost, 97), in-flight helicopter gearbox fault monitoring (Japkowicz et al., 95), information retrieval and filtering (Lewis and Catlett, 94), diagnoses of rare medical conditions such as thyroid diseases (Murphy and Aha, 94)

Workshop on Learning from Imbalanced Datasets II, ICML, Washington DC, 2003.

ing the effect of overfitting avoidance in sparse data sets (Schaffer, 93). Given the relationship between the problem of overfitting the data and dealing with class imbalances (see (Kubat et al., 98)) it seemed reasonable to assume that this framework would apply to our case as well.

The 125 generated domains of our study were generated in the following way: each of the domains is one-dimensional with inputs in the [0, 1] range associated with one of the two classes (1 or 0). The input range is divided into a number of regular intervals (i.e., intervals of the same size), each associated with a different class value. Contiguous intervals have opposite class values and the degree of concept complexity corresponds to the number of alternating intervals present in the domain. Actual training sets are generated from these backbone models by sampling points at random (using a uniform distribution), from each of the intervals. The number of points sampled from each interval depends on the size of the domain as well as on its degree of imbalance. An example of a backbone model is shown in Figure 1.

Five different complexity levels were considered ($c = 1..5$) where each level, c, corresponds to a backbone model composed of 2^c regular intervals. For example, the domains generated at complexity level $c = 1$ are such that every point whose input is in range [0, .5) is associated with a class value of 1, while every point whose input is in range (.5, 1] is associated with a class value of 0; At complexity level $c = 2$, points in intervals [0, .25) and (.5, .75) are associated with class value 1 while those in intervals (.25, .5) and (.75, 1] are associated with class value 0; etc., regardless of the size of the training set and its degree of imbalance.[2]

Five training set sizes were considered ($s = 1..5$) where each size, s, corresponds to a training set of size $round((5000/32) * 2^s)$. Since this training set size includes all the regular intervals in the domain, each regular interval is, in fact, represented by $round(((5000/32) * 2^s)/2^c)$ training points (before the imbalance factor is considered). For example, at a size level of $s = 1$ and at a complexity level of $c = 1$ and before any imbalance is taken into consideration, intervals [0, .5) and (.5, 1] are each represented by 157 examples; If the size is the same, but the complexity level is $c = 2$, then each of intervals [0, .25), (.25, .5), (.5, .75) and (.75, 1] contains 78 training examples; etc.

Finally, five levels of class imbalance were also considered ($i = 1..5$) where each level, i, corresponds to the situation where each sub-interval of class 1 is represented by all the data it is normally entitled to (given c and s), but each sub-interval of class 0 contains only $1/(32/2^i)$th (rounded) of all its normally entitled data. This means that each of the sub-intervals of class 0 are represented by $round((((5000/32) * 2^s)/2^c)/(32/2^i))$ training examples. For example, for $c = 1$, $s = 1$, and $i = 2$, interval [0, .5) is represented by 157 examples and (.5, 1] is represented by 20; If $c = 2$, $s = 1$ and $i = 3$, then [0, .25) and (.5, .75) are each represented by 78 examples while (.25, .5) and (.75, 1] are each represented by 20; etc.[3]

The number of testing points representing each sub-interval was kept fixed (at 50). This means that all domains of complexity level $c = 1$ are tested on 50 positive and 50 negative examples; all domains of complexity level $c = 2$ are tested on 100 positive and 100 negative examples; etc.

3. The Effect of Class Imbalances

This set of experiments attempts to determine whether the class imbalance problem always causes a degradation in performance or whether it does so only in certain cases. In order to answer this question in the case of decision tree induction, we ran C5.0 (Quinlan, 2001) on the 125 domains described in the previous section. The results of our experiments are displayed in Figures 2 and 3 which plots the error C5.0 obtained for each combination of concept complexity, training set size, and imbalance level, on the entire testing set. For each experiment, we reported a single type of result: the *corrected results* in which no matter what degree of class imbalance is present in the training set, the contribution of the false positive error rate is the same as that of the false negative one in the overall report.[4]

[2]In this paper, complexity is varied along a single very simple dimension. Other more sophisticated models could be used in order to obtain finer-grained results. In (Estabrooks, 00), for example, a k-DNF model using several dimensions was used to generate a few artificial domains presenting class imbalances. The study was less systematic than the one in this paper, but it yielded results corroborating those of this section.

