A novel hybrid undersampling method for mining unbalanced datasets in banking and insurance

Article *in* Engineering Applications of Artificial Intelligence · January 2015

DOI: 10.1016/j.engappai.2014.09.019

2 authors:

G. Ganesh Sundarkumar
University of Hyderabad
4 PUBLICATIONS **33** CITATIONS
SEE PROFILE

Ravi Vadlamani
Institute for Development & Research in Ban…
71 PUBLICATIONS **346** CITATIONS
SEE PROFILE

Engineering Applications of Artificial Intelligence 37 (2015) 368–377

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Brief paper

A novel hybrid undersampling method for mining unbalanced datasets in banking and insurance

G. Ganesh Sundarkumar [a,b], Vadlamani Ravi [a,*]

[a] Center of Excellence in CRM and Analytics, Institute for Development and Research in Banking Technology, Castle Hills Road No. 1, Masab Tank, Hyderabad 500057, AP, India

[b] School of Computer & Information Sciences, University of Hyderabad, Hyderabad 500046, AP, India

ARTICLE INFO

Keywords:
Insurance fraud detection
Credit card churn prediction
Undersampling
K- Reverse Nearest Neighbourhood method
One-class support vector machine

ABSTRACT

In this paper, we propose a novel hybrid approach for rectifying the data imbalance problem by employing *k* Reverse Nearest Neighborhood and One Class support vector machine (OCSVM) in tandem. We mined an Automobile Insurance Fraud detection dataset and customer Credit Card Churn prediction dataset to demonstrate the effectiveness of the proposed model. Throughout the paper, we followed 10 fold cross validation method of testing using Decision Tree (DT), Support Vector Machine (SVM), Logistic Regression (LR), Probabilistic Neural Network (PNN), Group Method of Data Handling (GMDH), Multi-Layer Perceptron (MLP). We observed that DT and SVM respectively yielded high sensitivity of 90.74% and 91.89% on Insurance dataset and DT, SVM and GMDH respectively produced high sensitivity of 91.2%, 87.7%, and 83.1% on Credit Card Churn Prediction dataset. In the case of Insurance Fraud detection dataset, we found that statistically there is no significant difference between DT (J48) and SVM. As DT yields "if then" rules, we prefer DT over SVM. Further, in the case of churn prediction dataset, it turned out that GMDH, SVM and LR are not statistically different and GMDH yielded very high Area Under Curve at ROC. Further, DT yielded just 4 'if–then' rules on Insurance and 10 rules on churn prediction datasets, which is the significant outcome of the study.

1. Introduction

While deploying machine learning algorithms to binary classification problems, data imbalance has inevitably become a challenge to data analysts. In many real world applications such as fraud detection, default prediction, churn prediction, oil-spill detection and network intrusion detection, the class distributions follow 90%:10% proportion and beyond (Vasu and Ravi, 2011). In such problems, machine learning algorithms wrongly predict all the samples of the minority class to be those of majority class. However, this minority class usually will be the more important class. In other words, algorithms tend to be overwhelmed by the majority class and ignore the minority class. Its importance increased as more and more researchers realized that this imbalance causes suboptimal classification performance, by most algorithms (Vasu and Ravi, 2011).The study of class imbalance problem has been a hot topic in machine learning in recent years (Diamantini and Potena., 2009).

Insurance fraudulent claims blow a severe loss over billions for the insurance companies. These fraudulent claims negatively not only effect the insurance firms but also hurt the socio-economic structures (Xu et al., 2011). Fraudulent claims hurt the insurance companies a lot and if they can be detected either in on-line or off-line manner, it could reduce the loss to a great extent. Most of the insurance companies publish few data on the occurrence of insurance fraud (Lang and Wambach, 2013). Consequently, detection becomes a challenge. Further, insurance frauds grow rapidly in fields like Telecom, e-business, accounts, credit cards, etc., (Phua et al., 2007). Chan et al. (1999) mentioned various reasons for increase in credit card frauds like lack of deploying efficient technique to handle massive millions of transactions, due to highly skewed data, and further each transaction has different specified amount, and attributing the same misclassification cost rate to all transactions is yet another potential flaw in cost based mining techniques. Most of the analyses agree that approximately 20% of all insurance claims are fraudulent in some way (Sublej et al., 2011). But most of these claims go unnoticed, as fraud investigation is usually performed manually by a domain expert or investigator and he/she rarely takes support of a computer (Sublej et al., 2011). Inappropriate data representation is also a common problem, making the job for fraud investigators extremely difficult (Chan et al., 1999; Phua et al., 2004). Hence, these types of frauds should be addressed by developing models that can trigger early warnings or red flags. Jensen (1997) observed several technical difficulties indetecting fraud. Firstly, only a small portion of accident claims

* Corresponding author. Tel.: +91 40 23494042; fax: +91 40 23535157.
E-mail addresses: govindaraju.gsk@gmail.com (G.G. Sundarkumar), rav_padma@yahoo.com (V. Ravi).

http://dx.doi.org/10.1016/j.engappai.2014.09.019
0952-1976/

are fraudulent (skewed class distribution) making them extremely difficult to detect. Next, there is a severe lack of labeled data sets as labeling is expensive and time consuming. Any approach for detecting such fraud should thus be founded on moderate resources (data sets) in order to be applicable in practice. In addition to this, fraudsters discover innovative types of frauds continually.

Further, Churn/attrition is a phenomenon, where some of the existing customers cease doing business with a bank/financial institution/service provider. Over the decade and half, the number of customers with banks and financial companies is increasing. Banks provide services through various channels, like ATM, debit cards, credit cards, internet-banking, etc. The enormous increase in the number of customers has made banks conscious of the quality of service they offer. This phenomenon triggered tremendous competition amongst various banks resulting in a significant increase in reliability and quality of service from banks. The problem of customers defecting or shifting loyalties from one bank to another has become common. Churn occurs owing to reasons such as availability of latest technology, customer-friendly bank staff, low interest rates, proximity of geographical location, varied services offered, etc. Hence, there is a pressing need to develop models that can predict which existing 'loyal' customer is going to churn out or attrite in near future.

In this study, we predict fraud in automobile insurance and customer churn in credit cards by first proposing a combination of machine learning algorithms to rectify the data imbalance problem and then invoking a few classifiers for classification purpose. We propose *k*-reverse nearest neighborhood (*k*RNN) and one-class SVM (OCSVM) respectively for outlier detection and undersampling of majority class of a dataset. The article makes an in-depth description, evaluation and analysis of the proposed system.

The rest of the paper is organized as follows: Section 2 reviews work reported in the related areas. Section 3 describes proposed methodology while Section 4 presents the experimental methodology followed in the study. Section 5 discusses the results obtained and Section 6 concludes the paper.

2. Literature review

In this section, we present a brief overview of the work reported in fraud detection. A meta classifier system (Stolfo et al., 1997a, 1997b) is presented for detecting the fraud, by merging the results obtained from local fraud detection tools at different corporate sites to yield a more accurate global tool. This work was extended by Chan et al. (1999) and Stolfo et al. (2000). They proposed a scalable distributed data mining model to evaluate classification techniques using a realistic cost model. They reported significant improvement by partitioning a large data set into smaller subsets to generate classifiers using different algorithms, experimenting with fraud/non-fraud distributions within training data and using stacking to combine multiple models. Stefano and Gisella (2001) proposed a fuzzy logic control (FLC) model to evaluate an "index of suspects" on each claim efficiently for highlighting fraudulent claims. Brockett et al. (2002) introduced a mathematical technique for an apriori classification of objects when no training data with target variable exists in the context of fraud detection in body injury claims in automobile insurance. They reported that by using principal component analysis of Relative to Identified Distribution (RIDIT) scores, an insurance fraud detector can reduce uncertainty and increase the chances of targeting the appropriate claims. Later, Phua et al. (2004) proposed a fraud detection method which makes use of a single meta-classifier (stacking) to choose the best base classifiers, and then combine these base classifiers' predictions (bagging) to improve cost savings. They called this method as stacking–bagging. In this method, they used MLP together with Naïve Bayesian (NB) and C4.5 algorithms, on data partitions derived from minority over-sampling with replacement and demonstrated that stacking–bagging performs slightly better than the best performing bagged algorithm, C4.5 in terms of cost savings on a publicly available automobile insurance fraud dataset. Phua et al. (2007) conducted an exhaustive survey of data mining based fraud detection methods highlighting higher cost savings.

