**JIRSS (2003)**
**Vol. 2, No. 2, pp 189-205**

# Scorecard construction with unbalanced class sizes

**Veronica Vinciotti[1], David J. Hand[2]**

[1]Department of Information Systems and Computing, Brunel University, Uxbridge. (Veronica.Vinciotti@brunel.ac.uk)
[2]Department of Mathematics, Imperial College, London.
(d.j.hand@imperial.ac.uk)

**Abstract.** A long-running issue in scorecard construction in retail banking is how to handle dramatically unbalanced class sizes. This is important because, in many applications, the class sizes are very different. We describe the impact ignoring such imbalance can have and review the various strategies which have been proposed for tackling it, embedding them in a common theoretical framework. We then describe a new 'local' method of scorecard construction which both theory and our experiments show yields superior performance to standard methods, while retaining their interpretative simplicity. We illustrate using real banking data sets.

## 1 Introduction

A long-running issue in scorecard construction is the issue of how to handle dramatically unbalanced class sizes. This is important be-

Received: September 2003
*Key words and phrases:* Classification performance, credit scoring, unbalanced classes.

cause, in many applications, the class sizes are very different. For example, it is common to find that 'bad' customers constitute less than 10% of the customer base, and in mass marketing campaigns a response rate of 1% or less is common. Situations which are even more extreme arise in fraud detection (Bolton and Hand, 2002). Brause et al (1999) remark that in their database of credit card transactions 'the probability of fraud is very low (0.2%) and has been lowered in a preprocessing step by a conventional fraud detecting system down to 0.1%,' while Hassibi (2000) comments that 'out of some 12 billion transactions made annually, approximately 10 million - or one out of every 1200 transactions - turn out to be fraudulent. Also, 0.04% (4 out of every 10,000) of all monthly active accounts are fraudulent.' A common problem with such situations is that, as we explain below, often the minimum number of misclassifications is achieved simply by assigning everyone to the larger class. Thus, for example, if 0.1% of a set of transactions are fraudulent, then if all transactions are taken as legitimate then only 0.1% of the transactions will be misclassified. Such a course of action is seldom acceptable.

For simplicity, in this paper we will restrict ourselves to the situation in which the aim is to predict which of just two classes the customer lies (or will lie) in. We assume that we have available a retrospective data set, consisting of information about a set of customers (e.g., application form data, data on past behaviour with a financial product, or bureau data), and that for these customers we know into which of the two classes they eventually fell. Using this information, we aim to construct a scorecard which will allow us to predict the class of a new customer using the descriptive information about that customer. The action we will take with a new customer will depend upon which class we predict them to lie in. Examples of different action pairs are (grant loan, do not grant loan) and (treat as normal, send warning letter). In this paper we restrict ourselves to action pairs for which the true classes of all customers eventually become known.

Let $x$ represent the information describing a customer, and $s = s(x)$ the customer's score on some scorecard. This score represents a position on a continuum, for which, without loss of generality, we will take high scores as generally being associated with class 1 and low scores as generally being associated with class 0. The score, $s$, is thus (monotonically increasingly) related to $\hat{p}(1|x)$, an estimate of the probability $p(1|x)$ that someone with characteristic vector $x$ will

belong to class 1. Also, at this stage, we should note that we are assuming that the data (either entire populations of customers, or subsamples from these populations) are drawn in such a way that the proportion of customers which belong to each class in the dataset are unbiased estimates of the probability of belonging to that class. That is, we assume that the class proportions in the available data are, subject to sampling variability, equal to the prior probabilities in the population to which the scorecard will be applied. Later we will consider sampling from the available data in a way which will distort these proportions.

Given the estimate $\hat{p}(1|x)$, an obvious and popular strategy to assign customers to classes is to assign them to the class to which they are estimated as being most likely to belong. That is, one could adopt:

**Rule 1.** Assign to class 1 if $\hat{p}(1|x) > 0.5$ and to class 0 otherwise.

However, suppose that class 0 is much larger than class 1 - that the classes are *unbalanced.* Then it can easily happen that $p(1|x) \ll 0.5$ and $\hat{p}(1|x) \ll 0.5$ *for all vectors* $x$. Using Rule 1 would mean that no customers would be assigned to class 1. The 'classification threshold' 0.5 will minimise the overall number of customers misclassified (the number from class 0 assigned to class 1 plus the reverse) but may do this by the simple expedient of assigning all of the smaller class (class 1) to the larger class. If the smaller class represents potential frauds or potential bad risk customers, this is not at all what we want. The straightforward overall number of customers misclassified is thus an inappropriate measure of performance, so that Rule 1, which minimises this measure, is an inappropriate classification rule. As it happens, many other performance measures are used in retail banking, but most of them are also inappropriate (Hand, 2002).

