Statistical Methods & Applications
https://doi.org/10.1007/s10260-021-00573-7

ORIGINAL PAPER

robROSE: A robust approach for dealing with imbalanced data in fraud detection

Bart Baesens[1] · Sebastiaan Höppner[2] · Irene Ortner[3] · Tim Verdonck[4]

Accepted: 24 May 2021

Abstract

A major challenge when trying to detect fraud is that the fraudulent activities form a minority class which make up a very small proportion of the data set. In most data sets, fraud occurs in typically less than 0.5% of the cases. Detecting fraud in such a highly imbalanced data set typically leads to predictions that favor the majority group, causing fraud to remain undetected. We discuss some popular oversampling techniques that solve the problem of imbalanced data by creating synthetic samples that mimic the minority class. A frequent problem when analyzing real data is the presence of anomalies or outliers. When such atypical observations are present in the data, most oversampling techniques are prone to create synthetic samples that distort the detection algorithm and spoil the resulting analysis. A useful tool for anomaly detection is robust statistics, which aims to find the outliers by first fitting the majority of the data and then flagging data observations that deviate from it. In this paper, we present a robust version of ROSE, called robROSE, which combines several promising approaches to cope simultaneously with the problem of imbalanced data and the presence of outliers. The proposed method achieves to enhance the presence of the fraud cases while ignoring anomalies. The good performance of our new sampling technique is illustrated on simulated and real data sets and it is shown that robROSE can provide better insight in the structure of the data. The source code of the robROSE algorithm is made freely available.

Keywords Fraud analysis · Skewed data · Outliers · Oversampling · Binary classification

1 Introduction

The Association of Certified Fraud Examiners (ACFE), estimates that a typical organization loses 5% of its revenues to fraud each year. The Nilson Report, a publication covering news and analysis of the global payment industry, reported that

Extended author information available on the last page of the article

Published online: 07 June 2021

global card fraud losses equaled 22.8 billion in 2016, which is an increase of 4.4 percent over 2015. According to the European Insurance Committee, fraud takes up $5 - 10\%$ of the claim amounts paid for non-life insurance and the FBI estimates that the total cost of insurance fraud (non-health insurance) in the US is more than $40 billion per year. The national audit, tax and advisory firm Crowe Clark Whitehill, together with the University of Portsmouth's Centre for Counter Fraud Studies (CCFS), estimate that the fraud epidemic costs the UK 110 billion pounds a year. According to their report, businesses lose an average of 6.8% of total expenditure and it is concluded that fraud is the last great unreduced business cost. Also, in the public sector fraudulent spending is a massive issue. Popular examples are tax evasion fraud, VAT carrousel fraud, counterfeit, bribery, corruption and social security fraud. As reported by the European Court of Auditors, on *Fighting fraud in EU spending: action needed*, the European Commission does not have an estimation on undetected fraud, but in their PIF Report 2017 they communicate, that 0.29% of EU spending was fraudulent spending, which amounts to €390.7 million for detected fraud only. These are just a few numbers to indicate the severity of the fraud problem. It is also seen that losses due to fraudulent activities keep increasing each year and affect organizations worldwide. Therefore, fraud detection and prevention is more important than ever before and developing powerful fraud detection systems is of crucial importance in order to reduce losses.

The Oxford Dictionary defines fraud as "wrongful or criminal deception intended to result in financial or personal gain". This definition captures the essence of fraud but it does not very precisely describe the nature and characteristics of fraud. A more thorough and detailed characterization of the multifaceted phenomenon of fraud is provided by Van Vlasselaer et al. (2016): "fraud is an uncommon, well-considered, imperceptibly concealed, time-evolving and often carefully organized crime which appears in many types of forms". This definition highlights five characteristics that are associated with particular challenges related to developing a fraud detection system and as such also describes the requirements of a successful fraud detection system.

In this paper we will focus on the first emphasized characteristic and associated challenge of this detailed definition, namely the fact that fraud is uncommon or rare. For example, in a credit card fraud setting, typically less than 0.5% of transactions are fraudulent. Such a problem is commonly referred to as the needle in a haystack problem. Independent of the exact setting or application, only a minority of the involved population of cases typically concerns fraud, of which furthermore only a limited number will be known to concern fraud. Highly imbalanced or skewed data make it difficult to detect fraud, since the fraudulent cases are covered by the non-fraudulent ones, and to learn from historical cases to build a powerful fraud detection system since only few examples are available.

A stream of literature has reported upon the adoption of data-driven approaches for developing fraud detection systems, see for example (Phua et al. 2010; Ngai et al. 2011). These data-driven methods have three important benefits towards an expert-based approach: they significantly improve the efficiency of fraud detection systems, they are more objective and they are easier to maintain. From a machine learning perspective, the task of detecting fraudulent transactions is a binary classification problem. Popular data-driven techniques for fraud detection are logistic regression and/or decision trees. These methods are so-called white box models and therefore yield a clear explanation behind how they reach their classification. These models hence enable the user to understand the underlying reasons why the model signals an observation to be suspicious. Besides their interpretability, they are also operationally efficient. To increase the detection power, these simple analytical models can be extended by adding a penalty or regularization term or using the idea of ensemble learning. Alternative complex techniques are neural networks and support vector machines. Although the latter are definitely power analytical techniques, they suffer from a very important drawback which is not desirable from a fraud detection perspective: they are black box models which means that they are very complex to interpret.

It is not our aim to give a detailed overview of the various machine learning techniques that could be applied for fraud detection. Instead, we focus on the imbalance or skewness of the data, meaning that typically there are plenty of historical examples of non-fraudulent cases, but only a limited number of fraudulent cases. This problem typically causes an analytical technique to experience difficulties in learning to create an accurate model. Every classifier faced with a skewed data set typically tends to favor the majority class. In other words, the classifier tends to label all observations as non-fraudulent since it then already achieves a classification accuracy of more than 99%. Classifiers typically learn better from a more balanced distribution. Two ways to accomplish this is by random undersampling, whereby non-fraudulent transactions in the training set are removed, or random oversampling, whereby fraudulent transactions in the training set are replicated. Better results are obtained when applying synthetic oversampling, which oversample the minority class by creating synthetic examples to improve the performance of the fraud detection model. Synthetic Minority Oversampling technique (SMOTE) is the first and probably the most well-known synthetic oversampling technique to deal with skewed data (Chawla et al. 2002). Since then many variants and alternatives for SMOTE have been presented in the academic literature. In this paper we focus on ROSE (Menardi and Torelli 2014), which generates new minority samples based on the kernel density estimate around existing, real minority cases. When outliers or anomalies are present in the data, ROSE unfortunately also creates synthetic examples based on these outliers. This may distort the detection algorithm and actually lower the performance. Therefore, we introduce a robust version of the ROSE algorithm, which is not sensitive to the presence of outliers. Synthetic examples are then only generated from observations which are considered clean or normal observations. Moreover, our robust version also takes the covariance structure of the data into consideration to create more realistic artificial observations.

