Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IMPACT FACTOR: 5.258

IJCSMC, Vol. 5, Issue. 7, July 2016, pg.344 – 352

An Efficient Approach for Minority Class Oversampling of Imbalanced Dataset

Mr. Dasharath C. Magar[1], Prof. S.M. Rokade[2]

[1]Department of Computer Engineering, Savitribai Phule Pune University, India

[2]Department of Computer Engineering, Savitribai Phule Pune University, India

[1] magardasharat@yahoo.com; [2] smrokade@yahoo.com

Abstract— *In the dataset the data is stored in terms of majority class and minority class form. The class which contains more data samples is called majority class and the class which contains less data samples is called minority class. Such dataset is called imbalanced dataset. When the dataset contains an unequal distribution of data samples among different classes then it becomes a big challenge to any classifier as it becomes hard to learn the minority class samples. A Majority Weighted Minority Oversampling Technique(MWMOT) method address this problem by adding the synthetic minority class samples in minority class to balance the majority and minority classes. Thus dataset becomes balance. Then the accuracy of classifier is increased. The classifier classifies data sample by referring the majority class samples, resulting in a large classification error over the minority class samples. The MWMOT is Synthetic oversampling method that adds samples by generating the synthetic minority class samples. The results shows that MWMOT method is better than or comparable with some other existing methods in terms of various assessment metrics, such as Kappa Statistics, Correctly Classified, Incorrectly Classified, Mean Absolute Errors. Square root of Mean Absolute Error.*

Keywords— *classifier, clustering, Dataset, Imbalanced, oversampling, synthetic sample*.

I. Introduction

The basic aim of any classifier is to reduce its classification errors and to maximize its overall accuracy. But an imbalanced Dataset is a great challenge to the classifier as it becomes very difficult to refer the minority class samples. It is because the classifier learned from the imbalanced data tends to favour the majority class samples, resulting in a large classification error over the minority class samples. This becomes very costly when identification of the minority class samples is crucial. Thus, the classifier learned from the imbalanced data needs to perform equally well both on the minority class and the majority class samples.

Imbalance that exists between the samples of two classes is usually known as between-class imbalance. The actual cause for the bad performance of conventional classifiers on the minority class samples is not necessarily related to only on the between-class imbalance. Classifiers' performance have been found to depreciate in the presence of within-class imbalance and small disjunct problems. Besides, the complexity of data samples is another factor for the classifiers' poor performance. If the samples of the majority and minority classes have more than one concepts in which some concepts are rarer than others and the regions between some concepts of different classes overlap, then the imbalance Dataset becomes very severe .

The most popular approaches to deal with imbalanced Dataset are based on the synthetic oversampling methods. A novel synthetic oversampling method, i.e., Majority Weighted Minority Oversampling TEchnique (MWMOTE), whose goal is to alleviate the problems of imbalanced Dataset and generate the useful synthetic minority class samples.

The main task of classifier is to minimize classification error, i.e., to improve its accuracy However, the imbalance data set is a great challenge to the classifier as it becomes very hard to learn the minority class samples It is because the classifier learned from the imbalanced data gives more wattage to the majority class samples, resulting in a large classification error over the minority class samples .

Thus, the classifier need to perform equally well both on the minority class and the majority class samples. However, imbalanced data set pose a great challenge to the classifier as it becomes very hard to learn the minority class samples .

It is because the classifier learned from the imbalanced data refers the majority class samples, resulting in a large classification error over the minority class samples . This becomes very expensive when identification of the minority class samples is compulsory.

Thus, the classifier needs to perform equally well both on the minority class and the majority class samples. So it is necessary to balance the data set.

II. LITERATURE SURVEY

This paper deals with the discussion of the numerous papers developed at various institutes which give us idea about this topic. The literature balancing Dataset is very broad and numerous papers have been published in different institutes. The purpose of Oversampling is balancing dataset and to minimize classification errors of different classifiers,

Several algorithms exists, and this worksheet focuses on a particular one developed by Sukarna Barua, Md. Monirul Islam, Xin Yao, Fellow, IEEE, and Kazuyuki Murase in 2014[1]. They had presented a new approach for Imbalanced Dataset. Imbalanced learning problems contain an unequal distribution of data samples among different classes and pose a challenge to any classifier as it becomes hard to learn the minority class samples. Synthetic oversampling methods address this problem by generating the synthetic minority class samples to balance the distribution between the samples of the majority and minority classes. This paper identifies that most of the existing oversampling methods may generate the wrong synthetic minority samples in some scenarios and make learning tasks harder. To this end, a new method, called Majority Weighted Minority Oversampling TEchnique (MWMOTE), is presented for efficiently handling imbalanced learning problems. MWMOTE 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. It then generates the synthetic samples from the weighted informative minority class samples using a clustering approach. This is done in such a way that all the generated samples lie inside some minority class cluster. MWMOTE has been evaluated extensively on four artificial and 20 real-world data sets.

