Geometric Mean based Boosting Algorithm to Resolve Data Imbalance Problem

Myoung-Jong Kim
Department of Business Administration,
School of Business, Pusan National University
Busan, South Korea
mjongkim@pusan.ac.kr

***Abstract*—Data imbalance problem has received a lot of attention in machine learning community because it is one of the causes that degrade the performance of classifiers or predictors. In this paper, we propose geometric mean based boosting algorithm (GM-Boost) to resolve the data imbalance problem. GM-Boost enables learning with consideration of both majority and minority classes because it uses the geometric mean of both classes in error rate and accuracy calculation. We have applied GM-Boost to bankruptcy prediction task. The results indicate that GM-Boost has the advantages of high prediction power and robust learning capability in imbalanced data as well as balanced data distribution.**

Keywords - data imbalance; GM-Boost; bankruptcy prediction

I. INTRODUCTION

Data imbalance problem is frequently observed in various classification and prediction tasks when most of training samples belong to one majority class. Data imbalance problem is reported in a wide range of classification tasks, such as oil spill detection [16], response modeling [23], remote sensing [1], scene classification [27], card fraud detection [9] and credit rating [18].

Data imbalance problem could be one of the main causes that degrade the performance of machine learning algorithms in classification tasks. There are two main reasons why data imbalance causes degradation in performance of machine learning algorithms [13,14,24]. The first reason is associated with the objective function of classification algorithms. One of widely used objective functions for classification algorithms is the arithmetic mean based accuracy (hereafter, arithmetic accuracy) which is a ratio of the number of correctly classified instances over the number of total instances. However, in the presence of data imbalance, arithmetic accuracy can be inappropriate because the accuracy is highly dependent on the classification accuracy of majority class samples. More specifically, in very imbalanced domains, most standard classifiers will tend to learn how to predict the majority class. While these classifiers can obtain higher predictive accuracies than those that also try to consider the minority class more, this seemingly good performance can be argued as being meaningless [24].

The second reason for the degradation in performance is the distortion of decision boundaries resulting from imbalanced distribution of the classes. As the imbalance of data is getting severe, the decision (classification) boundary of majority class tends to invade the decision boundary of the minority class, so that the decision boundary of majority class is gradually expanded while the decision boundary of minority class is gradually reduced. This problem eventually causes the decrease in the accuracy for minority class.

For the alternatives to solve this problem, various methods have been proposed including under-sampling, over-sampling, cost adaptive strategies, and boosting algorithms. Recently, various boosting algorithms have been proposed as alternatives for data imbalance problems including SMOTEBoost [3] and RUSBoost [22]. In particular, SMOTEBoost is an application of boosting techniques to over-sampled data generated by synthetic minority over-sampling technique (SMOTE) [2]. SMOTE effectively creates a new minority samples, while boosting algorithm proceeds training on over-sampled data through repetitive sampling process which focuses on misclassified observations. In this way, SMOTEBoost can reinforce the training over samples from minority class to be likely misclassified. However, the boosting algorithm can be inappropriate as for over-fitting problem because its objective function is still measured in terms of arithmetic accuracy and arithmetic errors. New minority class samples, which are generated from SMOTE, are likely to have the higher similarity than majority data samples. Most standard learning algorithms will tend to generate classifiers focusing on samples with higher similarity because that strategy is helpful to maximize the objective function, i.e. arithmetic accuracy. This drawback might increase generalization errors when classifiers are applied to new validation data set which is not trained.

This paper proposes geometric mean based boosting (GM-Boost) which is a novel boosting algorithm applying the concept of geometric accuracy to AdaBoost algorithm [10]. It has the advantage of enabling balanced learning against both majority and minority classes. The proposed GM-Boost algorithm is applied to bankruptcy prediction task which is one of the typical data imbalance problems in business domains. Two different data samples, imbalanced data and balanced data samples, are constructed to verify the performance of GM-Boost algorithm.

Experimental results show that GM-Boost has the advantages of high prediction power and robust learning capability in imbalanced data distribution as well as in balanced data distribution.