The remainder of the article is organized as follows. In Section 2, we will present SMOTE and other popular sampling techniques to solve the class imbalance problem. Section 3 describes performance measures for classification that are suited to evaluate a fraud detection model. In section 4 our robust oversampling technique for dealing with imbalanced data in fraud detection is proposed. We illustrate its good performance on simulated data in Section 5, whereas Section 6 analyzes a credit card transaction dataset for fraud. Section 7 shows that the proposed methodology can also be used in other domains than fraud detection and Section 8 concludes.

2 Selection of popular sampling techniques

The literature review given by He and Garcia (2009) shows the large extent to which techniques for handling imbalanced learning problems are researched. The solutions to imbalanced data sets can be divided into four categories: sampling-based methods, cost-based methods, kernel-based methods and active learning-based methods (He and Garcia 2009). In this paper, we are interested in sampling methods and provide a brief overview of the works performed in this category. A summary of the work performed in the other categories can be found in He and Garcia (2009). For more information about cost-sensitive approaches in the context of fraud detection, we refer to Hand et al. (2008); Bahnsen et al. (2013). Zhu et al. (2019) recently implemented techniques for imbalanced classification in an R library called IRIC.

Sampling methods operate at the data level as they change the distribution between the majority class and the minority class samples of the imbalanced data set. The balanced data set is then provided to the classification algorithm which typically learns better from a balanced distribution than from an imbalanced one and so the detection rate of minority cases is improved (Weiss and Provost 2001). Balancing the distribution can be done by either reducing the majority class samples or by adding minority class samples. The former is called undersampling and the latter is called oversampling. Random undersampling is the simplest form of undersampling as it randomly takes away samples from the majority class, while informed undersampling uses some statistical knowledge to remove majority samples (Liu et al. 2008). A form of kernel-based random undersampling for unsupervised classification methods is discussed in an anti-fraud context by Cerioli and Perrotta (2014).

Our focus, however, lies in oversampling methods. Random oversampling simply duplicates samples from the minority class which could lead to very specific rules and hence overfitting (Holte et al. 1989). Synthetic oversampling, on the other hand, adds new information to the original data set by generating synthetic minority class samples to improve the performance of the classifier. Various synthetic oversampling methods exist in the literature such as Synthetic Minority Oversampling TEchnique (SMOTE) (Chawla et al. 2002), Borderline-SMOTE (Han et al. 2005), Adaptive Synthetic Sampling Technique (ADASYN) (He et al. 2008), MWMOTE (Barua et al. 2012) and ROSE (Menardi and Torelli 2014). The ways in which these oversampling methods generate synthetic minority samples can be described based on how they answer the following three questions.

1. Which minority samples do we want to oversample? All of them or are there minority outcasts (i.e. “anomalous minorities”) which we exclude from the process?
2. By how much do we want to oversample the minority cases? Should we oversample some minority cases more than others?

3. How should we oversample or, in other words, how do we create synthetic minority cases?

SMOTE (Chawla et al. 2002) considers all minority samples and creates a synthetic case z as a point on the line segment between two minority cases x and y:

$$z = x + \alpha(y - x)$$

where α is a random number in the unit interval [0, 1]. Borderline-SMOTE (Han et al. 2005) does not deal with every minority class sample, but instead it identifies so-called border-line minority class samples which are most likely to be misclassified by a classifier. These border-line samples are then used for generating the synthetic samples in the same way as SMOTE does. ADASYN (He et al. 2008) assigns weights to the minority class samples, so minority samples are not treated equally. A large weight helps in generating many synthetic samples from the corresponding minority class sample. MWMOTE (Barua et al. 2012) first identifies the hard-to-learn informative minority class samples and assigns them weights according to their euclidean distance from the nearest majority class samples. Next, it generates the synthetic samples from the weighted informative minority class samples in the same way as SMOTE using a clustering approach. This is done in such a way that all the generated samples lie inside some minority class cluster. All previously mentioned methods are related to SMOTE in the sense that they create synhetic minority samples on a line segment between two existing minority cases. ROSE (Menardi and Torelli 2014), on the other hand, generates new minority samples based on the kernel density estimate around existing, real minority cases.

A practical question concerns the optimal, non-fraud/fraud rate, which should be the goal by doing oversampling. One popular trial-and-error approach to determine the optimal class distribution works as follows: In the first step, a classifier is built on the original data set with the skewed class distribution, which has, for example, 99% non-fraudsters and 1% fraudsters. The performance of this model is then measured on an independent validation data set. In a next step, oversampling is used to change the class distribution to for example, 90% and 10%. Again, the model is evaluated. Subsequent models are built on samples of 85% versus 15%, 80% versus 20%, 75% versus 25%, and so on. Each time the performance is recorded. When the performance starts to stagnate or drop, the procedure stops and the optimal odds ratio is found. Although it does depend on the data characteristics and quality, practical experience shows that the ratio 90% non-fraudsters versus 10% fraudsters is quite commonly used in the industry. In the next section, we review performance measures that are suited for imbalanced data sets.

3 Model evaluation for imbalanced data sets

When performing a classification task, assessing the classifier's quality plays a crucial role that is at least as important as estimating the model, especially in a class imbalance context. By labeling one class as a positive and the other class as a negative, the performance of binary classification algorithms is typically measured by using a confusion matrix as illustrated in Table 1. The rows represent the class as predicted by the model and the columns are the actual class. Typically, the minority class (i.e. fraud) is used as the positive class and the majority class (i.e. legitimate) as the negative class. In the confusion matrix, we count the following numbers:

- TN = number of correctly classified negative cases (True Negatives), e.g. correctly identified legitimate cases
- FP = number of negative cases incorrectly classified as positive (False Positives), e.g. legitimate cases wrongly labeled as fraudulent
- FN = number of positive cases incorrectly classified as negative (False Negatives), e.g. undetected fraud cases
- TP = number of correctly classified positive cases (True Positives), e.g. detected fraud cases

Several performance measures can be derived from Table 1. The two most common are predictive accuracy and error rate which are defined as

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \text{ and } Error\ rate = 1 - Accuracy.$$

The main problem associated with the accuracy and error rate measures is their dependence on the distribution of positive class and negative class samples in the data set. This makes them not well suited for imbalanced learning problems (He and Garcia 2009). Other evaluation metrics can be derived from Table 1 to assess learning from imbalanced data, such as:

$$\begin{aligned} Precision &= \frac{TP}{TP + FP} \\ Recall &= \frac{TP}{TP + FN} \\ F_1\text{-}measure &= \frac{2 \cdot precision \cdot recall}{recall + precision} \end{aligned}$$

Precision is the proportion of actual fraud cases among the cases that are predicted as fraud by the model. Recall or sensitivity (true positive rate), on the other hand, is the proportion of fraudulent transfers that are detected by the model. The F_1-measure combines precision and recall as an harmonic mean for maximizing the performance on a single class. Hence, it is used for measuring the performance of the classifier on the minority class samples.