Yanping Yang, Guangzhi M Suggested an ensemble-based active learning algorithm to address the class imbalance problem. The artificial data are created ac-cording to the distribution of the training dataset to make the ensemble diverse, and the random sub-space re-sampling method is used to reduce the data dimension. In selecting member classifiers based on misclassification cost estimation, the minority class is assigned with higher weights for misclassification costs, while each testing sample has a vari-able penalty factor to induce the ensemble to cor-rect current error. In our experiments with UCI disease datasets, instead of classification accuracy, F-value and G-means are used as the evaluation rule. Compared with other ensemble methods, their method shows best performance[2]

Cieslak and Chawla proposed a cluster-based algorithm, called local sampling, in which the Hellinger distance measure is used first for partitioning the original data set. A sampling method is then applied to each partition and finally data of all partitions are merged to create the new data set[3].

Naheed Azeem, Shazia Usmani proposed different data mining techniques for identifying fault prone modules as well as compare the data mining algorithms to find out the best algorithm for defect prediction[4].

Miss Reshma K. Dhurjad Prof. Mr. S. S. Banait Proposed sampling techniques that have been shown to be very successful in recent years. To address imbalanced learning issue oversampling of minority class is done. There are various Oversampling techniques which can be used to re-establish the class balance. Oversampling method is a data level method. The main advantage of data level methods is that they are self-sufficient. The methods at data level modify the distribution of the imbalanced datasets, and then these modified i.e. balanced datasets are provided to the algorithm to improve the Imbalanced learning[5].

Date Shital Maruti proposed a method that finds minority samples which are difficult to learn and computes Euclidean distance between nearest majority class samples. Using clustering approach and weighted minority class samples it generates synthetic samples for oversampling purpose. Proposed work will evaluate this approach on real & artificial datasets[6].

Kehan Gao , Taghi M, Khoshgoftaar and Randall Wald employ two sampling techniques, random undersampling (rus) and synthetic minority oversampling technique (smote), and two ensemble boosting approaches, rusboost and smoteboost (in which rus and smote, respectively, are integrated into a boosting technique), as well as six feature ranking techniques. They apply the proposed techniques to several groups of datasets from two real-world software systems and use two learners[7]

Talayeh Razzaghi introduced a cost-sensitive learning method (CSL) to deal with the classification of imperfect data. Typically, most traditional approaches for classification demonstrate poor performance in an environment with imperfect data. They propose the use of CSL with Support Vector Machine, which is a well known data mining algorithm. The results reveal that the proposed algorithm produces more accurate classifiers and is more robust with respect to imperfect data. Furthermore, they explore the best performance measures to tackle imperfect data along with addressing real problems in quality control and business analytics[8].

Hong Cao and Xiao-Li Li proposes a novel Integrated Oversampling (INOS) method that can handle highly imbalanced time series classification. They introduce an enhanced structure preserving oversampling (ESPO) technique and synergistically combine it with interpolation-based oversampling. ESPO is used to generate a large percentage of the synthetic minority samples based on multivariate Gaussian distribution, by estimating the covariance structure of the minority-class samples and by regularizing the unreliable Eigen spectrum. To protect the key original minority samples, They used an interpolation-based technique to oversample a small percentage of synthetic population[9].

III. IMPLEMENTATION STRATEGY

The MWMOT collects nearest more informative samples from majority class as well as some neighbourhood samples from minority class, then assigns weights to data samples. Then the synthetic samples are generated and using clustering methods the samples are combined and added in minority class. This makes dataset balanced. Using classifiers the accuracy of classifier is tested and measured using various dataset.