II. Data Imbalance Problem in Binary classification problems

A. Data Imbalance Problem

Kang and Cho [13] constructed six sample groups according to different data balance rates (1:1, 1:3, 1:5, 1:10, 1:30, and 1:50) in order to analyze the effects of data imbalance on classification accuracy of SVM. From their experimental results, it can be seen that, for the two sample groups with little or no data imbalance problem (1:1, and 1:3), the sizes of classification boundary areas of the two classes are similar to each other. However, for the sample groups with serious data imbalance problems (1:5, and 1:10), the area of minority class is reduced because the area of the majority class invades the area of minority class, and thus the classification accuracy for minority class samples is degraded. Especially, for the sample groups with extreme data imbalance (1:30, and 1:50), it is reported that the classification boundary area for minority class is excessively small, which makes the classification for minority class meaningless. Also, they report that, as the data imbalance is getting severe, arithmetic accuracy over total samples steadily increases due to the high accuracy over samples of majority class, while the arithmetic accuracy for minority class is dramatically reduced, and thereby geometric accuracy over total samples gradually decreases. They argue that these results demonstrate that arithmetic accuracy is not a suitable objective function for imbalanced data.

Wu and Chang [24] assert two following results as a cause of skewed boundaries of SVM due to data imbalance. Firstly, data imbalance problem causes a tendency that samples of minority class do not reside in the boundary area of minority class. Secondly, as the data imbalance is getting severe, boundary area of majority class is expanded and boundary area of minority class is reduced due to the imbalance of support vectors. This problem causes the distortion of boundary area. Consequently, the possibility becomes very high that the classifier will classify a sample as a majority class.

B. The Approaches to Resolve Performance Measure Problem

Arithmetic accuracy is a proper performance measure for classifiers in balanced data set. However, under data imbalance, it is not a proper performance measure anymore because it is highly influenced by the classification accuracy of majority class [12, 13, 23]. Geometric accuracy and ROC analysis are proposed to resolve this problem. The geometric accuracy is calculated as a square root of sensitivity multiplied by specificity [14] where sensitivity and specificity are TP/(TP+FN) and TN/(FP+TN) respectively. In ROC analysis, we usually plot and connect each sample to generate a polyline ordered by their classification score in two dimensional Cartesian coordinate system where x axis denotes 1- specificity and y axis denotes sensitivity. The accuracy of the classifier is calculated as an area under the ROC curve (AUROC). In a perfect model, AUROC is 1.0 and in a random guess model, AUROC is 0.5. Most models generally have AUROC which is higher than 0.5 and lower than 1.0. As AUROC becomes closer to 1.0, the model is regarded as more accurate [7].

C. The Approaches to Resolve Data Distribution Problem

The previously proposed methods to resolve data imbalance can be divided into twofold: data sampling and the assignments of weights (penalties) to misclassified instances [13].

There have been two types of data sampling strategies, under-sampling and over-sampling, which is generally used to resolve data imbalance problems. Under-sampling removes a portion of majority class samples randomly or predefined rules in accordance with the number of minority class samples. Obviously this method incurs information loss of majority class samples. However, it has been shown that, if we adopt adequate rules to select and remove samples, it can successfully resolve data imbalance problems [11, 15, 18]. Over-sampling increases the number of minority class samples using data duplication and data generation [3, 11]. This method is advantageous in that there is no information loss in the majority class, however, overall learning time will increase as the number of data samples increases. In particular, since it creates new samples based on the similarity among minority samples, it could trigger the over-fitting problem and generalization error for novel data samples.

In weights assignments methods, cost adaptive learning strategies are generally used to impose different penalties on misclassified patterns. That is, if a sample in minority class is misclassified, the higher penalty is imposed on the misclassification than the penalty when a sample in majority class is misclassified [6, 20]. Although this method does not have problems like information loss of under-sampling or generalization error of over-sampling, it can cause to generate unstable classifiers due to the excessive sensitivity about the samples.

Recently, the combinations of sampling and boosting algorithms such as SMOTEBoost [3], RUSBoost [21], etc. have been applied to data imbalance problem and shown successful results. Boosting algorithms sequentially generate ensemble of classifiers, assigning higher weights to misclassified observations than to correctly classified observations, and thereby it has an advantage that it can strengthen learning on minority class samples with the high probabilities of misclassification.