Several researchers proposed some standard data mining algorithms like neural networks, fuzzy logic, genetic algorithms, support vector machines, logistic regression, classification trees to handle fraud detection (Bolton and Hand, 2002; Brockett et al., 2002; Viaene et al., 2002, 2005; Perez et al., 2005; Estevez et al., 2006; Yang and Hwang, 2006; Hu et al., 2003; Kirkos and Spathis, 2007; Rupnik et al., 2007; Quah and Sriganesh, 2008; Sanchez et al., 2009). Recently, Ngai et al. (2011) conducted a comprehensive literature review on financial fraud detection and various data mining techniques handling the problem. They reported that the main data mining techniques used in this domain are logistic models, neural networks, the bayesian belief network, and decision trees. Zhu et al. (2011) proposed Nonnegative matrix factorization approach for health care fraud detection. Xu et al. (2011) proposed Random Rough subspace based Neural Network Ensemble for Insurance fraud detection and used rough set reducts to improve the consistency in the datasets. They reported a detection rate of fraud cases as 88%. They concluded that other ensemble strategies can further improve the problem solution. Sublej et al. (2011) proposed a graph based network approach for detecting automobile insurance frauds. They proposed Iterative Assessment Algorithm based on Graph components. However, most of these above mentioned studies did not handle the class imbalance problems of insurance data. Benard and Vanduffel (2014) studied mean-variance efficient portfolios. It is a quantitative approach where mean returns get maximized and variance of risk is minimized. Then they derived bounds on Sharpe ratios and demonstrated that this will be useful for fraud detection.

Regarding Churn prediction, Mozer et al. (2000) analyzed subscriber data of a major wireless carrier, by applying logistic regression, naïve neural network and a sophisticated neural network. They concluded that using a sophisticated neural net, $93 could be saved per subscriber. Smith and Gupta (2000) employed multilayer perceptron, Hopfield neural networks and self-organizing neural networks to solve churn problems. Larivie're and Van den Poel (2004) concluded that in the financial services industry, two 'critical' churn periods can be identified: the early years after becoming a customer and a second period after being a customer for some 20 years. Ferreira et al. (2004) analyzed wireless dataset from Brazil using multilayer perceptron, C4.5 decision trees, hierarchical neuro-fuzzy systems and a data-mining tool named rule evolver based on genetic algorithms (GAs). Larivie're and Van den Poel (2004) studied Application of fuzzy ARTMAP for churn prediction 431 the defection of SI customers of a large Belgian financial services provider using survival estimates, hazard ratios and multinomial probit analysis. Recently, Kumar and Ravi (2008) conducted the most comprehensive investigation on the credit card churn prediction problem in bank credit cards by resorting to data mining. They employed the balancing method such as SMOTE, undersampling, oversampling and combination of undersampling and oversampling before invoking multilayer perceptron, logistic regression, decision tree (J48), random forest, radial basis function network and support vector machine. Naveen et al. (2009) designed a new feature selection method called the 'union method' by considering the union of feature subsets selected by *t*-statistic and mutual information and employed Fuzzy ARTMAP with the selected features for predicting churn in the same dataset.

In the following, we review the past work reported in undersampling methods. Hart (1968) proposed an undersampling method, Condensed Nearest Neighbor (CNN). This method first randomly draws one example from the majority class to be combined with all

examples from the minority class to form a training set S, then use a 1-NN over S to classify the examples in the training set and move every misclassified example from the training set to S. Laurikkala (2001) proposed Neighborhood Cleaning Rule (NCR). It employed the Wilson's Edited Nearest Neighbor Rule to remove selected majority class examples. Further, Chawla et al. (2002) proposed Synthetic Minority Oversampling Technique (SMOTE) approach, in which the minority class samples are oversampled by generating artificial samples rather than just oversampling with replacement. Japkowicz (2000), Japkowicz and Stephen (2002) and Japkowicz (2003) used a synthetic dataset to study the class imbalance problem and found that independent of the training size, linearly separable domains are not sensitive to imbalance. Then they compared the effectiveness of (a) oversampling, (b) undersampling and (c) cost modifying for tackling the imbalance problem. They concluded that the bad performance of the unbalanced dataset is usually caused by small subcluster that cannot be classified accurately. Further, outliers skew the performance of a model. So, outliers need to be carefully handled first (Lee et al., 2013). Taking cue from Lee et al. (2013), we believe that outlier elimination plays a vital role in the insurance fraud detection. Soujanya et al. (2006) proposed with the concept of k reverse nearest neighborhood, having the capability of eliminating outliers simultaneously while building the clusters. It implies that it removes the noisy records which lie distantly from the normal records. Further, we agree with their point that outliers may be considered as noisy points lying outside a set of defined clusters. In our close analogy with our fraud dataset, outliers are the records which exhibit dissimilarity with the defined set of clusters and they cannot be part of representatives while undersampling the data and further noise needs to be eliminated to enhance the data quality. Hence, in the proposed methodology, we chose kRNN as a first step in the process of undersampling.

3. Proposed method

We first extracted the validation dataset, comprising 20% of the original unbalanced dataset using stratified random sampling from the dataset and left it intentionally untouched so as to validate the efficiency of the proposed model in a real life scenario. The remaining 80% dataset is subjected to the proposed novel under sampling approach. We propose a novel hybrid under sampling approach by eliminating the outliers first from the majority class using kRNN and from the resulting outlier-free dataset, we extract the support vectors using OCSVM. We particularly chose SVM here as it performs row dimensionality reduction (by way of picking up Support Vectors) while classifying the datasets. Therefore, we implicitly accomplish undersampling by way of selecting a few, important samples of the majority class. These resulting samples are then merged with the minority class samples thereby yielding a modified balanced dataset. The block diagram of the proposed approach is depicted in Fig. 1.

3.1. Algorithms used in the proposed approach

3.1.1. kReverse Nearest Neighbor (kRNN) (Soujanya et al., 2006)

Let X be d-dimensional data set,$X=\{x_1,x_2,x_3,\ldots,x_i,x_j,\ldots,x_n\}$, where n is size of the data set and x_i, x_j are any two points (samples) in X. If d_{ij} is the distance between two samples x_i and x_j then k nearest neighbor set – kNN(x_i) is defined as $\{x_j/d_{ij} < k\text{th nearest distance of } x_i\}$, where for a given point x_i, the kth smallest distance after sorting all the distances from x_i to the remaining points, kth smallest distance is the kth nearest distance of x_j.

Then, k Reverse Nearest Neighbor set kRNN(x_i) is defined as $\{x_j | x_i \in k\text{NN}(x_j)\}$, the set of all points x_j that consider x_i as their k-nearest neighbor. A point x_j belongs to kRNN(x_i) if and only if $x_i \in k$NN(x_j). Note that, in case of kNNs, for a given k value, each point in the dataset will have at least k nearest neighbors ($> k$ in case of ties), but the set of kRNNs of a point could have zero or more elements. The set of kRNNs of point x_i gives a set of points that consider x_i as their k-nearest, for a given value of k. If a point x_i has higher number of kRNNs than another point x_j, then we can say that x_i has a denser neighborhood than x_j. In other words, less the numbers of kRNNs, the farther apart are the points in the dataset to x_j, i.e. the neighborhood is sparse.