A. System Architecture

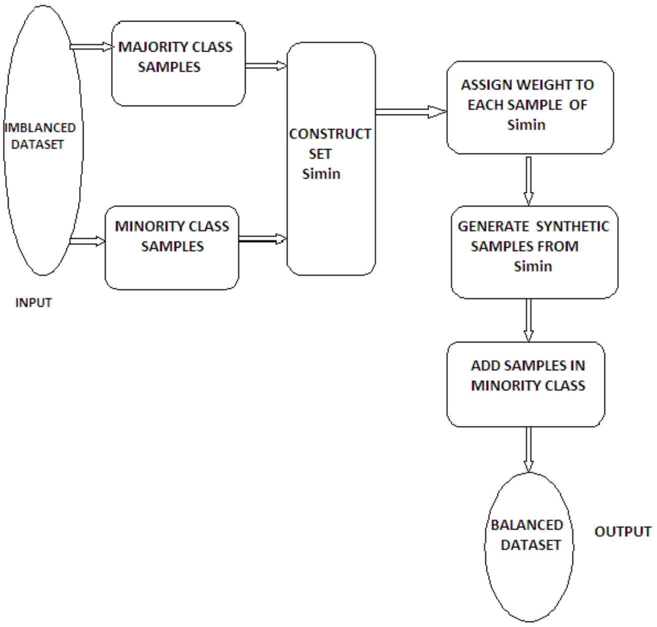


Figure 3. 1: System Architecture of MWMOT

The system architecture is divided into following three phases:-

1. **Phase 1:** Construct a set S_{imin}
2. **Phase 2:** For each member of $_{Simin}$ is given a selection weight, $_{Sw}$,
3. **Phase 3:** Generates the synthetic samples from S_{imin} using S_ws and produces the output

1. **Phase 1:** Construct a set S_{imin},

   In the first phase, MWMOT identifies the most important and hard-to-learn minority class samples from the original minority set, S_{min} and construct a set, S_{imin}, by the identified samples.

2. **Phase 2:** For each member of S_{imin} is given a selection weight, S_{w},

In the second phase, each member of S_{imin} is given a selection weight, S_w, according to its importance in the data. Following formula is used to assign weight to minority class samples.

$$S_w(x_i) = \sum_{y_i \in S_{bmaj}} I_w(y_i, x_i).$$

Where

Where

$$I_w(y_i, x_i) = C_f(y_i, x_i) \times D_f(y_i, x_i).$$

$$C_f(y_i, x_i) = \frac{f\left(\frac{1}{d_n(y_i, x_i)}\right)}{C_f(th)} * CMAX,$$

,

$$d_n(y_i, x_i) = \frac{dist(y_i, x_i)}{l},$$

where Cf and CMAX are the user defined parameters and f is a cut-off function., l is the dimension of future space.

3. **Phase 3:** Generates the synthetic samples from S_{imin} using S_ws and produces the output. In the third phase, MWMOT generates the synthetic samples from S_{imin} using S_ws and produces the output set S_{omin}

B. Methodology

Module 1: Classifier Test using ID3 and Naive Bayes classifiers.
Input:-Imbalanced Dataset
Output:-Calculation and measurement of Assessment matrices.

Module 2 : Run MWMOT Algorithm to balance Dataset.
Input:-Imbalanced Dataset
Output:- Balanced Dataset.

Module 3: Classifier Test using ID3 and Naive Bayes classifiers
Input:-Balanced Datasets
Output:-Calculation and measurement of Assessment metrics

C. Algorithm

Algorithm for selecting minority class samples with reference to majority class .

Algorithm MWMOT(S_{maj},S_{min},N,k1,k2,k3)
INPUT:-

S_{maj} :Set of majority class samples

S_{min} :Set of minority class samples

N : Number of synthetic samples to be generated

k1 : Number of neighbours used for predicting noisy minority class samples

k2 : Number of majority neighbours used for constructing informative minority set

k3 : Number of minority neighbours used for constructing informative minority set

M : No of clusters.

OUTPUT:-

Somin : Oversampled minority set

1) For each minority example xi € Smin, compute the nearest neighbour set, NN(xi).NN(xi) consists of the nearest k1 neighbours of xi according to Euclidean distance.
2) Construct filtered minority set, S_{minf} by removing those minority class samples which have no minority example in their neighbourhood:
S_{minf}= S_{min} –{xi € S_{min}: NN(xi) contains no minority example}
3) For each xi € S_{minf}, compute the nearest majority set, N_{maj}(xi). N_{maj}(xi) consists of the nearest k2 majority samples from xi according to Euclidean distance.
4) Find the borderline majority set, S_{bmaj}, as the union of all N_{maj}(xi)s, i.e.,
S_{bmaj} = Ụ$_{\text{xi Sminf}}$ N_{maj}(xi)

5) For each majority example yi € S_{bmaj}, compute the nearest minority set, N_{min} (yi). N_{min} (yi) consists of the nearest k3 minority examples from yi according to Euclidean distance.