Note that the measures above depend on the cut-off value.

Each of the measures above are calculated for a given confusion table that is based on a certain cutoff value. The receiver operating characteristic (ROC) curve, as shown on the left plot in Fig. 1, is obtained by plotting for each possible cutoff value the false positive rate (*FPR*) on the *X*-axis and the sensitivity or true positive rate (*TPR*) on the *Y*-axis, where

Table 1 Confusion matrix for binary classification

	Actual	Actual
	Negative	Positive
Predicted	True negatives	False negatives
Negative	(TN)	(FN)
Predicted	False positives	True positives
Positive	(FP)	(TP)

$$FPR = \frac{FP}{FP + TN} \quad \text{and} \quad TPR = \frac{TP}{TP + FN}.$$

The false positive rate is also referred to as the inverted specificity (i.e. 1-specificity) where specificity or true negative rate is the total number of true negatives divided by the sum of the number of true negatives and false positives. Provost and kohavi (1998) argued that ROC curves as an alternative to accuracy estimation for comparing classifiers would enable stronger and more general conclusions. For more information about ROC curves we refer to Krzanowski and Hand (2009) and Swets (2014).

Probably the most popular tool today to measure the performance of a classifier is then the area under this ROC curve (Fawcett 2004, 2006; Ling et al. 2003), usually known as AUC. The AUC of a classifier can be interpreted as being the probability that a randomly chosen minority case (i.e. fraud) is predicted a higher score than a randomly chosen majority case (i.e. legitmate). Therefore, a higher AUC indicates superior classification performance.

However, when dealing with highly imbalanced data sets, AUC (and ROC curves) may be too optimistic and the area under the Precision-Recall curve (AUPRC) gives a more informative picture of an algorithm's performance (Davis and Goadrich 2006). As the name suggest, the Precision-Recall curve (right plot in Fig. 1) plots the precision (Y-axis) against the recall (X-axis). Both ROC as PR curves use the recall or sensitivity, but the ROC curve also plots the *FPR* whereas PR curves focus on precision. In the denominator of *FPR*, one sums the number of

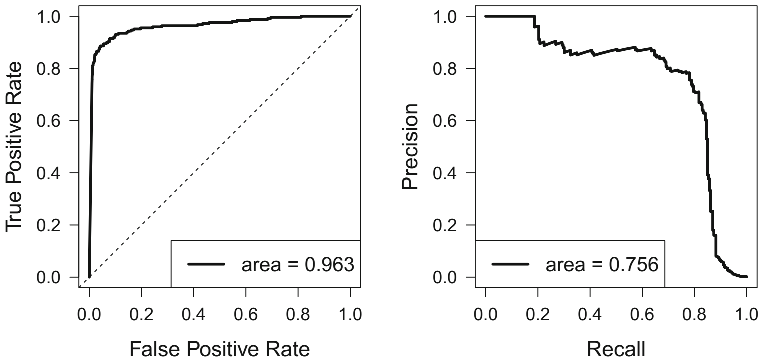

Fig. 1 (Left) example of an ROC curve. (Right) example of an Precision-Recall curve

true negatives and false positives. In highly imbalanced data, the number of negatives (good observations) is much larger than the number of positives (fraudulent observations) and hence the number of true negatives is typically very high compared to the number of false positives. Therefore, a large increase or decrease in the number of false positives will have almost no impact on *FPR* in the ROC curves. Precision, on the other hand, compares the number of false positives to the number of true positives and hence copes better with the imbalance between positives and negatives. Because Precision is more sensitive to class imbalance, the area under the Precision-Recall curve is better to highlight differences between models for highly imbalanced data sets.

4 Methodology: robROSE

We introduce a robust version of the ROSE algorithm, which does not oversample minority outcasts and additionally takes the covariance structure of the data into consideration. Artificial observations are generated only from observations which are considered clean or normal observations. Anomalous observations can have a huge influence on the model estimation. To reduce the influence of such observations robust modelling techniques can be used. However, even with robust models we may introduce too many clustered outliers with oversampling such that the model is distorted. Therefore, we introduce a method, called robROSE, as a robust alternative to the ROSE algorithm which uses only non-outlying minority cases for oversampling.

For the identification of outliers in the minority group we use robust Mahalanobis distances, i.e. Mahalanobis distances with respect to the robust center $\hat{\boldsymbol{\mu}}_1$ and scatter estimator $\hat{\boldsymbol{\Sigma}}_1$ of the minority samples. More precisely, we apply the Reweighted MCD estimator (Rousseeuw and Driessen 1999) on the minority samples. Since we compute the covariance structure for outlier identification we can also use this information to define a density around each observation of the minority samples to generate artificial observations. In contrast to the ROSE algorithm this offers the advantage that we are able to consider the covariance structure of the minority samples when we generate artificial observations.

The algorithm is outlined in detail in Algorithm 1. Let $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ denote the full data set, $\boldsymbol{X}_1$ and $\boldsymbol{X}_0$ denote its subsets belonging to the minority class and majority class, respectively. We compute the center and scatter of the minority samples with the Reweighted MCD estimator and obtain $\hat{\boldsymbol{\mu}}_1$ and $\hat{\boldsymbol{\Sigma}}_1$, respectively. Then we compute robust Mahalanobis distances (MD) for each $\boldsymbol{x}_i \in \boldsymbol{X}_1$ to identify outliers within the group of minority samples:

$$MD(\boldsymbol{x}_i, \hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{\Sigma}}_1) = \sqrt{(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_1)^T \hat{\boldsymbol{\Sigma}}_1^{-1} (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_1)}$$

The squared Mahalanobis distances follow a χ^2-distribution with p degrees of freedom. Observations with squared MD larger than the 99.9% quantile of the χ^2-distribution are considered as outliers and are excluded from oversampling. The cutoff quantile of 99.9% is relatively large compared to other applications of outlier detection. We want to avoid loosing good observations and therefore we take this rather conservative approach to outlier detection since the number of observations in the minority group is already rather small in the context of imbalance classification. Note that the covariance estimation is based on a Reweighted MCD estimator with 50% breakdown point.