The problem with Rule 1 arises from the failure to recognise that different types of misclassification carry different penalties. For example, assigning a fraudulent customer to the non-fraud class is more serious than the reverse: we wish to avoid this if at all possible, even if it means that we might misclassify some non-fraudulent customers as potentially fraudulent (and take some action, such as phoning them to see if their credit card has been stolen). If we suppose that misallocating a class 1 customer to class 0 is $r$ times as serious as the reverse, and weight such misclassifications $r$ times as heavily as the class-0-to-class-1 misclassifications (and also, for simplicity, assume

that correct classifications incur no cost), then it is easy to show that comparing $\hat{p}(1|x)$ with classification threshold $(1+r)^{-1}$ minimises the overall weighted number of customers misclassified. That is, the overall weighted misclassification rate is minimised by

**Rule 2.** Assign to class 1 if $\hat{p}(1|x) > (1+r)^{-1}$ and to class 0 otherwise.

The term $(1+r)^{-1}$ will be very different from the 0.5 used above if $r$ is large.

Four broad strategies have been developed for implementing Rule 2.

- Introduce misclassification costs at the *classification* stage, so that misclassifications of the smaller class are explicitly regarded as more serious than the reverse. This simply adopts rules of the form of Rule 2 directly, choosing the threshold $(1+r)^{-1}$ appropriately.

- Ignore the lack of balance, and use performance assessment measures which focus on the separability between the distributions of the estimates $\hat{p}(1|x)$ for customers from the two classes. While this can be used for choosing which scorecards are likely to be effective, one still needs to choose a threshold in order to make actual classifications.

- Preprocess the data to adjust the class sizes, either by subsampling from the larger class or by oversampling from the smaller class. Thus, for example, by applying Rule 1 to a data set in which the larger class has been reduced in size one can achieve the same results as applying Rule 2 to the unmodified data (apart from differences arising from random variation, of course).

- Introduce misclassification costs at the scorecard *construction* stage, so that again misclassifications of the smaller class are explicitly regarded as more serious than the reverse, but, in contrast to strategy 1, this information is not used merely for the choice of threshold.

We review each of these in turn, and draw some conclusions about their relative merits. We then describe and illustrate a method for implementing strategy 4 which has the practical merits of the popular

logistic regression approach, but which leads to improved scorecard performance.

## 2 Strategies for unbalanced classes

### 2.1 Strategy 1: Using costs at the classification stage

In the introductory section, we described the most obvious approach to handling unbalanced classes. This is a two stage approach. Stage 1 involves estimating $\hat{p}(1|x)$ using all of the available data. For example, linear and logistic regression and tree classifiers are particularly common, but other methods include neural networks, tree classifiers, and so on. Linear and logistic regression methods have the property that the resulting scores are simply weighted sums of the raw customer characteristics (perhaps partitioned, or combined in some way - see Hand and Adams (2000), for an example), which is often desirable in consumer credit applications. Stage 2 involves comparing the estimate $\hat{p}(1|x)$ with a threshold. Ideally, as noted above, the threshold will reflect some measure of the relative severity of the two kinds of misclassification.

Many authors have followed this two stage procedure. It is perhaps the most natural, from a traditional perspective, which often regards such problems as two stage processes: estimate the distributions involved (model building) and then make the classification (decision making). However, despite the simplicity and popularity of this approach, it is often not ideal. In particular, if the estimate $\hat{p}(1|x)$ is misspecified in some way (for example, it is based on a parametric form which does not properly reflect the true distributions), then it may lead to a decision surface which is a poor approximation to that for the desired threshold. We discuss the implications of this, and how to avoid it, below. For now, however, a simple example will illustrate.

In standard linear discriminant analysis the assumed common covariance is estimated as a weighted average of estimates of the two within-class covariance matrices. The weights are normally taken to be the observed class sizes in the data. If one class is much larger than the other, then the estimate will be biased towards that class. In linear discriminant analysis, the decision surface is taken to be linear, and such surfaces will have the same orientation for all classification thresholds. This orientation is a function of the vector

difference between the sample centroids of the two classes and of the assumed common covariance matrix. If the estimate of the latter is determined essentially by the larger class, then it may lead to a suboptimal decision surface for unbalanced problems.