6) Find the informative minority set, S_{imin}, as the union of all N_{min}(yi)s, i.e., S_{imin}= Ụ$_{\text{yi € Sbmaj}}$ N_{min}(yi)
7) For each yi € S_{bmaj} and for each xi ϵ S_{imin}, compute the information weight, I_w(yi,xi).
8) For each xi € S_{imin}, compute the selection weight S_w(xi) as
S_w(xi)=∑$_{\text{yiϵSbmaj}}$ I_w(yi,xi)

9) Convert each S_w(xi) into selection probability S_p(xi) according to
S_p(xi)=S_w (xi)/∑$_{\text{zi €Simin}}$ S_w(zi)

10) Find the clusters of S_{min}. Let, M clusters are formed which are L1,L2,...,L_M.

11) Initialize the set, S_{omin} =S_{min}.

12) Do for j = 1...N.
a) Select a sample x from S_{imin} according to probability distribution {S_p(xi)}. Let, x is a member of the cluster L_k,1 ≤k ≤M.
b) Select another sample y,at random,from the members of the cluster L_k.
c) Generate one synthetic data, s, according to s = x +α(y –x), where α is a random number in the range [0,1].
d) Add s to S_{omin}: S_{omin} = S_{omin} Ụ{s}.

13) End Loop

Algorithm for selecting neighbourhood minority class samples with reference to more informative minority class samples.

Algorithm Contribution(T, N, k)

INPUT:
T:Number of minority class samples
N: Number of nearest neighbours of k .
K; User defined Parameter

OUTPUT: (N/100)* T synthetic minority class samples .

1. // If N is less than 100%, randomize the minority class samples as only a random percent.
2. if N < 100
3. then Randomize the T minority class samples .
4. T = (N/100) ∗ T
5. N = 100
6. end if
7. N = (int)(N/100) // The amount of SMOTE is assumed to be in integral multiples of 100.
8. k = Number of nearest neighbours
9. numattrs = Number of attributes
10. Sample[ ][ ]: array for original minority class samples

```
newindex: keeps a count of number of synthetic samples generated, initialized to 0
Synthetic[ ][ ]: array for synthetic samples   // Compute k nearest neighbors for each minority class sample only.
for i ← 1 to T
    Compute k nearest neighbours for i, and save the indices in the nnarray
    Populate(N, i, nnarray)
for  Populate(N, i, nnarray)   // Function to generate the synthetic samples.
    while N not equalto 0
        //Choose a random number between 1 and k, call it nn. This step chooses one of the k nearest neighbors of i.
        for attr ← 1 to numattrs
            Compute: dif = Sample[nnarray[nn]][attr] − Sample[i][attr]
            Compute: gap = random number between 0 and 1
            Synthetic[newindex][attr] = Sample[i][attr] + gap ∗dif
        end for loop
        newindex++
        N = N − 1
    End while loop
    return   // End of Populate.
```

IV. EXPERIMENT AND RESULTS

In this section the effectiveness of proposed method-MWMOT is evaluated by calculating following assessment matrices.

- **Kappa Statistics(KS)**

('x' Label Predicted by classifier1 & classifier 2)+('Y' Label Predicted by classifier1 & classifier 2) / Total no. of Instances

- **Correctly Classified accuracy (CC)**

No of correctly label predicted / Total no of known label

- **Incorrectly Classified (IC)**

Total Instances - Correctly Classified Instance

- **Mean Absolute Error(MAE)**

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|f_i - y_i| = \frac{1}{n}\sum_{i=1}^{n}|e_i|$$

where f is known as Label Instance ,yi is Predicted as Label Instances, n is Total no of Instances

- **RMSE**

Square-root of(MAE) .