According to kRNN concept, *outlier point* is defined as a point that has less than k number of kRNNs, i.e., $|k\text{RNNs}(x_i)| < k$. Thus, less the number of kRNNs, the more distant is the point from its neighbors. To find the noisy samples/outliers, we need to set the optimal combinations of k_1, k_2 values, which depend on the nature of the data. Here $k_1 = k$th smallest distance and $k_2 = |k\text{RNNs}| =$ no of reverse neighbors. A sample is called outlier if it has less no of reverse neighbors (k_2) than k_1. The higher the difference between k_1 and k_2 the higher is the probability of a sample being an outlier. But if we choose too small value for k_1 and too large value for k_2 then we may not identify all the outliers. The values of k_1 and k_2 chosen for the two datasets are presented in Section 5.

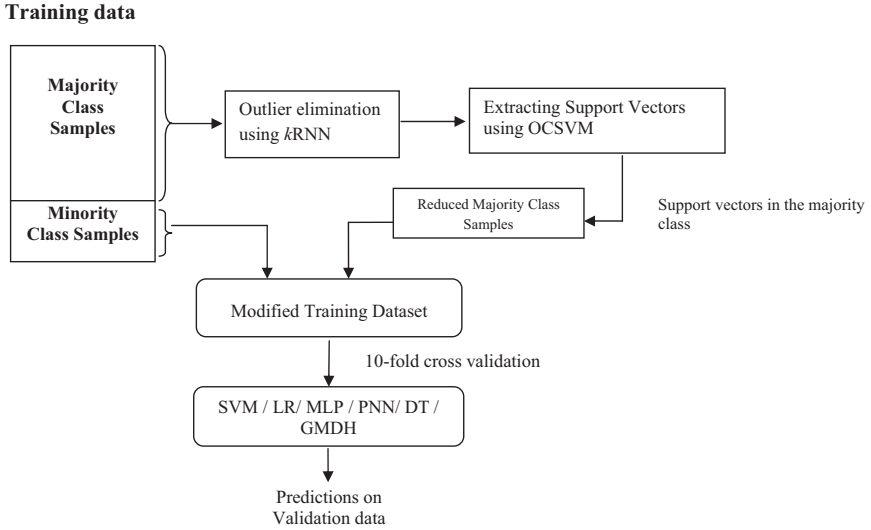

Fig. 1. Data flow of the proposed approach.

3.2. Overview of classification techniques

The techniques Logistic Regression (LR), Multi Layer Perceptron (MLP), and Decision Tree are too popular to be described here. So we present a brief overview of Support Vector Machines (SVM), Probabilistic Neural Network (PNN) and Group Method of Data Handling (GMDH).

3.2.1. Support vector machines (SVM)

A Support Vector Machine (SVM) (Vapnik, 1998) performs classification by constructing an N-dimensional hyperplane that optimally separates the data into two categories. SVM models are closely related to neural networks. Using a kernel function, SVMs are an alternative training method for polynomial, Radial Basis Function (RBF) networks and MLP classifiers, in which the weights of the network are found by solving a quadratic programming problem with linear constraints, rather than by solving a non-convex, unconstrained minimization problem, as in standard neural network training. The goal of SVM modeling is to find the optimal hyperplane that separates samples, in such a way that the samples with one category of the target variable should be on one side of the plane and the samples with the other category are on the other side of the plane. The samples near the hyperplane are the support vectors. An SVM analysis finds the hyperplane that is oriented so that the margin between the support vectors is maximized. One idea is that performance on test cases will be good if we choose the separating hyperplane that has the largest margin.

OCSVM is different from SVM in the sense that the training data belongs to only one class. It builds a boundary that separates the class from the rest of the feature space (Tax and Duin, 2004). Given the training vectors, $x_i \in R^n$, $i=1, \ldots, l$, without any class information, the primary problem is

$$\min_{w,\xi,\rho} \frac{1}{2}w^T w-\rho+\frac{1}{vl}\sum_{i=1}^{l}\xi_i, \quad \text{such that} \quad w^T\phi(x_i)\geq\rho-\xi_i \quad \text{and} \quad \xi_i>0 \tag{1}$$

where v is an upper bound on the fraction of outliers and lower bound on the fraction of support vectors. The dual problem is

$$\min_{\alpha}\frac{1}{2}\alpha^T Q\alpha\alpha, \quad \text{such that} \quad \leq\alpha_i\leq\frac{1}{l}, \quad i=1,\ldots,l, \quad e^T\alpha=v, \quad \text{and,} \quad y^T\alpha=0 \tag{2}$$

Where $Q_{i,j}=y_i y_j k(x_i,x_j)$ and for a hyperplane w that separates the point x_i from the origin with margin ρ and ξ_i accounts for possible errors.

3.2.2. Group method of data handling (GMDH)

The group method of data handling (GMDH) was introduced by Ivakhnenko (1968) as an inductive learning algorithm for modeling of complex systems. It is a self-organizing approach based on sorting-out of gradually complicated models and evaluation of them using some criterion on separate parts of the data sample (Srinivasan, 2008). The GMDH was partly inspired by research in perceptrons and learning filters. GMDH has influenced the development of several techniques for synthesizing (or "self-organizing") networks of polynomial nodes. The GMDH attempts a hierarchical solution, by trying many simple models, retaining the best, and building on them iteratively, to obtain a composition (or feed-forward network) of functions as the model. The building blocks of GMDH, or polynomial nodes, usually have the quadratic form:

$$z=w_0+w_1x_1+w_2x_2+w_3x_1^2+w_4x_2^2+w_5x_1x_2 \tag{3}$$

for inputs x_1 and x_2, coefficient (or weight) vector w, and node output, z. The coefficients are found by solving the Linear Regression equations with $z=y$, the response vector.

The GMDH network learns in an inductive way and tries to build a function (called a polynomial model), which would result in the minimum error between the predicted value and expected output. The majority of GMDH networks use regression analysis for solving the problem. The first step is to decide the type of polynomial that the regression should find. The initial layer is simply the input layer. The first layer created is made by computing regressions of the input variables and then choosing the best ones. The second layer is created by computing regressions of the values in the first layer along with the input variables. This means that the algorithm essentially builds polynomials of polynomials. Again, only the best are chosen by the algorithm. These are called survivors. This process continues until a pre-specified selection criterion is met.

3.2.3. Probabilistic neural network (PNN)

PNN was introduced by Specht (1990). It is an implementation of the statistical algorithm called kernel discriminant analysis in which the operations are organized into multilayer feed forward network with four layers: input layer, pattern layer, summation layer, and output layer. It is a pattern classification network based on the classical Bayes classifier, which is statistically an optimal classifier that seeks to minimize the risk of misclassifications. Any pattern classifier places each observed data vector $x=[x_1, x_2,\ldots, x_n]^T$, into one of the predefined classes c_i, $i=1, 2,\ldots,m$ where m is the number of possible classes. The classical Bayes pattern classifier implements the Bayes conditional probability rule that the probability $P(c_i/x)$ of x being in class c_i is given by

$$p\left(\frac{C_i}{x_i}\right)=\left(\frac{p(X/C_i)p(c_i)}{\sum_{j=1}^{m}p(x/c_i)p(c_j)}\right) \tag{4}$$

where $P(c_i/x)$ is the conditional probability density function of x given set, $P(c_i)$ is the probability of drawing data from class c_i. Vector x is said to belong to a particular class c_i, if $P(c_i/x)>P(c_j/x)$, for all $j=1, 2,\ldots,m$ and j is not equal to i. This input x is fed into each of the patterns in the pattern layer. The summation layer computes the probability $P(c_i/x)$ of the given input x to be in each of the classes c_i. The output layer selects the class for which the highest probability is obtained in the summation layer. The input is then made to belong to this class. Effectiveness of the network depends on the smoothing parameter r.