After identifying the collection of non-outlying minority samples, this set of observations is used for oversampling. We start by randomly selecting one of the non-outlying minority samples denoted by $\boldsymbol{x}_j$. An artificial observation is then generated from the multivariate normal distribution with center $\boldsymbol{x}_j$ and covariance matrix

$$\hat{\boldsymbol{\Sigma}}_x = \text{ diag } (H, ..., H)\hat{\boldsymbol{\Sigma}}_1 \text{ diag } (H, ..., H) \tag{1}$$

with $H = h * c$ and $c = (4/((p+2)n))^{(1/(p+4))}$ and a tuning constant here set to $h = 0.5$. $\hat{\boldsymbol{\Sigma}}_x$ is a shrunken version of $\hat{\boldsymbol{\Sigma}}_1$ with the same shape, but different size. Together with $\boldsymbol{x}_j$ as center, $\hat{\boldsymbol{\Sigma}}_x$ defines a multivariate normal distribution describing the neighbourhood of $\boldsymbol{x}_j$. Random samples drawn from this distribution are similar to $\boldsymbol{x}_j$.

The motivation for $c = (4/((p+2)n))^{(1/(p+4))}$ originates from Gaussian kernels with diagonal smoothing matrix as it is used for ROSE (Bowman et al. 1997). We follow this approach such that our proposed method is equivalent to ROSE in case of a diagonal covariance matrix and no identified outliers.

Remarks robROSE only oversamples observations from the minority class whose squared Mahalanobis distances are smaller than the 99.9% quantile of the χ^2-distribution. In other words, the minority samples whose squared Mahalanobis distances are larger than the 99.9% quantile of the χ^2-distribution are considered outliers by robROSE and, therefore, they are excluded from the oversampling procedure. Using the 99.9% quantile can be considered conservative. However, this is done on purpose. Since the number of minority samples is already (very) small by definition, we want to be conservative when excluding minority cases. Therefore, we propose to use the 99.9% quantile in order to exclude only very extreme outlying minority observations from the oversampling procedure. The term "conservative" thus refers to the low number of minority cases that are potentially excluded from oversampling by robROSE instead of the "extreme nature" of those minority samples. In order to support our choice of the 99.9% quantile, we performed additional simulations whose results can be consulted in Sect. 1 of the Supplementary Material.

The robROSE method uses the Mahalanobis distances as calculated by the Reweighted MCD with 50% breakdown point in order to identify outlying minority cases. The main reason for using the Reweighted MCD is because the reweighting step increases the efficiency of the MCD estimator while not decreasing its breakdown value. The robROSE method uses a default value of 50% for the α parameter that controls the size of the subsets over which the determinant is minimized. Thus, roughly 50% of minority cases are used for computing the determinant. The user may choose to increase the value for α, for example $\alpha = 0.75$.

Allowed values are between 0.5 and 1 and the default is 0.5 which provides the best breakdown value. If we choose $\alpha = 1$, then we risk that robROSE will oversample outlying fraud cases which should be avoided. However, if the user is convinced that (almost) no outliers are present, then the user could of course increase the α parameter. Section 2 of the Supplementary Material contains additional simulations. We conclude that it is a good choice to take $\alpha = 0.5$ in order to achieve the highest robustness while a high efficiency is achieved by the reweighthing step of the Reweighted MCD.

Algorithm 1: robROSE.

Let $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ denote the full data set and let $\boldsymbol{X}_1$ and $\boldsymbol{X}_0$ denote its subset belonging to the minority class and majority class, respectively.

- Calculate robust covariance $\hat{\boldsymbol{\Sigma}}_1$ and robust center $\hat{\boldsymbol{\mu}}_1$ of $\boldsymbol{X}_1$ with the Reweighted MCD estimator.
- Calculate robust Mahalanobis distance for $\boldsymbol{X}_1$

$$MD(\boldsymbol{x}_i, \hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{\Sigma}}_1)^2 = (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_1)^T \hat{\boldsymbol{\Sigma}}_1^{-1} (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_1)$$

- Identify the index set of non-outlying minority observations

$$I = \{i : \boldsymbol{x}_i \in \boldsymbol{X}_1 \text{ and } MD(\boldsymbol{x}_i)^2 < \chi^2_{p,0.999}\} \tag{2}$$

  with $\chi^2_{p,0.999}$ the 99.9% quantile of the χ^2-distribution with p degrees of freedom.
- Generate artificial observations until the desired balance is reached by repeating the following:
  - Randomly select the index $j \in I$ and obtain the observation $\boldsymbol{x}_j$
  - Generate a random sample $\boldsymbol{z} \sim N_p\left(\boldsymbol{x}_j, \hat{\boldsymbol{\Sigma}}_x\right)$ from the multivariate normal distribution with center $\boldsymbol{x}_j$ and scatter matrix

$$\hat{\boldsymbol{\Sigma}}_x = \operatorname{diag}(H, ..., H) \hat{\boldsymbol{\Sigma}}_1 \operatorname{diag}(H, ..., H)$$

    where $H = h * c$ and $c = (4/((p+2)n))^{(1/(p+4))}$ and $h = 0.5$.
- Return a matrix $\boldsymbol{Z}$ of artificial observations.

This oversampling technique can also be applied if categorical variables are present in which case the artificial sample inherits the same categories as the minority sample on which it is based.

The `R` code implementation of robROSE is available in the `R` package `robROSE` at github.com/SebastiaanHoppner/robROSE.

Oversampling can be considered as a preprocessing step for the training data which is independent of the chosen model class. The artificial observations generated from the training data are joined with the real observations of the training data. The resulting data set is then used for model estimation.

In order to be able to assess the model it is crucial to have an independent test data set. In Menardi and Torelli (2014) it is recommended to use artificial data for the model assessment as well. They argue that it can help improving the estimation of the posterior probabilities of the minority class. However, if tuning constants are not properly chosen, then this way of assessing the model's performance can not only result in bad models but also in misleading evaluation. Therefore, we highly recommend to use part of the original data as test data for the model evaluation.

We will illustrate the oversampling techniques SMOTE, ROSE and our robROSE on an artificial toy example. Consider the data in the top-left corner of Figure 2 which contains both legitimate cases (class 0 in blue) and fraud cases (class 1 in black). Notice how the two fraud cases on the right hand side deviate from the other fraud cases. In order to balance the distribution between the two classes, SMOTE creates synthetic minority (i.e. fraud) samples (in red). However, many of these synthetic cases coincide with regular cases due to the two outlying fraud cases. This further complicates analytical models in detecting fraud. Similarly, ROSE creates synthetic cases around the neigborhood of the two outlying minority samples. Our proposed robust version, robROSE, does not oversample the minority outcasts and additionally takes the covariance structure of the data into consideration as illustrated by the elliptical contours.