## 2.2 Strategy 2: Separability criteria based on within class score distributions

Measures such as the Gini coefficient (equivalently, the area under the ROC or Lorenz curve, or the test statistic used in the Mann-Whitney-Wilcoxon test to compare two distributions), the information value, the standardised mean score difference, and the Kolmogorov-Smirnov test statistic are popular criteria for choosing between scorecards. As described in detail in Hand (2002), they have the particular merit of ignoring the classification threshold (the 0.5 and the $(1+r)^{-1}$ in rules 1 and 2), which is especially useful when the threshold is not yet known or may vary. On the other hand, this merit is also the disadvantage of such methods (Adams and Hand, 1999; Hand, 2002), since such a threshold *must* be chosen in order to make a classification. Criteria which average, in some sense, over all possible values may lead to the choice of a rule which is globally optimal in this average sense, but which in fact performs poorly for the actual situation one is faced with.

Despite this serious shortcoming, the use of such global separability measures is widespread: every retail bank and credit rating agency uses such measures. An example for unbalanced data is given in Ling and Li (1998), who study a marketing application in which as little as 1% of the population responds to a promotion. If those likely to respond can be identified *a priori*, then a more targeted and hence cost-effective promotions strategy can be adopted. If the scores, s, are categorised into groups, $s_i$, $i = 1, \ldots, g$, then Ling and Li (1998) use a measure of separability equivalent to a weighted sum of the estimated probability of being in class 1 in each of the groups: $\sum_i w_i p(1|s_i)$.

## 2.3 Strategy 3: Preprocessing the data

Many studies adopt the strategy of preprocessing the data to (roughly) equalise the numbers of elements in the two classes - to achieve better balance. Kubat and Matwin (1997), for example, preprocessed the data by removing unnecessary instances from the majority class. Iso-

lated points from the majority class in regions dense with points from the other class, and examples which are redundant in the sense that their removal does not affect the decision surface, or those that are close to the decision boundary can all be considered as candidates for removal. The ideas parallel those developed some two decades previously, in attempts to speed up the processing time of nearest neighbour classification rules (see, for example, Hart, 1968; Gates, 1972; Hand and Batchelor, 1978).

An et al (2001) adopted the opposite approach. Rather than subsampling the larger class, they experimented with duplicating the elements of the smaller class (so that this class comprised from 4% to 50% of the training data). Lee (1999, 2000), also duplicated elements of the smaller class, but added small random perturbations to the replicated points.

Chan and Stolfo (1998) studied a credit card fraud data set, which had about 20% in the smaller (fraudulent) class. This is exceptionally large for the proportion of fraudulent customers in a data set, and arises because a pre-processing stage has been applied which eliminated many of those thought very unlikely to be fraudulent. Chan and Stolfo tackled the lack of balance by randomly partitioning the larger set into four non-overlapping samples, and combining each of these with the smaller set, to yield four smaller data sets with equal numbers from each class. The four resulting classification rules were merged to yield a meta-classifier. This might not be a very effective strategy when the imbalance is marked, or if very few points are available from the smaller class.

Many of the subsampling or oversampling procedures are rather ad hoc. Elkan (2001) describes what sampling fractions are appropriate for given cost ratios.

Rule 2 can be alternatively written as: assign a customer with characteristic vector $x$ to class 1 if

$$\hat{p}(x|1)\hat{p}(1)/\hat{p}(x|0)\hat{p}(0) > 1/r \tag{1}$$

and to class 0 otherwise, where $\hat{p}(x|k)$ is the estimated probability that a customer from class $k$ will have characteristic vector $x$, and $\hat{p}(k)$ is the estimated overall probability of belonging to class $k$.