4. Experimental methodology

The effectiveness of the proposed hybrid approach was demonstrated on an insurance fraud detection dataset and a customer credit card churn dataset taken from literature.

4.1. Insurance fraud detection dataset description

The insurance fraud detection dataset is taken from Phua et al. (2004). This dataset mainly comprises the information regarding the various automobile insurance claims during the period 1994–96. The dataset, described in Table 1, comprises 31 predictor variables and 1 class variable. It consists of 15,420 samples of which 14,497 are non-fraudulent and 923 are fraudulent, which means there are 94% genuine samples and 6% fraudulent samples. Hence, the dataset is highly unbalanced in terms of the proportion of fraudulent and non-fraudulent samples.

Table 1
Attribute information of the insurance data.

S. No	Attribute name	Description
1	Month	Month in which accident took place
2	Week of month	Accident week of month
3	Day of week	Accident day of week
4	Month claimed	Claim month
5	Week of month claimed	Claim week of month
6	Day of week claimed	Claim day of week
7	Year	1994, 1995 and 1996
8	Make	Manufacturer of the car (19 companies)
9	Accident area	Rural or urban
10	Gender	Male or female
11	Marital status	Single, married, widow and divorced
12	Age	Age of policy holder
13	Fault	Policy holder or third party
14	Policy type	Type of the policy (1–9)
15	Vehicle category	Sedan, sport or utility
16	Vehicle price	Price of the vehicle with 6 categories
17	Rep. number	ID of the person who process the claim(16 ID's)
18	Deductible	Amount to be deducted before claim disbursement
19	Driver rating	Driving experience with 4 categories
20	Days: policy accident	Days left in policy when accident happened
21	Days: policy claim	Days left in policy when claim was filed
22	Past number of claims	Past number of claims
23	Age of vehicle	Vehicle's age with 8 categories
24	Age of policy holder	Policy holder's age with 9 categories
25	Policy report filed	Yes or no
26	Witness presented	Yes or no
27	Agent type	Internal or external
28	Number of supplements	Number of supplements
29	Address change claim	No of times change of address requested
30	Number of cars	Number of cars
31	Base policy(BP)	All perils, collision or liability
32	Class	Fraud found (yes or no)

4.2. Data preprocessing

It is observed that the *age* attribute in the dataset appeared twice in numerical and categorical form as well. Hence, the age attribute with numerical values is removed from the data to reduce the complexity caused by too many unique values it possesses. Further, the attributes *Year*, *Month*, *Week of the month* and *Day of week* represent the date of the accident and the attributes *Month claimed*, *Week of the month claimed* and *Day of week claimed* represent the date of the insurance claim. Thus, a new attribute *Gap* is derived from seven attributes such as *Year*, *Month*, *Week of the month*, *Day of week*, *Month claimed*, *Week of the month claimed* and *Day of week claimed*. The attribute *Gap* represents the time difference between the accident occurrence and insurance claim. Thus 24 variables which included some derived variables are selected for further study. Hence, we have 15,420 samples with 24 predictor variables and 1 class variable. The attributes of the preprocessed dataset are presented in Table 2.

4.3. Churn prediction dataset description

The churn prediction dataset is taken from a Latin-American bank, which suffered an increasing number of credit card customer churn and decided to improve its retention system. The dataset consists of two groups of features for each customer: socio-demographic and behavioral data, which are described in Table 6. The dataset consists of 21 features and 1 class label. It consists of 14,814 records of which 13,812 are non-churners and 1002 are churners, which means there are 93.24% loyal customers and mere 6.76% are churners. It is clear that the dataset is highly unbalanced in terms of proportion of churners vs. loyal customers (Business Intelligence Cup, 2004).

Table 2
Attribute information of the *preprocessed* insurance data.

S. No	Attribute name	Description
1	Gap	Time difference of accident and insurance claim
2	Make	Manufacturer of the car (19 companies)
3	Accident area	Rural or urban
4	Gender	Male or female
5	Marital status	Single, married, widow and divorced
6	Fault	Policy holder or third party
7	Policy type	Type of the policy (1–9)
8	Vehicle category	Sedan, sport or utility
9	Vehicle price	Price of the vehicle with 6 categories
10	Rep. number	ID of the person who process the claim (16 ID's)
11	Deductible	Amount to be deducted before claim disbursement
12	Driver rating	Driving experience with 4 categories
13	Days: policy accident	Days left in policy when accident happened
14	Days: policy claim	Days left in policy when claim was filed
15	Past number of claims	Past number of claims
16	Age of vehicle	Vehicle's age with 8 categories
17	Age of policy holder	Policy holder's age with 9 categories
18	Policy report filed	Yes or no
19	Witness presented	Yes or no
20	Agent type	Internal or external
21	Number of supplements	Number of supplements
22	Address change claim	No of times change of address requested
23	Number of cars	Number of cars
24	Base policy (BP)	All perils, collision or liability
25	Class	Fraud found (yes or no)

		Predicted	
		P	N
Actual	P	True Positive	False Negative
	N	False Positve	True Negative

P- Postive Class, N- Negative Class

Fig. 2. Confusion matrix.

4.4. Proposed undersampling methodology

First, the data is partitioned using the stratified random sampling method into two subsets in the ratio 80% and 20% to ensure that each subset has the same proportion of positive and negative samples as in the original data. The 80% data (11,597 non-fraudulent records and 738 fraudulent records) is used for undersampling and to build the model, while the 20% validation data (with 2900 non-fraudulent records and 185 fraudulent records) is set aside to validate the effectiveness of the model, as it is close to real world scenario (see Fig. 1).

This 80% of original unbalanced dataset is fed to different classifiers mentioned in Section 4 to build the model. To obtain statistically reliable results, 10-fold cross validation method is used throughout this study. In other words, the training dataset (80%) is divided into 10 subsets of equal size using stratified random sampling method. For all the possible choices of 9 subsets, the union of 9 subsets (training data) is used for training, and the remaining subset (test data) is used for testing (see Fig. 2). Then the results on test fold are averaged. Finally, the model is validated using validation data (20%).

To find the noisy samples/outliers, we need to set the optimal combinations of k_1, k_2 values, which depend on the nature of the data. Here k_1=kth smallest distance and k_2=|kRNNs|=number of reverse neighbors. A sample is called outlier if it has less number of reverse neighbors (k_2) than k_1. First, we need to fix k_1 and then increase the values of k_2 from 1 to k_1. As we increased the values of k_2 from 1 to k_1, the number of outliers increased. When there is a steep increase in the number of outliers over previous k_2, the particular previous k_2 can be considered. If we keep k_1=k_2, then there shall be more number of outliers or in other words more number of samples shall be labeled as outliers. For both the datasets, based on the above discussion, we chose k_1, k_2 by fine tuning them in the identification of number of outliers.

In the case of fraud detection data, we chose k_1=5 and k_2=3 and accordingly 4569 samples are identified as outliers. Then, the identified outliers are removed from the majority class in the training dataset and the residual majority class has 7028 samples. On this residual majority class, we employed one-class SVM (OCSVM) to extract the support vectors. We trained the OCSVM with the samples of reduced majority class and extracted support vectors. Consequently, the class distribution ratio of majority class versus minority class in this modified training dataset became almost perfectly balanced, i.e., 703 majority samples (non-fraudulent records – which are indeed support vectors) and 723 minority samples (fraudulent records, since 80% of minority samples are left untouched).