5 Simulation study

5.1 Simulation setting

5.1.1 Data simulation

A simulation study is performed to study the properties of the proposed method. We use a similar setting as in Menardi and Torelli (2014). For each class, we generate data from a multivariate normal distribution with 10 variables. The covariance matrix used for the majority class is a diagonal matrix with ones in the diagonal, denoted by $\mathbf{\Sigma_0}$. For the minority class, the covariance matrix $\mathbf{\Sigma_1}$ has ones in the diagonal, 0.5 in the first off-diagonals and zeros elsewhere. Let $\boldsymbol{\mu_0} = (\boldsymbol{0}, \ldots, \boldsymbol{0})$ and $\boldsymbol{\mu_1} = (\boldsymbol{1/3}, \ldots, \boldsymbol{1/3})$ denote the centers of the majority class and minority class, respectively. Samples for each class are generated from these two distributions, respectively.

We want to study the effect of imbalance between two classes and not focus on the decrease in classification performance due to a decreasing number of observations. Therefore, we fix the number of minority samples $n_1 = 100$ and increase the number of majority samples n_0 in order to change the ratio of the class sizes. With $n_0 \in \{900, 1900, 9900\}$ we obtain imbalance ratios of 10%, 5% and 1%.

We split the data into 70% training and 30% test data, stratified according to the class indicator. Note that the oversampling techniques may only be applied on the training data. The data generation and the split into training and test data is repeated

100 times. The performance of each classifier is reported as the average AUC and AUPRC and its standard errors over the 100 repetitions.

A second simulation setting is used to investigate the effect of anomalous minority samples. In the training set 10% outliers are generated in the minority class by replacing randomly selected minority samples by samples generated from a multivariate normal distribution with a different center $\boldsymbol{\mu}_{out} = (\boldsymbol{-10}, \boldsymbol{-2}, \ldots, \boldsymbol{-2})$ and the same covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{1}}$.

Figure 3 visualizes the first and second principal component of the simulated data with contamination and oversampling by robROSE (left) and ROSE (right).

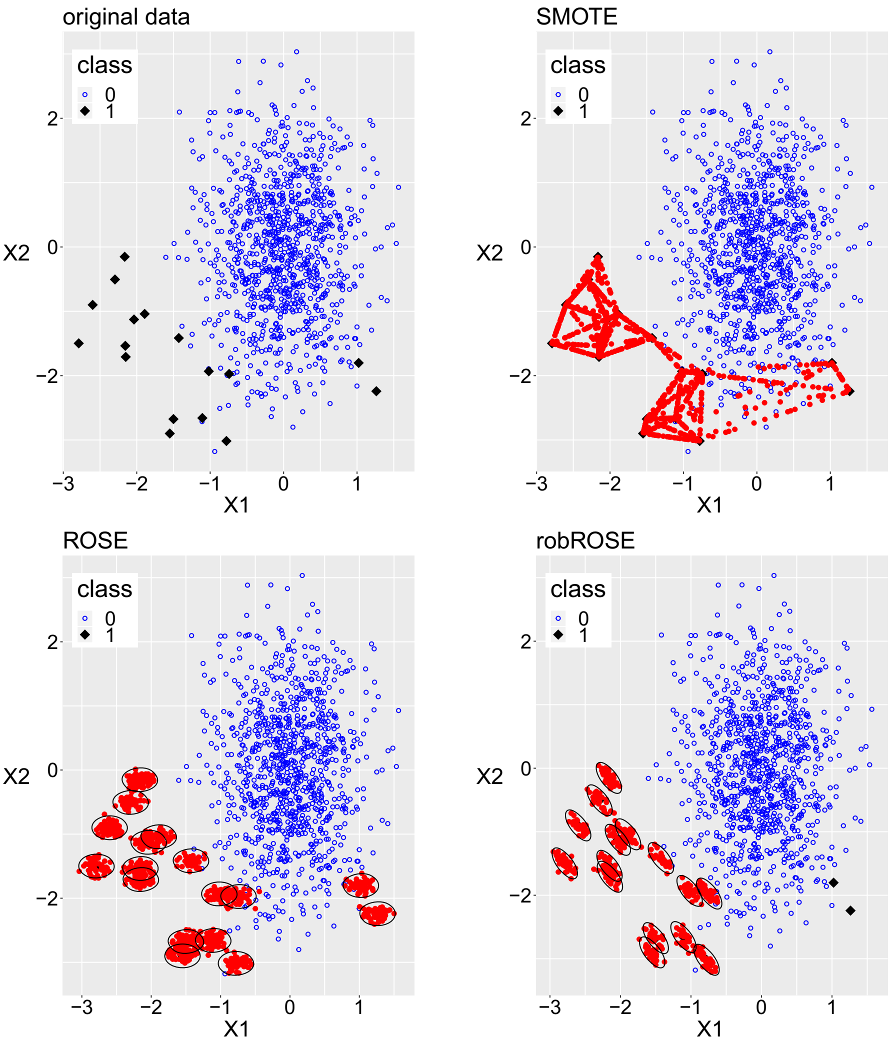

Fig. 2 Illustration of SMOTE, ROSE and robROSE on small artificial dataset

5.1.2 Oversampling and model estimation

In the simulation study we consider four approaches for model estimation with imbalanced data. First, we train the classifiers on the the generated, imbalanced data without applying any balancing strategy before the modeling. Next, we re-balance the training set by using the oversampling methods SMOTE, ROSE and our proposed method robROSE. Finally, we train the classifiers on each of these three oversampled training sets and compare their performance. The oversampling proportion for all settings is 10, i.e. we generate a data set of artificial observations nine times as large as the number of minority samples. The oversampled minority class, which includes both the original and artificial observations, is 10 times the size of the number of original minority samples. On each training data set we estimate a logistic regression model and a robust logistic regression model with the function `glmrob()` in the `R` package `robustbase` (Maechler et al. 2018; Cantoni and Ronchetti 2001; Valdora and Yohai 2014). Note, however, that the oversampling strategies are model independent.

5.2 Simulation results

5.2.1 Simulation setting 1 - no outliers

Results for simulation setting 1 are summarized in Tables 2, 3, 4 and 5. In this setting all oversampling methods (i.e. SMOTE, ROSE and robROSE) slightly improve the results for AUC. For both logistic regression and robust logistic regression all oversampling methods perform equally good in terms of AUC. For AUPRC, there is no evidence of any difference if an oversampling method is applied or not, neither for logistic regression nor for robust logistic regression.