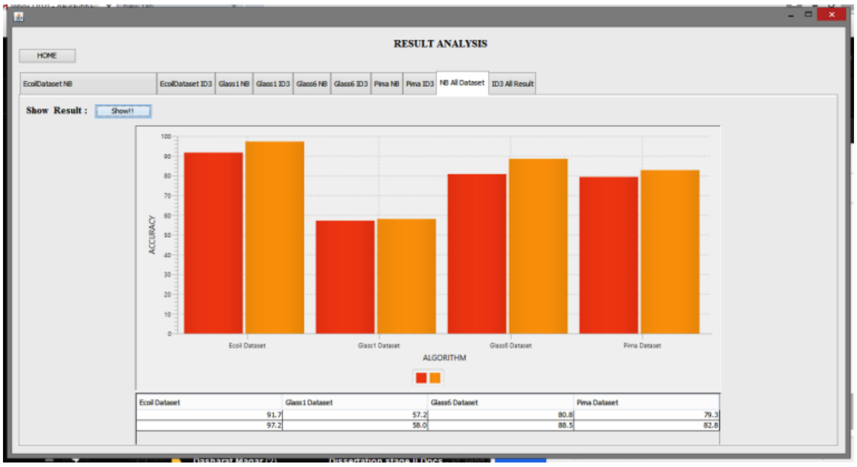
The classification accuracy of ID3 and Naïve Bayes classifiers is calculated before and after balancing different datasets. It has been observer that ,as shown in figure 4.1, figure 4.2 and Table 3,the accuracy of classifier is increased after implementing the oversampling i.e. after balancing dataset**.** Table 1 and Table 2 shows the performance measure for balancing dataset.

Table 1: Performance measures using Naïve Bayes

Dataset	Performance measures Before MWMOT(%)						
	CC	IC	MAE	RMSE	RAE	RRSE	KS
Ecoil	33-91	3-8.3	0.10	0.23	13.58	31.75	0.53
Performance measures After MWMOT							
Ecoil	**35-97**	1-2	0.04	0.17	8.29	34.31	**0.78**

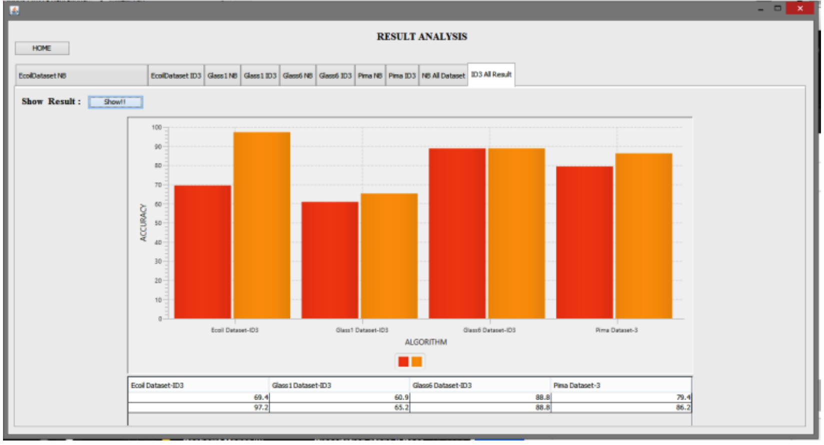
Table 2: Performance measures using ID3

Dataset	Performance measures Before MWMOT(%)						
	CC	IC	MAE	RMSE	RAE	RRSE	KS
Ecoil	25-69	11-30	0.25	0.39	34	53	0.18
Performance measures After MWMOT							
Ecoil	**45-96**	1-2	0.05	0.16	11.90	33	**0.68**



Ecoil Dataset	Glass1 Dataset	Glass6 Dataset	Pima Dataset
91.7	57.2	80.8	79.3
97.2	58.0	88.5	82.8

Figure 4.1: Comparison of Accuracy of NB All Dataset



Ecoil Dataset-ID3	Glass1 Dataset-ID3	Glass6 Dataset-ID3	Pima Dataset-3
69.4	60.9	88.8	79.4
97.2	65.2	88.8	86.2

Figure 4.2: Comparison of Accuracy of ID3 All Dataset

Result Comparison Table 3

Dataset	Accuracy Before MWMOT	Accuracy After MWMOT
Ecoil NB	91.7	97.2
Ecoil ID3	69.4	97.2
Glass 1 NB	57.2	58.0
Glass 1 ID3	60.9	65.2
Glass 6 NB	80.8	88.5
Glass 6 ID3	88.8	88.8
Pima NB	79.3	82.8
Pima ID3	79.4	86.2

V. CONCLUSION AND FUTURE SCOPE

Many oversampling methods exist in the literature for imbalanced dataset. These methods generate the synthetic minority class samples from the hard-to-learn minority class samples with an aim to balance the distribution between the samples of the majority and minority classes. However, in many conditions, existing methods are not able to select the hard-to-learn minority class samples effectively, assign relative weights to the selected samples appropriately, and generate synthetic samples correctly . Based on these observations, the proposed new method, i.e., MWMOT for imbalance dataset. The method not only selects the hard-to learn minority class samples effectively but also assigns them weights appropriately. Furthermore, it is able to generate correct synthetic samples.