[3]Note that throughout the paper, high values of c and s indicate high complexity and large training set sizes, respectively, but that high values of i corresponds to low levels of imbalance (while low values of i correspond to high levels of imbalance).

[4](Japkowicz and Shaju, 2002) lists the results obtained on imbalanced testing sets as well, and, in addition, reports on the false positive and false negative results separately. These results show that the amount of error reported for the large class (class 1) is negligible: most of the error was incurred on the small class (class 0). In addition, the paper lists the results obtained using Neural Networks and Support Vector Machines. For Neural Networks, the class im-

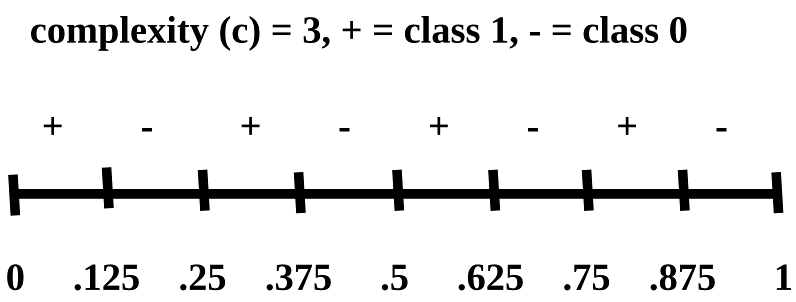
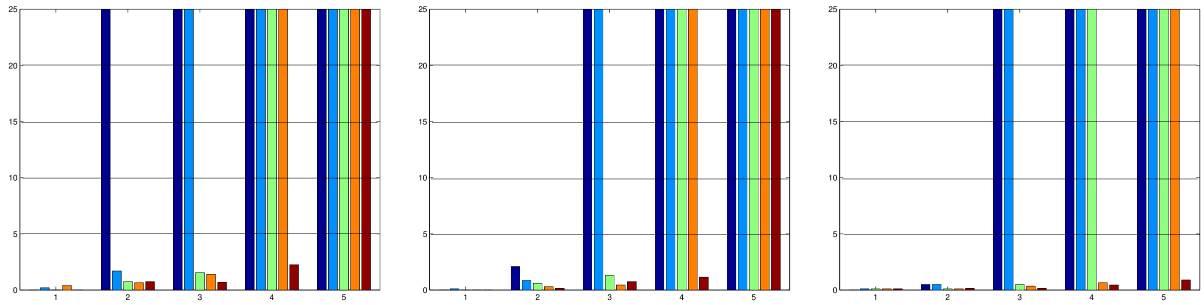
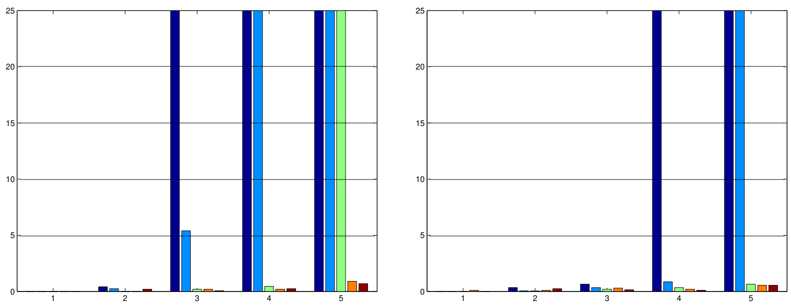


Figure 1. A Backbone Model of Complexity 3



(a) Size=1 **(b) Size=2** **(c) Size=3**

Figure 2. **C5.0 and the Class Imbalance Problem—Corrected**



(d) Size=4 **(e) Size=5**

Figure 3. **C5.0 and the Class Imbalance Problem—Corrected (Cont'd)**

Each plot in each of these figures represents the error obtained at a different training set size. The leftmost plot corresponds to the smallest size ($s = 1$) and progresses until the rightmost plot which corresponds to the largest ($s = 5$). Within each of these plots, each cluster of five bars represent the concept complexity level. The leftmost cluster corresponds to the simplest concept ($c = 1$) and progresses until the rightmost one which corresponds to the most complex ($c = 5$). Within each cluster, finally, each bar corresponds to a particular imbalance level. The leftmost bar corresponds to the most imbalanced level ($i = 1$) and progresses until the rightmost bar which corresponds to the most balanced level ($i = 5$, or no imbalance). The height of each bar represents the average percent error rate obtained by C5.0 (over five runs on different random data sets generated from the same backbone model) on the complexity, class size and imbalance level this bar represents. To make the comparisons easy, horizontal lines were drawn at every 5% marks. If a graph does not display any horizontal line, it is because all the bars represent an average percent error below 5%, and we consider the error negligible in such cases.