(1) is equivalent to

$$\hat{p}(x|1)/\hat{p}(x|0) > \hat{p}(0)/r\hat{p}(1) \tag{2}$$

Ideally, subsampling would reduce the class 0 size by a factor of $1/r$, so that the new class priors become $\pi(k),\ k = 0, 1$, estimated by $\hat{\pi}(k)$,

the proportion of the (sampled) data set which belong to class $k$. An easy calculation shows that then $\pi(0) = \frac{p(0)}{(p(0)+rp(1))}$. Expression (2) then becomes

$$\hat{p}(x|1)/\hat{p}(x|0) > \hat{\pi}(0)/\hat{\pi}(1) \tag{3}$$

so that the classification rule is simply: assign a customer with characteristic vector $x$ to class 1 if

$$\frac{\hat{p}(x|1)\hat{\pi}(1)}{\hat{p}(x|0)\hat{\pi}(0)} = \frac{\hat{\hat{p}}(1|x)}{\hat{\hat{p}}(0|x)} > 1 \tag{4}$$

and to class 0 otherwise, where the $\hat{\hat{p}}(k|x)$ are based on the sampled data. This is, of course, equivalent to Rule 1, but using the $\hat{\hat{p}}(k|x)$ in place of the raw (unsampled) data, so that an appropriate threshold is used.

A similar derivation applies if class 1 is oversampled, rather than class 0 subsampled.

The derivation of (4) assumed that the sampling fraction was $1/r$, this being the fraction which is appropriate to balance the relative severities of the two kinds of misclassification. If a different sampling fraction is used then a poor rule could result. For example, many authors simply try roughly to equalise the sizes of the two classes. This confounds differences between class sizes with differences between the relative severities of the two kinds of misclassification. The two need have no relationship at all.

The sampling approach (assuming the correct sampling fraction is used) has the merit that it focuses attention on the correct decision surface. That is, it is equivalent to using the optimal threshold $(1 + r)^{-1}$ of Rule 2, rather than the inappropriate threshold 0.5 of Rule 1. However, one might have doubts about the subsampling strategy, on the grounds that it sacrifices information. Likewise, the oversampling strategy either fails to model the variation of the smaller class properly (if the data from this class are simply replicated) or attempts to model this variation, but in a way which is not proven to be correct (by perturbing the replicates). Strategy 4, below, sidesteps these problems.

### 2.4 Strategy 4: Using costs when building the scorecard

Strategy 1 is based on the assumption that the relative misclassification severities, equivalently the particular threshold to use in the

classification rule, should not affect the estimate $\hat{p}(1|x)$. This is a reasonable assumption if one believes that the model form underlying the estimate $\hat{p}(1|x)$ is sufficiently flexible to include the true distributions. For example, if one believes that the contours of the function $p(1|x)$ really are linear in the raw characteristics, then linear and logistic regression models are appropriate to consider. Of course, the fact is that one will seldom have such confidence, although one might believe that a given parametric model form provides a reasonable approximation. Nonparametric approaches, such as kernel and nearest neighbour methods, do not restrict the model (indeed, subject to certain regularity conditions, they can be shown to be able to model any distribution, at least asymptotically). However, in the credit scoring context, there is a premium on simplicity, leading to an emphasis on simple parametric models (see Adams and Hand, 1999).

In these circumstances, model fitting procedures typically aggregate the quality of fit of the model over the entire data space. For example, least squares regression is based on a criterion which combines the sum of squared residuals from all data points. Likewise, maximum likelihood methods combine the contribution to the likelihood from all the observed data points. Such aggregation will yield a model which is the best overall model, where the meaning of 'best' depends on the particular criterion chosen - sum of squares, likelihood, etc. However, since they do aggregate over all data points - over the entire data space - they combine the accuracy of the model in the particular regions of interest (those given by the threshold) with all other regions (those far from the threshold). In particular, it is entirely possible that the fit in the region of interest is not very good, even though the overall average fit is the best that can be achieved. It means that a better local model, *in the region which matters*, might be possible. Put another way, it means that the relative severities of the two kinds of misclassification should be taken into account when the model is *constructed* and not merely at the classification stage. These ideas are described in more detail in Hand and Vinciotti (2002) and are illustrated below.

More generally than the particular model we have developed, several authors have explored the use of relative misclassification penalties when constructing classifiers, including Pazzani et al (1994), Turney (1995), Cardie and Howe (1997), Bradford et al (1998), Fan et al(1999), Domingos (1999), Veropoulos et al (1999), Wan et al (1999), and Ting (2002).