MWMOT uses the majority class samples near the decision boundary to effectively select the hard-to-learn minority class samples. It then adaptively assigns the weights to the selected samples according to their importance in learning. The samples closer to decision boundary are given higher weights than others. Similarly, the samples of the small-sized clusters are given higher weights for reducing within-class imbalance. The synthetic sample generation technique of MWMOT uses a clustering approach. The aim of using clustering is to ensure that the generated samples must reside inside the minority class area for avoiding any wrong or noisy synthetic sample generation.

Thus the classification accuracy of classifiers like ID3 and Naive Bayse is calculated , measured using the two types of dataset i.e.NB and ID3 and represented using graph. It has been observed that the classification accuracy of classifier is increased after MWMOT is implemented

The system uses two class dataset and using oversampling it adds data samples in minority class. Thus the system balances two class dataset. But the same system can be useful in multiclass dataset.

ACKNOWLEDGEMENT

I am glad to express our sentiments of gratitude to all who rendered their valuable guidance to me. I would like to express my appreciation and thanks to Prof. Dr. S.A. Patil, Principal, SVIT College of Engg., Chincholi, Nashik. I am also thankful to Prof. S. M. Rokade, Head of Computer Engineering Department, SVIT College of Engg., Chincholi, Nashik. I thank the anonymous reviewers for their comments.

REFERENCES

[1]. Sukarna Barua, Md. Monirul Islam, Xin Yao, Fellow, IEEE, and Kazuyuki Murase," MWMOTE—Majority Weighted Minority Oversampling Technique for Imbalanced Data Set Learning".

[2] Ensemble-based active learning for class imbalance problem Yanping Yang, Guangzhi Ma

[3] Combating Imbalance in Network Intrusion Datasets David A Cieslak, Nitesh V Chawla, Aaron Striegel

[4] Analysis of Data Mining Based Software Defect Prediction Techniques By Naheed Azeem, Shazia Usmani

[5]A survey on Oversampling Techniques for Imbalanced Learning By Miss Reshma K. Dhurjad And Prof. Mr. S. S. Banait

[6]Minority Oversampling Technique for Imbalanced Data by Date Shital Maruti

[7]Combining Feature Selection and Ensemble Learning for Software Quality Estimation Kehan Gao , Taghi M, Khoshgoftaar and Randall Wald

[8] Cost-sensetive Learning –Based Methods for Imbalanced Classification Problems with Applications, Talayeh Razzaghi

[9] Integrated Oversampling for Imbalanced Time Series Classification, Hong Cao and Xiao-Li Li

[10] P.M. M urphy and D.W. Aha, "UCI Repository of Machine Learning Databases," Dept. of Information and Computer Science, Univ. of California, Irvine, CA, 1994.

[11] D. Lewis and J. Catlett, "Heterogeneous Uncertainty Sampling for Supervised Learning," Proc. Int'l Conf. Machine Learning, pp. 148-156, 1994.

[12] T.E. Fawcett and F. Provost, "Adaptive Fraud Detection," Data Mining and Knowledge Discovery, vol. 3, no. 1, pp. 291-316, 1997.

[13] M. Kubat, R.C. Holte, and S. Matwin, "Machine Learning for the Detection of Oil Spills in Satellite Radar Images," Machine Learning, vol. 30, no. 2/3, pp. 195-215, 1998.

[14] C.X. Ling and C. Li, "Data Mining for Direct Marketing: Problems and Solutions," Proc. Int'l Conf. Knowledge Discovery and Data Mining, pp. 73-79, 1998.

[15] N. Japkowicz, C. Myers, and M. Gluck, "A Novelty Detection Approach to Classification," Proc. 14th Joint Conf. Artificial Intelligence, pp. 518- 523, 1995.

[16] S. Clearwater and E. Stern, "A Rule-Learning Program in High Energy Physics Event Classification," Computer Physics Comm., vol. 67, no. 2, pp. 159-182, 1991.