Our results reveal several points of interest: first, no matter what the size of the training set is, linearly separable domains (domains of complexity level $c = 1$) do not appear sensitive to any amount of imbalance. As a matter of fact, as the degree of concept complexity increases, so does the system's sensitivity to imbalances. Indeed, we can clearly see in Figure 2 and 3 that as the degree of complexity increases, high error rates are caused by lower and lower degrees of imbalance.

As could be expected, imbalance rates are also a factor in the performance of C5.0 and, perhaps more surprisingly, so is the training set size. Indeed, as the size of the training set increases, the degree of imbalance yielding a large error rate decreases. This suggests that in very large domains, the class imbalance problem may not be a hindrance to a classification system. Specifically, the issue of relative cardinality of the two classes—which is often assumed to be the problem underlying domains with class imbalanced—may in fact be easily overridden by the use of a large enough data set (if, of course, such a data set is available and its size does not prevent the classifier from learning the domain in an acceptable time frame). This question is considered in more detail in the next section.

balance was shown to cause a degradation of performance, though not to the same extent as in the case of decision trees. In the case of Support Vector Machines, no degradation was noted.

All in all, our study suggests that the imbalance problem is a *relative* problem depending on both the complexity of the concept[5] represented by the data in which the imbalance occurs and the overall size of the training set, in addition to the degree of class imbalance present in the data. In other words, a huge class imbalance will not hinder classification of a domain whose concept is very easy to learn nor will we see a problem if the training set is very large. Conversely, a small class imbalance can greatly harm a very small data set or one representing a very complex concept.

Though these results shed some light onto the nature of the class imbalance problem, we now question whether we can refine these conclusions further. In particular, we address an issue that was not fully considered in the experiments just reported: is the class imbalance problem really to blame?

4. Is the Class Imbalance Really to Blame?

In order to answer this question, we reviewed the domain generation process of Section 2. In particular, we looked at the cases where a large amount of performance degradation occurs. As per Figures 2 and 3, The worst degradation occurs when c is high (high concept complexity), i is low (large imbalance) and s is small (small training set size). A careful look at the domain generation process reveals that in these cases, class 0 (the small class) contains a very small number of training examples. For example, for $c = 5$ and $i = s = 1$, there is only 1 example contained in each class 0 interval. In contrast, class 1 contains 156 examples per interval. Even at size $s = 5$, for $c = 5$ and $i = 1$ (when the training set is very large, but the concept very complex and the imbalance very high), class 0's intervals contain only 10 training examples while class 1's intervals contain 156 training examples.

In order to test whether this situation is what causes degradation rather than the class imbalance problem, per se, we decided to modify our domain generation process by guaranteeing a reasonable number of examples per interval of class 0 in all cases, and generating domains of different c and i values. This means that we now completely disregard the value of s and only focus on the values of c and i. In particular, we set the number of training examples in each class 0 interval to 50. Based on these rules, we generate domains containing the following numbers of training examples per interval and per class:

[5]Where "concept complexity" corresponds to the number of subclusters into which the classes are subdivided.

- Number of Training examples per class 1 intervals: $50 \times 2^{5-i}$
- Number of Training examples per class 0 intervals: 50
- Number of intervals in each class: 2^{c-1}

The purpose of these experiments is to draw a contrast between the class imbalance problem and the small hidden disjunct problem. In particular, we want to find out whether it is 1) the class imbalance problem or 2) the fact that small subclusters are very poorly represented in domains of high concept complexity, small size and high imbalance levels that cause a sharp decrease in C5.0's classification accuracy. The results of our experiments are reported in Figure 4.[6]