## 3 Local scorecard models

Hand and Vinciotti (2002) give an example in which the contours of $p(1|x)$ are linear, but not parallel (the support of $p(x)$ is specified as zero in regions where different contours may cross, so that there are no conflicts). Logistic regression, however, assumes parallel linear contours. In effect, such a model 'averages' the non-parallel contours of the example over the entire data space. If, by accident, the particular contour corresponding to the threshold is parallel to this 'average' contour, then the model will yield good predictions. On the other hand, if the contour of interest is not parallel to this 'average' contour then the predictions could be poor. In the case when one of the classes is very small, the data points from this class may lie in a relatively small region of the data space. If this happens, the aggregation process when a global parametric model is fitted will yield a model which has greatest accuracy in the vicinity of the data points from the smaller class. This will generally not correspond to threshold values which weight the relative misclassifications appropriately, so that the effect will be more marked in unbalanced situations.

Given that the problem with the standard modelling approach is that they aggregate over all data points, 'averaging' over all the different contours, then one can ease the problem by focusing attention around the contour which matters. Data far from this contour are at best irrelevant, and at worst misleading, leading to a poor estimate, and should not influence the goodness of fit to a great extent. Thus one might weight the data so that points close to the relevant contour are weighted more heavily. The problem is, of course, that one cannot identify which data should be heavily weighted because one does not know the position of the contour. Thus suggests an iterative, or at least several-stage process, in which one uses an estimate of the relevant contour to provide information on the weights, which in turn leads to an improved estimate, and so on. Hand and Vinciotti (2002) describe such an approach using a modified likelihood function, and we use this method in the examples below, using real credit data sets. The idea is related to boosting, but our approach retains a simple model form.

The appropriate performance measure to use here is the overall misclassification cost, $n_0 + rn_1$, where $n_k,\ k = 0, 1$ is the number of customers from class $k$ which are misclassified. For given values of $r$, this measure was calculated for both the standard (global) logistic regression model and the local logistic model described above. As

$r$ varies (as one chooses different relative misclassification costs) so, of course, different contours become the most important contour. Thus, for both global and local models, a threshold will be chosen to match the costs: this threshold will be $(1 + r)^{-1}$. For global logistic models the same probability estimate $\hat{p}(1|x)$ will be used for all costs. In contrast, for the local model different estimates will be used - estimates which are tuned to the cost. To compare the two models, we used the difference between the global and local costs. A positive value of the (global-local) difference means that the global model has greater cost - that the local model yields superior cost weighted classifications.

Figures 1 to 3 show the values of global-local cost for three examples. Figures 2 and 3 were based on data sampled from a larger data set. We repeated this sampling 10 times, to provide the 95% confidence intervals shown in these figures. As can be seen, although not always superior, the local model generally outperforms the global model.

**Example 3.1.** These data were supplied by a major UK bank. They consist of 21618 unsecured personal loans with a 24-month term, collected over the two year period January 1995 to December 1996. An account is defined as bad if it is at least three months in arrears. With this definition, 11% of customers turn out to be bad. 16 variables describe the application for the loan.

**Example 3.2.** These data were supplied by a major UK credit card company. The aim of the analysis was to predict the future behaviour of a customer based on their previous behaviour. There were 782 observations on 8 variables, with 10% of the data in class 1.

**Example 3.3.** These data were supplied by a major UK bank. They describe customers who have defaulted on a loan in some well-defined sense and from whom the bank is trying to recover the loan. A bad account is defined as one that has spent more than a month in this "collections" state. The data consists of 6892 observations on 11 variables, with 10% being in class 1.