In the case of credit card churn prediction dataset, in order to remove the outliers, we applied kRNN with k_1, k_2 being chosen as 5 and respectively 1. Thus, we obtained 2336 outliers and on the residual 8713 samples, we employed OCSVM to extract the support vectors (720 in number) and derived the new training dataset by merging with minority class samples (80% of untouched minority class samples). Now, the training class is almost equally distributed i.e., 720 majority class samples and 783 minority class samples. For both the datasets, we chose k_1, k_2 based on the above discussion.

Vasu and Ravi (2011) proposed kRNN+K-Means algorithm for undersampling. The innovation in the proposed methodology lies in the use of OCSVM in place of K-Means Clustering algorithm, in order to perform the 2nd level of undersampling (after kRNN). This is distinctly advantageous over the use of K-Means because we extracted support vectors from OCSVM, which are very few in number and also they are the actual samples, forming a subset of the dataset that resulted from the application of the kRNN method. However, in the case of K-Means, the cluster centers, which are not the actual samples from the majority class dataset, are considered to accomplish under-sampling. Thus, in the proposed method, artificial samples, in the form of cluster centers, do not enter the modified dataset.

The classifiers are trained using this newly modified training dataset and tested with the test data. We performed the 10 fold cross validation method using the modified training dataset. Finally, the model is validated on the validation data (20%) which is unbalanced and represents the realistic scenario present in the data. The quantities employed to measure the quality of the classifiers are sensitivity, specificity, accuracy and Area Under ROC Curve (AUC), which are defined as follows (Fawcett and Provost, 1996). Sensitivity is the measure of proportion of the true positives (TP), which are correctly identified by a classifier.

$$Sensitivity = TP/(TP+FN)$$

Specificity is the measure of proportion of the true negatives (TN), which are correctly identified by a classifier.

$$Specificity = TN/(TN+FP)$$

Accuracy is the measure of proportion of true positives and true negatives, which are correctly identified.

$$Accuracy = (TP+TN)/(TP+TN+FP+FN)$$

where TP stands for *true positive*, TN stands for *true negative*, FP stands for *false positive*, FN stands for *false negative*. Further, we computed the AUC for each classifier and ranked the classifiers in the descending order of AUC (Ravi et al., 2007). Further, we represented the confusion matrix as well in Fig. 2.

5. Results and discussion

We performed 10-fold cross validation throughout the study and the average results obtained against validation data using original unbalanced data and the modified (balanced) data using proposed undersampling method are presented in Table 4. All classifiers using the proposed undersampling methodology outperformed their counterparts in the case of unbalanced dataset. Further, when compared the results of Table 3, our proposed novel methodology outperformed well in identifying fraudulent records. One can notice the variation in sensitivity when compared to previous results also. We employed the classifiers Decision Tree and Support Vector Machines (SVM) as implemented in the open source data mining tool Rapidminer (www.rapidminer.com). We employed Multi-Layer Perceptron, Group Method of Data Handling (GMDH) and Probabilistic Neural Network (PNN) is implemented in NeuroShell (www.neuroshell.com/). Logistic Regression is implemented in Weka embedded Knime (www.knime.org). For implementing OCSVM to extract support vectors for the majority class, we used libsvm module supported by OpenCV (http://docs.opencv.org/modules/ml/doc/support_vector_machines.html).

Various parameters are considered for different classifiers. In MLP, we chosen the important parameters for both the datasets such as *Activation function=sine*, *Scaling function=Tanh*, *Learning Rate=0.1*, *Momentum=0.01*. In DT, *confidence factor=0.5*, *0.1*, *minimum no of samples for leaf=12*, *minimum split size=20*, *maximal depth=30 and gain_ratio is chosen as criterion*, *Information Gain is chosen as criterion for Fraud detection dataset and Churn prediction dataset*. For SVM, *kernel type=anova, a statistical decomposition method for deriving kernal*, *Kernal gamma=1 and 5*, *kernel degree=1*, *cost parameter=0* and *convergence epsilon*=0.01, *maximum number of iterations=10,000* are chosen for Insurance Fraud and Churn prediction

Table 3
Comparison of the average results on Insurance validation dataset.

	Original unbalanced data (Vasu and Ravi, 2011)				Results for kRNN+k-means based under sampling approach (Vasu and Ravi, 2011)				Proposed methodology					
Tech*	Sen*	Spec*	Acc*	AUC*	Sen*	Spec*	Acc*	AUC*	Alg*	Sen*	Spec*	Acc*	AUC*	t-test#
LR	1	99.3	93.45	5015	67.03	69.66	69.5	6834.5	LR	90.12	58.89	60.81	7450.5	5.72
MLP	0	100	94.03	5000	83.78	60.93	62.3	7235.5	MLP	64.58	71.89	72.25	6819.5	5.36
DT	6.94	99.83	94.39	5338.5	**87.57**	56.48	58.35	7202.5	PNN	87.5	58.94	61.60	7322	5.35
SVM	**70.76**	63.2	63.65	6698	79.23	59.79	60.94	6951	**DT**	**90.74**	58.69	60.61	7471.5	**1.23**
GMDH	0	100	94.03	5000	72.43	63.62	64.15	6802.5	**SVM**	**91.89**	58.39	60.40	7514	-
GP	0	100	94.03	5000	72.43	63.62	64.15	6802.5	GMDH	56.86	80.20	78.16	6853	13.03

Tech*= technique, Sen*= sensitivity, Spec*=specificity, Acc*=accuracy and t-test#=w.r.t. Sens* of SVM, AUC*= area under ROC curve.

dataset. *Max no of variables in connections=3, max variable degree in connections=3, max variables in product team=3, selection criterion=full* and *model optimization=high* are chosen in GMDH for both datasets. For PNN, *Genetic pool size is set to 300, smoothing factor is chosen as 0.8*. For OCSVM, *nu value =0.1, γ=0.04, kernel function is sigmoid, coefficient of 10* is chosen and the parameter combination that is mentioned above is chosen after conducting experiments involving several combinations for fine tuning them and it yielded the highest sensitivity in both datasets.

Even though fraud occurrence rate is low, still it costs a huge loss to an organization. Therefore, many business decision makers place high emphasis on sensitivity alone thereby trying to minimize the losses and sustaining in the competition. In many real-life problems such as fraud detection in credit cards, fraud detection in telecom services, bankruptcy prediction, intrusion detection in computer networks, cancer detection in humans based on their genetic profiles, etc. detecting and predicting positive cases is accorded higher preference. Obviously, to misclassify a churner (fraudulent record in our case) is, on average, far more expensive than to misclassify a non-churner (non-fraudulent record) (Glady et al., 2009). At the same time, we accorded same priority in evaluating the classifiers based on receiver operating curve (AUC).

Consequently, in this study, sensitivity is accorded top priority ahead of specificity. Therefore, we discuss the performance of our proposed approach with respect to sensitivity alone. However, if management is interested in correctly predicting both fraudulent and genuine claims, then both sensitivity and specificity should be given equal priority. In other words AUC can be considered in ranking classifiers and sensitivity alone. From Table 3, it is observed that LR, MLP, RBF, Decision Tree, GMDH and PNN all yielded almost 0% sensitivity, 100% specificity and 94% accuracy on the original unbalanced insurance fraud detection data (Vasu and Ravi, 2011). This result is not surprising and the reason for the worst performance is the imbalance present in the dataset. However, SVM yielded considerably higher sensitivity of 70.76%, specificity of 63.2% and accuracy of 63.65% on original unbalanced data. From the results of SVM, we can observe that SVM is not highly affected by the class imbalance since the classification in SVM is accomplished by using the support vectors (which lie on the classification boundary) from majority class and minority class by ignoring the samples from both classes far away from the classification boundary.