5.2.2 Simulation setting 2 - with outliers

Tables 6,7,8and 9 present the results for simulation setting 2 where we introduce minority outcasts. Our robROSE method clearly outperforms other oversampling methods in terms of AUPRC and AUC for both logistic regression and robust logistic regression. For logistic regression models using ROSE and SMOTE still improves the results for AUC substantially compared to no oversampling but not for AUPRC.

Interestingly, robust logistic regression performs worse than logistic regression in the imbalanced setting with outliers in the minority class. The robust method suffers more heavily from the imbalance between both classes than its classical counterpart and has lower AUC and AUPRC. With robust logistic regression, the performance of AUC and AUPRC is improved by all oversampling methods. The best results are obtained by using robROSE which achieves comparable performance for both robust logistic regression and logistic regression models. The performance of the estimators using robROSE is comparable to the performance in the first simulation setting without outliers. We conclude that robROSE give slightly better results than ROSE, although the classifiers also perform well without applying oversampling

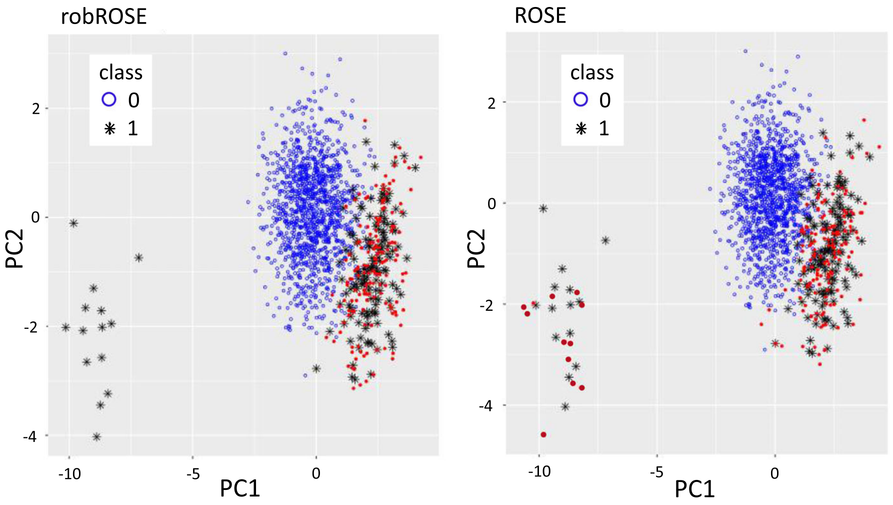

Fig. 3 First and second principal component of simulated data (majority class 0 in blue, minority class 1 in black) and artificial samples (in red) generated with robROSE (left) and ROSE (right)

techniques here. This might be due to the application of principal component analysis as a first preprocessing step, but unfortunately we don't have access to the raw data.

6 Credit card transaction data

We consider the Credit Card Transaction Data available at kaggle.com/mlg-ulb/creditcardfraud. The data consists of transactions made by credit cards in September 2013 by European cardholders. This data set presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The data set is highly imbalanced because the positive class (frauds) account for only 0.172% of all transactions. It contains only numerical input variables which are the result of a PCA transformation. Due to confidentiality issues, the original features and more background information about the data is not provided. Features V1, V2, ..., V28 are the principal components obtained with PCA. The only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the

Table 2 Simulation setting 1: average AUPRC (and standard error) for logistic regression models (imbalanced refers to no oversampling)

n_0	Imbalanced		SMOTE		ROSE		robROSE	
900	0.232	(.003)	0.235	(.003)	0.235	(.003)	0.235	(.003)
1900	0.124	(.002)	0.125	(.002)	0.125	(.002)	0.125	(.002)
9900	0.027	(.000)	0.027	(.000)	0.027	(.000)	0.027	(.000)

Table 3 Simulation setting 1: average AUC (and standard error) for logistic regression models

n_0	Imbalanced		SMOTE		ROSE		robROSE	
900	0.823	(.003)	0.835	(.003)	0.834	(.003)	0.836	(.003)
1900	0.812	(.003)	0.828	(.002)	0.827	(.002)	0.829	(.002)
9900	0.818	(.003)	0.824	(.002)	0.824	(.002)	0.824	(.002)

Table 4 Simulation setting 1: average AUPRC (and standard error) for robust logistic regression models

n_0	Imbalanced		SMOTE		ROSE		robROSE	
900	0.227	(.003)	0.235	(.003)	0.235	(.003)	0.234	(.003)
1900	0.123	(.003)	0.125	(.001)	0.125	(.002)	0.125	(.002)
9900	0.028	(.001)	0.027	(.000)	0.027	(.000)	0.027	(.000)

Table 5 Simulation setting 1: average AUC (and standard error) for robust logistic regression models

n_0	Imbalanced		SMOTE		ROSE		robROSE	
900	0.799	(.003)	0.837	(.002)	0.835	(.003)	0.837	(.002)
1900	0.780	(.004)	0.831	(.002)	0.827	(.002)	0.831	(.002)
9900	0.790	(.004)	0.806	(.003)	0.803	(.003)	0.801	(.003)

seconds elapsed between each transaction and the first transaction in the data set. The feature ‘Amount’ is the transaction amount. Feature ‘Class’ is the response variable which takes value 1 in case of fraud and 0 otherwise. We select the features V1, V2, ..., V28 and the logarithmically transformed Amount as predictor variables for the classification methods. Each of these 29 predictors are scaled to zero mean and unit variance. To keep the analysis manageable, we only consider main effects and we do not include interactions of any degree.

The following classifiers are used: a decision tree with a maximum depth of 8 built by the CART algorithm (Breiman et al. 1984) and logistic regression. The

Table 6 Simulation setting 2: average AUPRC (and standard error) for logistic regression models

n_0	Imbalanced		SMOTE		ROSE		robROSE	
900	0.158	(.003)	0.156	(.003)	0.160	(.003)	0.225	(.003)
1900	0.084	(.003)	0.085	(.002)	0.086	(.003)	0.125	(.002)
9900	0.015	(.001)	0.016	(.001)	0.016	(.001)	0.027	(.000)

Table 7 Simulation setting 2: average AUC (and standard error) for logistic regression models

n_0	Imbalanced		SMOTE		ROSE		robROSE	
900	0.647	(.005)	0.679	(.004)	0.677	(.004)	0.824	(.003)
1900	0.624	(.005)	0.664	(.004)	0.660	(.005)	0.828	(.002)
9900	0.602	(.005)	0.642	(.004)	0.643	(.005)	0.821	(.002)

Table 8 Simulation setting 2: average AUPRC (and standard error) for robust logistic regression models

n_0	Imbalanced		SMOTE		ROSE		robROSE	
900	0.153	(.004)	0.154	(.002)	0.159	(.003)	0.231	(.003)
1900	0.077	(.003)	0.086	(.003)	0.087	(.003)	0.128	(.001)
9900	0.011	(.000)	0.016	(.001)	0.016	(.001)	0.027	(.001)

Table 9 Simulation setting 2: average AUC (and standard error) for robust logistic regression models

n_0	Imbalanced		SMOTE		ROSE		robROSE	
900	0.605	(.006)	0.682	(.004)	0.678	(.004)	0.835	(.002)
1900	0.564	(.007)	0.658	(.005)	0.652	(.005)	0.836	(.002)
9900	0.497	(.005)	0.598	(.006)	0.601	(.007)	0.801	(.003)

performance of each classifier is assessed by doing two-fold cross validation 5 times such that in each repetition half of the data is used for trained and the other half is used for testing. The results are summarized in Tables 10 and 11 .