**Figure 1: Global-local costs for Example 1.**

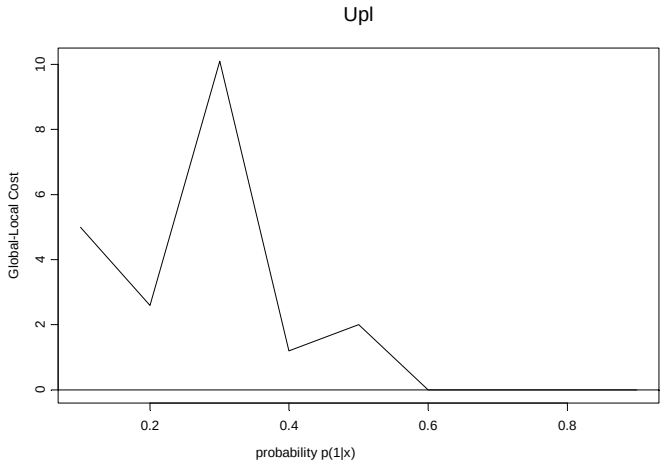



**Figure 2: Global-local costs for Example 2.**

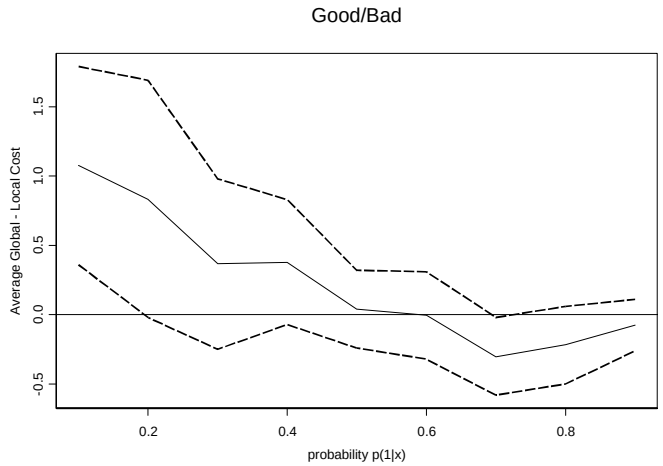
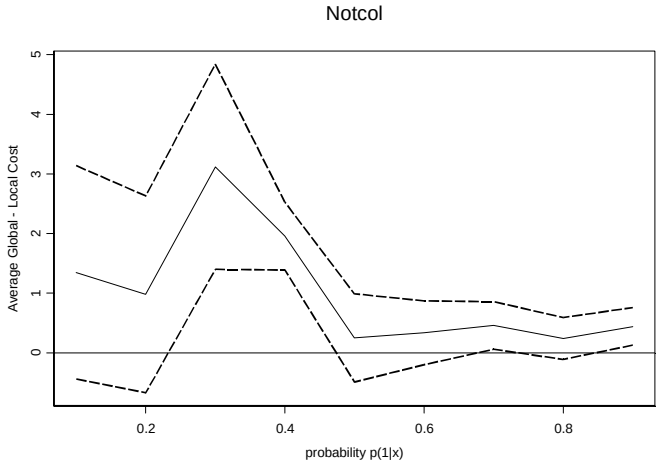


**Figure 3: Global-local costs for Example 3.**




# 4 Conclusion

Unbalanced data sets, in which one of the (two) classes is much larger than the other, are common in retail banking applications, and several strategies have been proposed for building scorecards using such data sets. The most straightforward approach is simply to adjust the threshold with which the estimated class membership probability $\hat{p}(1|x)$ is compared. However, credit scoring applications often place a premium on interpretative simplicity, yielding estimates which may be good for some $x$ and poor for others.

Another class of methods is based on selectively sampling from the two classes, either to reduce the size of the larger class or to increase the size of the smaller. Often the sampling fraction is taken to be such as to yield approximately equal class sizes. This, however, is unlikely to be the optimal sampling fraction. If this method is adopted, then sampling should be such as to yield a class size ratio determined by the relative costs of misclassification for customers from the two classes. The sampling approach is equivalent to adjusting the relative misclassification costs of customers from the two classes.

Even if an optimal sampling fraction is chosen, the sampling methods leave one with the suspicion that something better could be done.

After all, subsampling appears to discard information, while oversampling either ignores natural variability or artificially introduces it. In any case, just as with the simple method based on adjusting the threshold, sampling methods are global. They aggregate information from the entire data set and do not concentrate attention where it matters.

The final strategy is to take account of the misclassification costs - of which contour matters - when the probability estimates $\hat{p}(1|x)$ are made. In particular, we describe such an approach which is based on logistic regression, and so preserves the simple linear form of such models. This method then concentrates estimation power in the region of this contour, so that irrelevant contours do not influence the estimate. This strategy is appropriate whether or not the classes are unbalanced, though it may be particularly pertinent in the unbalanced case. Our empirical investigations show that this method generally improves on straightforward logistic regression.

## Acknowledgements

The work of Veronica Vinciotti on this project was supported in part by a research grant from Fair, Isaac.

## References

Adams, N. M. and Hand, D. J. (1999), Comparing classifiers when the misallocation costs are uncertain. Pattern Recognition, **32**, 1139-1147.

An, A., Cercone, N., and Huang, X. (2001), A case study for learning from imbalanced data sets. Advances in Artificial Intelligence Proceedings of the 14th Conference of the Canadian Society for Computational Studies of Intelligence, 1-15, Springer, Ottawa, Canada.

Bolton, R. J. and Hand, D. J. (2002), Statistical fraud detection: a review. To appear in Statistical Science.