Table 4
Average results of Farquad et al. (2012) on insurance validation dataset.

Classifiers	Sen*	Spec*	Acc*	AUC*
SVM	87.68	56.36	58.21	7202
ALBA	88.00	56.27	58.17	7213.5
ALBA (SVs)	88.43	55.13	57.19	7178
MALBA	88.22	55.64	57.59	7193
MALBA (normal)	88.00	55.74	57.67	7187
MALABA (logistic)	88.16	54.77	56.73	7146.5

Sen*= sensitivity, Spec*=specificity, Acc*=accuracy and AUC*=area under ROC curve.

A modified dataset with almost 50%:50% proportion of genuine and fraudulent claims are generated using the proposed hybrid under sampling approach. This modified data is used to train several classifiers and then performance of the classifiers is validated using validation data and results (validation set) are presented in Table 3. From Table 3, it is observed that by following proposed under sampling approach, classifiers LR, MLP, DT, SVM, GMDH and PNN yielded uniformly higher sensitivity compared to that of unbalanced case. It is observed that SVM yielded highest sensitivity of 91.89% and DT yielded 90.74% on the validation data which is unbalanced. If management is interested in correctly predicting both fraudulent and genuine claims, then both sensitivity and specificity should be given equal priority. Further, we performed *t*-test to see whether SVM's performance is statistically significant or not. We found that all other models except DT are statistically significantly different with respect to sensitivity at 18 degrees of freedom. Between DT and SVM, with respect to sensitivity, computed *t*-statistic value is 1.23, which indicates that there is no significant difference between DT and SVM. Therefore, any of them can be recommended to the management. Consequently, we recommend DT, as it is much faster to train and more importantly, it yields 'if–then' rules indicating the business knowledge extracted from the dataset.

In this context, we compare our results with that of the previous studies conducted on the same dataset. Phua et al. (2004) proposed a fraud detection method on the same dataset. The main objective of their research was to improve cost savings for which they used stacking–bagging approach. Since they did not report sensitivity, specificity and accuracy, our results are strictly not comparable to theirs. Secondly, Padmaja et al. (2007) also worked on the same dataset. However, quite surprisingly, they undersampled the minority class before applying SMOTE technique on it. We, however, do not subscribe to that line of thinking, since in our opinion, we should not undersample the minority class, how much ever noisy it may be as they are too rare to be removed. Kubat and Matwin (1997) also held the same opinion. Further, they adopted hold-out method whereas we evaluated our approach using 10 fold cross validation, which is more authentic. Thus, once again, our results cannot be compared with theirs.

When compared to Vasu and Ravi (2011), our approach yielded better sensitivity while maintaining the specificity and accuracy. We presented AUC results in Table 3 which reflects that we succeeded in identifying fraudulent records without drastic fall in other measures like specificity and accuracy. We further compared our results with that of Farquad et al. (2012), where they deployed SVM-Recursive Feature Elimination for feature selection and employed active learning methods for synthetic data generation. But they achieved a maximum sensitivity of only 88.16% (see Table 4). Our proposed methodology, with a sensitivity of 91.89 % using SVM, outperformed their method.

As regards rules, we have extracted just 4 rules using the decision tree as presented in Table 5. When compared with the rules obtained by Vasu and Ravi (2011) and Farquad et al. (2012), we obtained less number of rules that are completely different and without actually compromising on the sensitivity. This is a significant outcome of the study. Vasu and Ravi (2011) by applying

Table 5
Rules obtained by Decision Tree on Insurance dataset in the present study.

Rule#	Antecedents	Consequent
1	If **Basepolicy** is "`Liability`" then	Non Fraud
2	If **Basepolicy** is "`Collision or All Perils`" and **Fault** is "`Policy Holder`" then	Fraud
3	If **Basepolicy** is "`Collision or All Perils`" and **Fault** is "`Policy Holder or Third Party`" and **Deductible** is 400 and below then	Non Fraud
4	If **Basepolicy** is "`Collision or All Perils`" and **Fault** is "`Policy Holder or Third Party`" and **Deductible** is "`500 and above`" then	Fraud

Table 6
Attribute Information of the Churn prediction data used.

S.No.	Attribute name	Description
	Target	Target variable (churn or non-churn)
	CRED_T	Credit in month T
	CRED_T-1	Credit in month T-1
	CRED_T-2	Credit in month T-2
	NCC_T	Number of credit cards in months T
	NCC_T-1	Number of credit cards in months T-1
	NCC_T-2	Number of credit cards in months T-2
	INCOME	Customer's Income
	N_EDUC	Customer's educational level
	AGE	Customer's age
	SEX	Customer's sex
	E_CIV	Civilian status
	T_WEB_T	Number of web transaction in months T
	T_WEB_T-1	Number of web transaction in months T-1
	T_WEB_T-2	Number of web transaction in months T-2
	MAR_T	Customer's margin for the company in months T
	MAR_T-1	Customer's margin for the company in months T-1
	MAR_T-2	Customer's margin for the company in months T-2
	MAR_T-3	Customer's margin for the company in months T-3
	MAR_T-4	Customer's margin for the company in months T-4
	MAR_T-5	Customer's margin for the company in months T-5
	MAR_T-6	Customer's margin for the company in months T-6

decision tree achieved 83.6% sensitivity with 19 rules having the same root variable viz., "*Base Policy*". But they obtained additional variables like *V_Price*, *Accident Area*, *Gender and Past no of claims*, etc. Then, Farquad et al. (2012) by applying decision tree achieved 88.16% sensitivity with 12 rules having the root of "Policy Holder" and they got additional variables of *Vehicle Category, Age of Vehicle, Accident Area, Manufacturer, and Marital status.*

Further, in the process of validating the proposed methodology, we worked on Churn prediction dataset. We presented the results obtained by the proposed methodology in Table 7. When compared with Vasu and Ravi (2011), we observed GMDH, SVM, LR, DT yielded significant results with respect to sensitivity (83.1%, 87.7%, 83.45% and 91.2%) and AUC. Further, when we observed the AUC results of these classifiers, GMDH yielded superior results among the rest of the classifiers in the proposed methodology. We performed *t*-test as done earlier in the case of Insurance data. With respect to sensitivity of GMDH, we noticed that statistically there is no significant difference between SVM and LR. But as GMDH has higher AUC when compared with the other classifiers, we concluded that GMDH contributed significant outcome. Further, we extracted Decision rules for the Churn dataset and we observed that we obtained 10 rules yielding higher sensitivity (91.2%) when compared with Vasu and Ravi (2011), where they

Table 7
Comparison of the average results on Churn validation dataset.

	Original unbalanced data (Vasu and Ravi, 2011)				Results for *k*RNN+k-means based under sampling approach (Vasu and Ravi, 2011)				Proposed methodology					
Classifiers	Sen*	Spec*	Acc*	AUC*	Sen*	Spec*	Acc*	AUC*	Classifiers	Sen*	Spec*	Acc*	AUC*	t- test#
LR	3.15	99.73	93.21	5144	83	86.43	86.2	8471.5	LR	83.45	84.37	84.4	8391	**1.06**
MLP	0	100	93.25	5000	84.5	82.66	82.79	8358	MLP	25.5	97.32	93.09	6141	18.5
RBF	0	100	93.25	5000	80.5	77.16	77.39	7883	PNN	85.7	44.23	47.49	6496.5	**2.7**
DT	**61.5**	99.16	96.89	8033	**91**	79.08	79.89	8504	DT	91.2	69.65	71.02	8090.5	16.3
SVM	60.17	74.91	73.92	6754	83.5	71.88	72.66	7769	**SVM**	**87.7**	74.28	75.1	**8099**	**1.37**
GMDH	1.05	99.86	93.19	5045.5	83	83.31	83.294	8315.76	**GMDH**	**83.1**	**86.44**	86.28	**8477**	-

Sen*=sensitivity, Spec*=specificity, Acc*=accuracy and *t*-test#=w.r.t. Sens* of GMDH

Table 8
Average results of Farquad et al. (2012) on Churn validation dataset.