The results of Figure 4 show that all the errors obtained in our experiments are at or below 1% (note that none of the bars normally drawn at 5% points are visible since all the results are far below 5%). We, thus, consider them negligible. As well, the pattern of error does not follow the pattern we noticed in the previous graphs. This suggests that, it is not, as previously assumed, the presence of a class imbalance, of a high concept complexity or of a small training set, per se, that cause C5.0 a loss of accuracy. Rather, these phenomena cause the subclusters of the small class to be very sparse and it is this condition which, in turn, causes C5.0's decrease in accuracy. This new result is important since it points to the true nature of the class imbalance problem which turns out not to be a problem in itself, but rather to cause a serious condition that is commonly known as the small (hidden) disjunct problem. More specifically, rather than being a problem because of the *relative size* of the large and the small class, the class imbalance problem is only a problem when the size of its small class is very small with respect to the concept complexity; i.e., when it contains *very small subclusters.*

Practically speaking, this means that in very large domains in which there is a good chance that the subclusters of each class are represented by a reasonable number of examples, the class imbalance problem will be of no consequence. Conversely, when the class imbalance problem causes a hindrance, it would be more appropriate to focus on the small hidden disjunct problem than to find ways to rectify or counter the imbalances. The next section summarizes two approaches that we previously proposed to address this problem.

[6]Figure 4 is composed of a single graph because the notion of training set size, s, is meaningless in our new domain generation procedure.

5. Summary of Previous Approaches that deal with the Small Disjunct Problem Directly

We now summarize two approaches for dealing with the small disjunct problem. Both of them rely on an unsupervised learning step that allows us to discover all disjuncts.

5.1. Between-Class versus Within Class Imbalances

In this study (Japkowicz, 2001), (Nickerson et al., 2001), we distinguish between two types of imbalances: *between-class imbalances*, the type of imbalance already considered in Sections 2, 3, and 4 and *within-class imbalances*, the case where a single class is composed of various sub-clusters of different sizes, some of them, being tiny.

The strategy we propose to deal with such cases consists of:

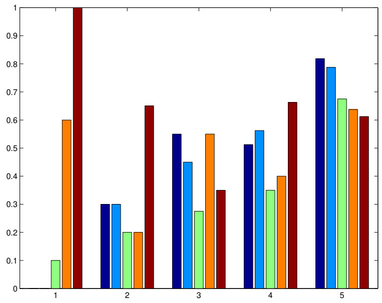
1. Using a clustering algorithm (e.g., k-means) on each class to identify the subclusters that constitute it.
2. Using the following re-sampling strategy:

    > Each subcluster of the large class (class 1) is re-sampled until it reaches the size of the biggest subcluster in that class. At this point, the overall size of the large class will be $maxclasssize$ and there will be no within-class imbalance in the large class. In order to prevent a between-class imbalance as well as within-class imbalances in the smaller class, each subcluster of the small class (class 0) is re-sampled until it reaches size $maxclasssize/N_{smallclass}$, where $N_{smallclass}$ represents the number of subclusters in the small class.

We tested this strategy on a letter recognition task (Japkowicz, 2001) as well as a text classification task (Nickerson et al., 2001) and found that our strategy is helpful as compared to a blind rebalancing strategy, in which the presence of subclusters is not considered.

5.2. Supervised Learning with Unsupervised Output Separation

The second approach (Japkowicz, 2002) does not use any kind of re-sampling, but rather subdivides the two-class problem into a multiclass problem in which each



The Size depends on the values of c and i

Figure 4. **C5.0 and the Class Imbalance Problem without the Small sample problem—Corrected**

subdisjunct represents a class in its own right. Because the most appropriate number of clusters is not clearly defined, we try different numbers and combine the results.[7]

In more detail, the approach we propose can be divided into three steps:

Step 1: We separate each class into a number of subclasses, using an unsupervised learning technique (e.g., k-means), and we re-label each training example as a function of these new subclasses.

Step 2: Supervised learning is applied to various versions of these new problems.

Step 3: The results obtained on each version are combined in a decisive vote.

This approach was evaluated on five different data sets all obtained from the UCI Repository for Machine Learning: Haberman, Ionosphere, Pima, Sonar, Wisconsin Breast Cancer Diagnostic (WBCD).

The results we report suggest that our scheme can be quite useful, although it needs to be refined further. Indeed, on three out of our five data sets, the method is shown improve performance, while in the other two, it either does not help or it hurts the performance.