7 Customer churn data

Imbalanced classification is of course not only a challenge in fraud detection, but is a problem that comes up in many real-world applications. Therefore, classification on imbalanced data sets is a popular topic in machine learning research (Krawczyk 2016). For example in churn prediction or credit scoring, it is also important to solve the class imbalance problem (Zhu et al. 2017; Marqués et al. 2013).

In this Section we will illustrate the good performance of robROSE on a real churn data set. This data set describes the customer churn of a Korean telecommunication firm focusing only on customers which are companies. Predictor variables are the active months of the customer, its total revenue, the number of employees and the corporate size. These four variables are used to predict customer churn. This is a classic example of imbalanced data since the majority of customers does not churn and we have a small group of minority samples constituting the group of churners. Due to the heterogeneity of companies extreme values can be expected, which may have a heavy influence on the model estimation. This motivates the investigation of oversampling and robust approaches.

The total number of observations in this data set is $n = 13601$ (churn: $n_1 = 3072$, regular: $n_0 = 10529$). The imbalance ratio is therefore 22.6% churn which is a rather high proportion for these kind of problems. We reduced the number of churn observations in our study from 22.6% to 5% and 1%, resulting in $n_1 \in \{3072, 554, 106\}$, respectively, to mimic a more extreme imbalance which is more common.

For each extracted data set the predictor variables were robustly centered by the median and scaled by the median absolute deviation (MAD). Similar as in the simulation framework, 70% of all observations are used for training and the remaining 30% is used for assessing the classifier's performance. The training and test set are stratified with respect to the churn indicator such that they have the same class ratio. The split into training and test data is repeated five times.

The results are summarized in Tables 12, 13, 14 and 15. The AUC of logistic regression and robust logistic regression is not affected by decreasing the number of churn observations and cannot be improved by any oversampling technique. The AUPRC on the other hand is heavily effected by the increasing imbalance of churn samples. SMOTE and ROSE cannot improve the AUPRC while robROSE achieves substantially better results. The results of AUPRC for robust logistic regression with robROSE are slightly worse than for logistic regression (since the results lie only within one standard deviation).

8 Conclusion

Fraud detection can be presented as a binary classification problem with a highly imbalanced class distribution, where fraudsters belong to the minority class. In most applications, the fraud rate is typically less than 0.5%. This imbalance problem brings significant challenges to fraud detection and has a great negative impact on standard classification algorithms. Most algorithms tend to bias towards the majority class and may even classify all observations as non-fraudulent, yielding a high overall accuracy but unacceptably low precision with respect to the minority class of interest.

A popular solution to solve the problem of learning from imbalanced data sets are sampling-based methods. In this paper we focus on synthetic oversampling

Table 10 Credit card transaction data: average AUC (and standard error)

	Imbalanced		SMOTE		ROSE		robROSE	
CART	0.8990	(.0181)	0.9045	(.0118)	0.8963	(.0160)	0.9102	(.0118)
Logit	0.9723	(.0050)	0.9735	(.0059)	0.9766	(.0046)	0.9737	(.0060)

Table 11 Credit card transaction data: average AUPRC (and standard error)

	Imbalanced		SMOTE		ROSE		robROSE	
CART	0.7123	(.0422)	0.7180	(.0301)	0.6428	(.0581)	0.7037	(.0363)
Logit	0.7553	(.0203)	0.7622	(.0210)	0.7541	(.0240)	0.7621	(.0222)

methods, which have been proven to be very successful in various business applications (e.g. credit scoring, churn prediction and fraud detection). These methods add new information to the original data set by creating extra synthetic minority class samples based on the existing minority samples that are available in the data set.

In real data sets, it often happens that outliers or anomalies are present. Outliers may be errors, but they could also have been recorded under exceptional circumstances. Outliers can be isolated or may come in clusters, indicating that there are subgroups in the population that behave differently. When outliers are present in the data, the oversampling techniques will generate synthetic samples based on these outliers, which may distort the detection algorithm and make the resulting analysis unreliable. Therefore, it is very important to be able to detect these outliers.

In practice, one often tries to detect these outliers using traditional diagnostics. However, classical methods can be affected by outliers so strongly that the resulting fitted model does not allow to detect the deviating observations. This is called the masking effect. In the worst case the effect of outliers on a classical fit can be so large that regular observations appear to be outlying, which is known as swamping. On the other hand, robust methods can resist the effect of outliers and therefore allow to detect outliers as the observations that deviate substantially from the robust fit.

In this paper, we have presented a robust version of ROSE, called robROSE, which can cope simultaneously with the problem of imbalanced data and the presence of outliers. Moreover, our robROSE algorithm also offers the advantage that we are able to consider the covariance structure of the minority samples when generating artificial observations. Its good performance is illustrated in a simulation study and on real data sets from fraud detection and churn prediction.

Table 12 Churn data Korean corporate: average AUPRC (and standard error) for logistic regression models

n_1	Imbalanced		SMOTE		ROSE		robROSE	
3072	0.392	(.008)	0.358	(.006)	0.352	(.005)	0.426	(.002)
554	0.130	(.011)	0.122	(.012)	0.124	(.014)	0.206	(.010)
106	0.055	(.022)	0.089	(.045)	0.056	(.017)	0.124	(.053)

Table 13 Churn data Korean corporate: average AUC (and standard error) for logistic regression models

n_1	Imbalanced		SMOTE		ROSE		robROSE	
3072	0.600	0.003	0.595	0.003	0.595	0.003	0.597	0.002
554	0.595	0.011	0.592	0.011	0.594	0.010	0.604	0.011
106	0.590	0.046	0.596	0.051	0.598	0.050	0.609	0.050

Table 14 Churn data Korean corporate: average AUPRC (and standard error) for robust logistic regression models

n_1	Imbalanced		SMOTE		ROSE		robROSE	
3072	0.392	0.008	0.347	0.006	0.346	0.007	0.395	0.001
554	0.130	0.011	0.122	0.012	0.124	0.014	0.175	0.005
106	0.055	0.022	0.087	0.044	0.075	0.027	0.109	0.052

Table 15 Churn data Korean corporate: average AUC (and standard error) for robust logistic regression models

n_1	Imbalanced		SMOTE		ROSE		robROSE	
3072	0.600	0.003	0.595	0.004	0.595	0.004	0.585	0.002
554	0.595	0.011	0.592	0.011	0.596	0.011	0.598	0.009
106	0.587	0.048	0.600	0.052	0.600	0.055	0.608	0.043

Acknowledgements This work was supported by the BNP Paribas Fortis Chair in Fraud Analytics and Internal Funds KU Leuven under Grant C16/15/068.