Classifiers	Sen*	Spec*	Acc*	AUC*
SVM	82.85	74.25	74.79	7855
ALBA	81.90	74.59	75.09	7825
ALBA (SVs)	83.05	75.46	75.97	7926
MALBA	82.35	76.44	76.84	7940
MALBA (normal)	82.30	76.36	76.76	7933
MALABA (logistic)	82.85	75.50	76.00	7918

Sen*= sensitivity, Spec*=specificity, Acc*=accuracy and AUC=area under ROC curve.

Table 9
Rules obtained by Decision Tree on the Churn dataset in the present study.

Rule#	Antecedents	Consequent
1	If (**CRED_T** ≤ 593.445 and **NCC_T** ≤ 0.500)	Churn
2	If(**CRED_T** ≤ 593.445 and **NCC_T** > 0.500 and **MAR_T-2** ≤ -0.005)	Churn
3	If(**CRED_T** ≤ 593.445 and **NCC_T** > 0.500 and **MAR_T-2** > -0.005 and **T_WEB_T-1** ≤ 2.500 and **MAR_T-4** ≤ 0.070)	Loyal
4	If(**CRED_T** ≤ 593.445 and **NCC_T** > 0.500 and **MAR_T-2** > -0.005 and **T_WEB_T-1** ≤ 2.500 and **MAR_T** <= 5.100)	Churn
5	If (**CRED_T** ≤ 593.445 and **NCC_T** > 0.500 and **MAR_T-2** > -0.005 and **T_WEB_T-1** ≤ 2.500 and **MAR_T** > 5.100)	Loyal
6	If (**CRED_T** ≤ 593.445 and **NCC_T** > 0.500 and **MAR_T-2** > -0.005 and **T_WEB_T-1** > 2.500)	Loyal
7	If (**CRED_T** > 593.445 and **CRED_T** ≤ 595.605 and **NCC_T** <= 0.500 and **MAR_T-2** ≤ 0.175)	Churn
8	If (**CRED_T** > 593.445 and **CRED_T** ≤ 595.605 and **NCC_T** <= 0.500 and **MAR_T-2** > 0.175)	Loyal
9	If(**CRED_T** > 593.445 and **CRED_T** ≤ 595.605 and **NCC_T** > 0.500)	Loyal
10	If(**CRED_T** > 595.605)	Churn

achieved 12 rules with 91% sensitivity. When compared with Farquad et al. (2012), the proposed methodology yielded significant results. This can be observed by comparing Tables 7 and 8. Further, rules extracted are presented in Table 9.

6. Conclusion and future work

In this paper, we proposed a novel hybrid approach for undersampling the majority class in largely skewed unbalanced datasets in order to improve the performance of classifiers. This paper demonstrates the significance of eliminating outliers (noisy) and redundant samples from the majority class in the case of highly skewed unbalanced datasets. In the proposed approach, we first applied *k*RNN to detect and remove the noise present in the form of outliers in the majority class. Later, we applied OCSVM for extracting support vectors in the majority class.

Finally, we combined these noisy and redundancy free majority class samples with the original minority samples and carried out experiments with the modified dataset thus obtained. We tested the effectiveness of our approach with different classifiers such as MLP, PNN, LR, SVM, DT, and GMDH using the validation dataset which is unbalanced and close to real environment. The effectiveness of the proposed hybrid approach was demonstrated on the dataset namely fraud detection dataset taken from Phua et al. (2004) in insurance sector. We achieved 90.74% and 91.89% fraudulent claims detection rate (sensitivity) on Insurance fraud detection data by DT and SVM respectively. Regarding Churn prediction dataset, GMDH yielded significant result of 83.1% sensitivity. The results show that using our hybrid undersampling approach, the classifiers performed better compared to when original unbalanced data was presented to them. Finally, the proposed methodology can be applied to other problems of intruder detection in computer networks, terminal diseases prediction in humans, default prediction in banks, etc. The scope can be further extended by determining whether the presence of *k*RNN has its influence on the proposed model. If not so, we feel that OCSVM can itself undersample the majority class. We would like to investigate further in that perspective.

References

Benard, C., Vanduffel, S., 2014. Mean-variance optimal portfolios in the presence of a benchmark with applications to fraud detection. Eur. J. Oper. Res. 234, 469–480.

Business Intelligence Cup, Organized by the University of Chile, downloaded in 2004 at: ⟨http://www.tis.cl/bicup04/text-bicup/BICUP/202004/20public/20data.zip⟩.

Bolton, R.J., Hand, D.J., 2002. Statistical fraud detection: a review. Stat. Sci. 17 (3), 235–249.

Brockett, P., Derrig, R., Golden, L., Levine, A., Alpert, M., 2002. Fraud classification using principal component analysis of RIDITs. J. Risk Insur. 69 (3), 341–371.

Chan, P.K., Fan, W., Prodromidis, A.L., Stolfo, S.J., 1999. Distributed data mining in credit card fraud detection. IEEE Intell. Syst. 14, 67–74.

Diamantini, C., Potena, D., 2009. Bayes vector quantizer for class-imbalance problem. IEEE Trans. Knowl. Data Eng. 21 (5), 638–651.

Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., 2002. SMOTE: synthetic minority oversampling technique. J. Artif. Intell. Res. 16 (1), 321–357.

Estevez, P.A., Held, C.M., Perez, C.A., 2006. Subscription fraud prevention in telecommunications using fuzzy rules and neural networks. Expert Syst. Appl. 31 (2), 337–344.

Fawcett, T., Provost, F., 1996. Combining data mining and machine learning for effective user profiling. In: Proceedings of First International Conference on Knowledge Discovery and Data Mining, pp. 8–13.

Farquad, M.A.H., Ravi, V., BapiRaju, S., 2012. Analytical CRM in banking and finance using SVM: a modified active learning-based rule extraction approach. Int. J. Electron. Cust. Relatsh. Manag. 6 (1), 48–73.

Ferreira, J.B., Vellasco, M., Pacheco, M.A., Barbosa, C.H., 2004. Data mining techniques on the evaluation of wireless churn. In: (ESANN'2004) Proceedings – European Symposium on Artificial Neural Networks Bruges (Belgium), d-sidepubli., ISBN 2-930307-04-8, pp. 483–488.

Glady, N., Baesens, B., Croux, C., 2009. Modelling churn using customer lifetime value. Eur. J. Oper. Res. 197 (1), 402–411.

Hart, P.E., 1968. The condensed nearest neighbor rule. IEEE Trans. Inf. Theory 14 (3), 515–516.

Hu, W., Liao, Y., Vemuri, V.R., 2003. Robust anomaly detection using support vector machines. In: Proceedings of the International conference on Machine Learning.

Ivakhnenko, A.G., 1968. The group method of data handling – a rival of the method of stochastic approximation. Sov. Autom. Control 13 (3), 43–55.

Japkowicz, N., 2003. Class Imbalances: Are We Focusing on the Right Issue?. ICML-KDD'2003. Workshop: Learning from Imbalanced Data Sets. Washington DC, USA.

Japkowicz, N., Stephen, S., 2002. The class imbalance problem: a systematic study. Intell. Data Anal. 6 (5), 429–450.