Bradford, J., Kunz, C., Kohavi, R., Brunk, C., and Brodley, C. E. (1998), Pruning decision trees with misclassification costs. Pro-

ceedings of the Tenth European Conference on Machine Learning, 131-136, Springer, Chemnitz, Germany.

Brause, R., Langsdorf, T. and Hepp, M. (1999), Neural data mining for credit card fraud detection. Proceedings 11th IEEE International Conference on Tools with Artificial Intelligence, 103-106, IEEE Press, Toulouse.

Cardie, C. and Howe, N. (1997), Improving minority class prediction using case-specific feature weights. Proceedings of the Fourteenth International Conference on Machine Learning, 57-65, Morgan Kaufmann, San Mateo, CA.

Chan, P. K. and Stolfo, S. J. (1998), Towards scalable learning with non-uniform class and cost distributions: a case study in credit card fraud detection. Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, 164-168, AAAI Press, New York.

Domingos, P. (1999), Metacost: a general method for making classifiers cost sensitive. Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining, 155-164, ACM Press, San Diego, CA.

Elkan, C. (2001), The foundations of Cost-Sensitive Learning. Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, 973-978, Morgan Kaufmann, Seattle.

Fan, W., Stolfo, S. J., Zhnag, J., and Chan, P. K. (1999), AdaCost: misclassification cost-sensitive boosting. Proceedings of the Sixteenth International Conference on Machine Learning, 99-105, Morgan Kaufmann, Bled, Slovenia.

Gates, G. W. (1972), The reduced nearest neighbour rule. IEEE Transactions on Information Theory, **18**, 431.

Hand, D. J. (2002), Measuring scorecard performance in retail credit applications. Tech. Rep., Department of Mathematics, Imperial College, London.

Hand, D. J. and Adams, N. M. (2000), Defining attributes for scorecard construction. Journal of Applied Statistics, **27**, 527-540.

Hand, D. J. and Batchelor, B. G. (1978), An edited condensed nearest neighbour rule. Information Sciences, **14**, 171-180.

Hand, D. J. and Vinciotti, V. (2003), Local versus global models for classification problems: fitting models where it matters. The American Statistician, **57**, 124-131.

Hart, P. E. (1968), The condensed nearest neighbour rule. IEEE Transactions on Information Theory, **14**, 515-516.

Hassibi, K. (2000), Detecting payment card fraud with neural networks. Business Applications of Neural Networks. P.J.G. Lisboa, A.Vellido, B.Edisbury Eds. Singapore: World Scientific.

Kubat, M. and Matwin, S. (1997), Addressing the curse of imbalanced data sets: one-sided sampling. Proceedings of the Fourteenth International Conference on Machine Learning, 179-186, Morgan Kaufmann, San Mateo, CA.

Lee, S. S. (1999), Regularization in skewed binary classification. Computational Statistics, **14**, 277-292.

Lee, S. S. (2000), Noisy replication in skewed binary classification. Computational Statistics and Data Analysis, **34**, 165-191.

Ling, C. X. and Li, C. (1998), Data Mining for Direct Marketing: Problems and Solutions. Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, 73-79, AAAI Press, New York.

Pazzani, M., Merz, C., Murphy, P., Ali, K., Hume, T., and Brunk, C. (1994), Reducing misclassification costs. Proceedings of the Eleventh International Conference on Machine Learning, 217-225. Morgan Kaufmann, New Brunswick, NJ.

Ting, K. M. (2002), Cost-sensitive classification using decision trees, boosting, and MetaCost. In Heuristic and Optimization for Knowledge Discovery, eds. Sarker R., Abbass H. and Newton C., Idea Group Publishing, USA.

Turney, P. D. (1995), Cost-sensitive classification: empirical evaluation of a hybrid genetic decision tree induction algorithm. Journal of Artificial Intelligence Research, **2**, 369-409.

Veropoulos, K., Campbell, C., and Cristianini, N. (1999), Controlling the sensitivity of support vector machines. Proceedings of

the Sixteenth International Joint Conference on Artificial Intelligence. Workshop on Support Vector Machines, 55-60, Morgan Kaufmann, Stockholm, Sweden.

Wan, C., Wang, L. and Ting, K. M. (1999), Introducing cost-sensitive neural networks. Proceedings of the Second International Conference on Information, Communications and Signal Processing, 1-4, IEEE, Singapore.