References

Bahnsen Alejandro Correa, Stojanovic Aleksandar, Aouada Djamila, Ottersten Björn (2013) Cost sensitive credit card fraud detection using bayes minimum risk. In *2013 12th international conference on machine learning and applications*, volume 1, pages 333–338. IEEE

Barua Sukarna, Islam Md Monirul, Yao Xin, Murase Kazuyuki (2012) Mwmote–majority weighted minority oversampling technique for imbalanced data set learning. IEEE Trans Knowl Data Eng 26(2):405–425

Bowman Adrian W, Azzalini Adelchi (1997) Applied smoothing techniques for data analysis: the kernel approach with S-Plus illustrations, volume 18. OUP Oxford

Breiman Leo, Friedman Jerome, Olshen Richard, Stone Charles (1984) Classification and regression trees. wadsworth int. Group 37(15):237–251

Cantoni Eva, Ronchetti Elvezio (2001) Robust inference for generalized linear models. J Am Statistical Assoc 96(455):1022–1030

Cerioli Andrea, Perrotta Domenico (2014) Robust clustering around regression lines with high density regions. Adv Data Anal Classification 8(1):5–26

Chawla Nitesh V, Bowyer Kevin W, Hall Lawrence O, Kegelmeyer W Philip (2002) Smote: synthetic minority over-sampling technique. J Artif Intell Res 16:321–357

Davis Jesse, Goadrich Mark (2006) The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pages 233–240. ACM

Fawcett Tom (2004) Roc graphs: Notes and practical considerations for researchers. Mach Learn 31(1):1–38

Fawcett Tom (2006) An introduction to roc analysis. Patt Recog Lett 27(8):861–874

Han Hui, Wang Wen-Yuan, Mao Bing-Huan (2005) Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing*, pages 878–887. Springer

Hand David J, Whitrow Christopher, Adams Niall M, Juszczak Piotr, Weston Dave (2008) Performance criteria for plastic card fraud detection tools. J Operational Res Soc 59(7):956–962

He Haibo, Bai Yang, Garcia Edwardo A, Li Shutao (2008) Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pages 1322–1328. IEEE

He Haibo, Garcia Edwardo A (2009) Learning from imbalanced data. IEEE Trans knowl Data Eng 21(9):1263–1284

Holte Robert C, Acker Liane, Porter Bruce W, et al (1989) Concept learning and the problem of small disjuncts. In *IJCAI*, volume 89, pages 813–818. Citeseer

Krawczyk Bartosz (2016) Learning from imbalanced data: open challenges and future directions. Prog Artif Intell 5(4):221–232

Krzanowski Wojtek J, Hand David J (2009) ROC curves for continuous data. Chapman and Hall/CRC

Ling Charles X, Huang Jin, Zhang Harry, et al. (2003) Auc: a statistically consistent and more discriminating measure than accuracy. In *Ijcai*, volume 3, pages 519–524

Liu Xu-Ying, Wu Jianxin, Zhou Zhi-Hua (2008) Exploratory undersampling for class-imbalance learning. IEEE Trans Syst, Man, Cybernetics, Part B (Cybernetics) 39(2):539–550

Maechler M, Rousseeuw PJ, Croux C, Todorov V, Ruckstuhl A, Salibian-Barrera M, Verbeke T, Koller M, Conceicao ELT, Anna di Palma M (2018) *robustbase: Basic Robust Statistics*. R package version 0.93-3

Marqués Ana Isabel, García Vicente, Sánchez José Salvador (2013) On the suitability of resampling techniques for the class imbalance problem in credit scoring. J Operational Res Soci 64(7):1060–1070

Menardi Giovanna, Torelli Nicola (2014) Rose: random over-sampling examples. Data Min Knowl Dis 28(1):92–122

Ngai Eric WT, Hu Yong, Wong Yiu Hing, Chen Yijun, Sun Xin (2011) The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. Decision Support Syst 50(3):559–569

Phua Clifton, Lee Vincent, Smith Kate, Gayler Ross (2010) A comprehensive survey of data mining-based fraud detection research. arXiv preprint arXiv:1009.6119

Provost F Fawcett T, kohavi r (1998) the case against accuracy estimation for comparing classifiers. In *Proceedings of the Fifteenth International Conference on Machine Learning*,

Rousseeuw Peter J, Driessen Katrien Van (1999) A fast algorithm for the minimum covariance determinant estimator. Technometrics 41(3):212–223

Swets John A (2014) Signal detection theory and ROC analysis in psychology and diagnostics: Collected papers. Psychology Press,

Valdora Marina, Yohai Víctor J (2014) Robust estimators for generalized linear models. J Statistical Plan Inference 146:31–48

Van Vlasselaer Véronique, Eliassi-Rad Tina, Akoglu Leman, Snoeck Monique, Baesens Bart (2016) Gotcha! network-based fraud detection for social security fraud. Manag Sci 63(9):3090–3110

Weiss Gary M, Provost Foster (2001) The effect of class distribution on classifier learning: an empirical study. Technical Report ML- TR-43, Dept. of Computer Science, Rutgers Univ

Zhu Bing, Baesens Bart, Broucke Seppe KLM vanden (2017) An empirical comparison of techniques for the class imbalance problem in churn prediction. Inform Sci 408:84–99

Zhu Bing, Gao Zihan, Zhao Junkai, Broucke Seppe KLM vanden (2019) Iric: An r library for binary imbalanced classification. SoftwareX 10:100341

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Bart Baesens[1] · **Sebastiaan Höppner**[2] · **Irene Ortner**[3] · **Tim Verdonck**[4]

[1] Faculty of Economics and Business, KU Leuven, Naamsestraat 69, 3000 Leuven, Belgium

[2] Department of Mathematics, KU Leuven, Celestijnenlaan 200B, 3001 Leuven, Belgium

[3] Applied Statistics GmbH, Taubstummengasse 4/10, 1040 Vienna, Austria

[4] Department of Mathematics, University of Antwerp, Middelheimlaan 1, 2020 Antwerp, Belgium