Japkowicz, N., 2000. The class imbalance problem: significance and strategies. In: Proceedings of the International Conference on Artificial Intelligence (IC-AI'2000). Las Vegas, Nevada, USA. pp. 111–117.

Jensen, D., 1997. Prospective assessment of AI technologies for fraud detection and risk management. In: Proceedings of the AAAI Workshop on AI Approaches to Fraud Detection and Risk Management, pp. 34–38.

Kirkos, E., Spathis, C., Manolopoulos, Y., 2007. Data mining techniques for the detection of fraudulent financial statements. Expert Syst. Appl. 32 (4), 995–1003.

Kubat, M., Matwin, S., 1997. Addressing the curse of imbalanced training sets: one sided selection. In: Proceedings of the Fourteenth International Conference on Machine Learning. Nashville, Tennessee, USA. pp. 179–186.

Kumar, D.A., Ravi, V., 2008. Predicting credit card customer churn in banks using data mining. Int. J. Data Anal. Tech. Strat. 1 (1), 4–28.

Laurikkala, J., 2001. Improving identification of difficult small classes by balancing class distribution. In: Proceedings of the 8th Conference on AI in Medicine. pp. 63–66.

Lang, M., Wambach, A., 2013. The fog of fraud – mitigating fraud by strategic ambiguity. J. Games Econ. Behav. 81, 255–275.

Larivie're, B., Van den Poel, D., 2004. Investigating the role of product features in preventing customer churn, by using survival analysis and choice modeling: the case of financial services. Expert Syst. Appl. 27 (2), 277–285.

Lee, Y.J., Yi-Ren, Y., Wang, Y.F., 2013. Anamoly detection via online oversampling principal component analysis. IEEE Trans. Knowl. Data Eng. 25 (7), 1460–1470.

Mozer, M.C., Wolniewicz, R., Grimes, D.B., Johnson, E., Kaushansky, H., 2000. Predicting subscriber dissatisfaction and improving retention in the wireless telecommunications industry. IEEE Trans. Neural Netw. 11 (3), 690–696.

Ngai, E.W.T, Hu, Y., Wong, Y.H., Chen, Y., Sun, X., 2011. The application of data mining techniques in financial fraud detection: a classification framework and an academic review of literature. Decis. Support Syst. 50 (3), 559–569.

Naveen, N., Ravi, V., Kumar, D.A., 2009. Application of fuzzy ARTMAP for churn prediction in bank credit cards. Int. J. Inf. Decis. Sci. 1 (4), 428–444.

Padmaja, T.M., Narendra, D., Raju, S.B., RadhaKrishna, P., 2007. Unbalanced data classification using extreme outlier elimination and sampling techniques for fraud detection. In: Proceedings of the 15th International Conference on Advanced Computing and Communications (ADCOM-07), pp. 511–516.

Perez, J.M., Muguerza, J., Arbelaitz, O., Gurrutxaga, I., Martin, J.I., 2005. Consolidated tree classifier learning in a car insurance fraud detection domain with class imbalance. In: Proceedings of the International Conference on Advances in Pattern Recognition, pp. 381–389.

Phua, C., Damminda, A., Lee, V., 2004. Minority report in fraud detection: classification of skewed data (Special Issue on Imbalanced Data Sets). SIGKDD Explor. 6 (1), 50–59.

Phua, C., Lee, V., Smith, K., Gayler R., 2007. comprehensive survey of data mining-based fraud detection research, http://arxiv.org/pdf/1009.6119&embedded=true&embed.

Quah, J.T.S., Sriganesh, M., 2008. Real-time credit card fraud detection using computational intelligence. Expert Syst. Appl. 35 (4), 1721–1732.

Ravi, V., Ravikumar, P., Srinivas, E., Kasabov, N.K., 2007. A semi-online training algorithm for the radial basis function neural networks: Applications to bankruptcy prediction in banks. In: V., Ravi (Ed.), Advances in Banking Technology and Management: Impact of ICT and CRM. IGI Global Inc., USA.

Rupnik, R., Kukar, M., Krisper, M., 2007. Integrating data mining and decision support through data mining based decision support system. J. Comput. Inf. Syst. 47 (3), 89–104.

Sanchez, D., Vila, M.A., Cerda, L., Serrano, J.M., 2009. Association rules applied to credit card fraud detection. Expert Syst. Appl. 36 (2), 3630–3640.

Smith, K.A., Gupta, J.N.D., 2000. Neural networks in business: techniques and applications for the operations researcher. Comput. Oper. Res. 27 (11–12), 1023–1044.

Sublej, L, Furlan, S., Bajec, M., 2011. An expert system for detecting automobile insurance fraud using social network analysis. Expert Syst. Appl. 38 (1), 1039–1052.

Soujanya, V., Satyanarayana, R.V., Kamalakar, K., 2006. A simple yet effective data clustering algorithm. In: Proceedings of the Sixth International Conference on Data Mining (ICDM'06). Hong Kong, pp. 1108–1112.

Specht, D.A., 1990. Probabilistic neural networks. Neural Netw. 3 (10), 109–118.

Srinivasan, D., 2008. Energy demand prediction using GMDH networks. Neurocomputing 72 (1–3), 625–629.

Stefano, B., Gisella, F., 2001. Insurance fraud evaluation: a fuzzy expert system. In: Proceedings of 10th IEEE International Conference Fuzzy Systems, vol. 3, pp. 1491–1494.

Stolfo, S.J., Fan, D.W., Lee, W., Prodromidis, A.L., Chan, P., 2000. Cost-based modeling for fraud and intrusion detection: results from the JAM project. In: Proceedings of the DARPA Information Survivability Conference and Exposition (DISCEX'2000), vol. 2, pp. 130–144.

Stolfo, S.J., Prodromidis, A.L., Tselepis, S., Lee, W., Fan, D.W., 1997a. JAM: Java agents for meta-learning over distributed databases. AAAI Workshop on AI Approaches to Fraud Detection. In: Proceedings of the 3rd International Conference Knowledge Discovery and Data Mining, pp. 74–81.

Stolfo, S.J., Fan, D.W., Lee, W., Prodromidis, A.L., 1997b. Credit card fraud detection using meta-learning: issues and initial results. AAAI Workshop on AI Approaches to Fraud Detection and Risk Management, pp. 83–90.

Tax, D.M.J., Duin, R.P.W., 2004. Support vector data description. Mach. Learn. 54, 45–66.

Vapnik, V.N., 1998. Statistical Learning Theory. John Wiley & Sons, New York.

Vasu, M., Ravi, V., 2011. A hybrid undersampling approach for mining unbalanced datasets: application to banking and insurance. Int. J. Data Min. Model. Manag. 3 (1), 75–105.

Viaene, S., Dedene, G., Derrig, R.A., 2005. Auto claim fraud detection using bayesian learning neural networks. Expert Syst. Appl. 29 (3), 653–666.

Viaene, S., Derrig, R.A., Baesens, B., Dedene, G., 2002. A comparison of state-of the-art classification techniques for expert automobile insurance claim fraud detection. J. Risk Insur. 69 (3), 373–421.

Xu, W., Wang, S., Zhang, D., Yang, B., 2011. Random rough subspace based neural network ensemble for insurance fraud detection. In: Proceedings of the IEEE International Joint Conference on Computational Sciences and Optimization, pp. 1276–1280.

Yang, W.S., Hwang, S.Y., 2006. A process-mining framework for the detection of healthcare fraud and abuse. Expert Syst. Appl. 31 (1), 56–68.

Zhu. S., Wang, Y., Wu, Y., 2011. Health care fraud detection using non-negative matrix factorization. In: Proceedings of the IEEE International Conference on Computer Science and Education, pp. 499–503.