6. Discussion and Future Work

The results presented in this paper discuss only a small aspect of the problem related to the degradation observed in data presenting a class imbalance. In order to study this question in more depth, several further approaches can be taken. First, it would be interesting to inject various degrees and types of noises in our domains. Given the reported effect of noise on small disjuncts [Weiss, 95], there is a good possibility that our results would too be quite affected by such a variant. Another important consideration has to do with the fact that the different classes of the domain may have different misclassification costs. In this study, we decided to place more emphasis on the small class than on the large one (by using as many testing data in each class), but only because we wanted to measure the ability of the learner to learn each class rather than just to measure predictive accuracy. Least but not last, it is imperative that we conduct experiments on a number of real-world data sets to verify that the hypothesis we posited on simple artificial data sets actually does apply to general and actual data.

Note that the link between small disjuncts and class distribution has also been made recently in a recent work by Weiss (2003).

7. Conclusions

The purpose of this paper was to find out whether the class imbalance problem is a problem in itself or whether the degradation in classification performance that is often reported in the presence of class imbalances could be caused by another phenomenon. Our experiments show that class imbalances do not systematically cause problems but that they do when they yield small disjuncts. We thus suggest that addressing the class imbalance problem is not necesarily the best way to deal with the problem of performance degradation in the presence of class imbalances. Instead, it may be more effective to tackle the small disjunct problem directly. We summarize two approaches that we previously designed in view of this problem as a suggestion for future research in this direction.

[7]Note that in the study of Section 5.1, the number of clusters was also uncertain and we dealt with that problem by simple estimation. The results would probably be improved if, like in the study of Section 5.2, several numbers of clusters were tried and the results combined.

Acknowledgements

This research was supported by an NSERC grant and a grant from the University of Ottawa. We thank Stephen Shaju for his help in running some of the experiments presented in this paper.

Bibliography

Estabrooks, A. (2000), *A Combination Scheme for Inductive Learning from Imbalanced Data Sets*, MCS Thesis, Faculty of Computer Science, Dalhousie University.

Fawcett, T.E. and Provost, F. (1997), "Adaptive Fraud Detection", *Data Mining and Knowledge Discovery*, volume 3, Number 1, pp. 291-316.

Holte, R. C., Acker L. E. and Porter, B. W. (1989), "Concept Learning and the Problem of Small Disjuncts", *Proceedings of the Eleventh Joint International Conference on Artificial Intelligence*, pp. 813-818.

Japkowicz, N., Myers, C. and Gluck, M. (1995), "A Novelty Detection Approach to Classification", *Proceedings of the Fourteenth Joint Conference on Artificial Intelligence*, pp. 518-523.

Japkowicz, N. (2001), "Concept-Learning in the Presence of Between-Class and Within-Class Imbalances", *Advances in Artificial Intelligence: Proceedings of the 14th Conference of the Canadian Society for Computational Studies of Intelligence,* pp. 67-77.

Japkowicz, N. (2002), "Supervised Learning with Unsupervised Output Separation", *IASTED International Conference on Artificial Intelligence and Soft Computing (ASC'2002)*

Japkowicz, N. and Stephen S. (2002), "The Class Imbalance Problem: A Systematic Study", *Intelligent Data Analysis,* Volume 6, Number 5, November 2002.

Kubat M., Holte, R. and Matwin, S. (1998), "Machine Learning for the Detection of Oil Spills in Satellite Radar Images", *Machine Learning*, Volume 30, pp. 195-215.

Murphy, P.M., and Aha, D.W. (1994), *UCI Repository of Machine Learning Databases*, University of California at Irvine, Department of Information and Computer Science.

Lewis, D. and Catlett, J. (1994), "Heterogeneous Uncertainty Sampling for Supervised Learning", *Proceedings of the Eleventh International Conference of Machine Learning*, pp. 148-156.

Nickerson, A., Japkowicz, N. and Milios, E. (2001), "Using Unsupervised Learning to Guide Re-Sampling in Imbalanced Data Sets", *Proceedings of the Eighth International Workshop on AI and Statitsics,* pp. 261-265.

Quinlan, R. (2001), "C5.0: An Informal Tutorial", *http://www.rulequest.com/see5-unix.html.*

Schaffer, C. (1993), "Overfitting Avoidance as Bias", *Machine Learning*, volume 10, pp. 153-178.

Weiss, G. (1995), "Learning with Rare Cases and Small Disjuncts", *Proceedings of the Twelfth International Conference on Machine Learning.*

Weiss, G. (2003), "The Effect of Small Disjuncts and Class Distribution on Decision Tree Learning", *Ph.D. Dissertation, Rutgers University.*