III. GM-Boost Algorithm

In this section, we will explain SMOTE, AdaBoost, and GM-Boost algorithms which are used in this research.

A. SMOTE Algorithm

SMOTE algorithm is used to generate new samples for minority class data. SMOTE algorithm combines a certain observation with *k* similar minority class samples to generate a new sample according to the following calculation: $X_{new}=X+rand(0,1)\times(X_n-X)$ where X_{new}, X, and X_n respectively means newly generated sample, the original sample, and the nearest *k* samples to the original sample. SMOTE algorithm consists of three steps as followings; Firstly, the nearest *k* samples to the original sample is chosen, secondly the distances of the original sample and *k* samples is multiplied by a random number between zero and one, and finally, the average of the multiplied distances is added to the original sample in order to generate a new sample. In this way, we repeat SMOTE sampling to increase the samples of minority class until both the numbers of the minority class and majority class become same.

B. AdaBoost

To explain AdaBoost, we assume an ensemble $C=\{C_1,C_2,\dots,C_K\}$ composed of K base classifiers from *n* training samples. Then the error rate for *k*th base classifier (e_k) is calculated as an arithmetic mean, which is as follows.

$$e_k=\sum_{i=1}^{n} w_k(i)L(C_k(x_i),y_i)$$

where, $L(C_k(x_i),y_i)=\begin{cases}1\ C_k(x_i)\neq y_i\\ 0\ C_k(x_i)=y_i\end{cases}$ and

$\sum_i w_k(i)=1$

Note that x_i is a vector of predictor variables for *i*th observation, y_i is a category of *i*th observation, and $C_k(x_i)$ is a classification result of *k*th classifier on the predictor variable vector x_i. For the (*k*+1)th classifier, the weight for *i*th observation is adjusted as follows, which impose higher weights on misclassified observations.

$$w_{k+1}(i)=\frac{w_k(i)exp(-\alpha_k C_k(x_i)y_i)}{Z_k}$$

where $Z_k=\sum_i w_k(i)exp(-\alpha_k C_k(x_i)y_i)$

Note that α_k is conceptually interpreted as an importance or accuracy of the classifier, and calculated as $\alpha_k=\frac{1}{2}ln\big((1-e_k)/e_k\big)$. When the training samples are constructed for (*k*+1)th classifier, since higher weights are assigned to misclassified observations, the boosting algorithm can proceed training focused on misclassified observations. The ensemble learning algorithm stops when $e_k > 0.5$, and the classification result of the ensemble for *i*th observation is a weighted mean of base classifiers' classification expressed as follows:

$$C(x_i)=sign\left(\sum_{k=1}^{K}\alpha_k C_k(x_i)\right)$$

Because of the advantage that AdaBoost algorithm provides learning opportunity to minority class samples, various boosting algorithms based on AdaBoost are frequently applied to data imbalance problem as an alternative solution. As data imbalance is more severe, the error rate for minority class is higher whereas the error rate for majority class is lower. Since higher weights are assigned to minority class samples in the process of constructing training samples for new classifier, the new classifier will strengthen its learning for minority class. In this way, although learning algorithm is concentrated on majority class samples in the beginning stage of ensemble learning, gradually there become more learning opportunities for minority class samples. Because of this characteristic, those boosting algorithms have an advantage that it yields robust learning performance even under data imbalance.

However, the boosting algorithms can exhibit the over-fitting and generalization problems because they try to maximize arithmetic accuracy. The error rate of the classifier e_k and the performance of the classifier, a_k, are measures based on arithmetic mean. As is mentioned before, measures based on arithmetic accuracy might not be valid as a useful objective function because the objective function based on arithmetic measures tends to generate a strongly biased classification function towards majority class or class with high similarity among samples. Especially, when the boosting algorithms are applied after SMOTE algorithm, which generates a new data sample from a group of adjacent data samples weighted with their inter-distances, it will increase the inductive bias due to the increased similarity among the group of data samples and will eventually aggravate the over-fitting effects. To alleviate these problems, we introduce a notion of accuracy based on geometric mean, which can consider predictive performances of both majority class and minority class, to machine learning algorithms.

C. GM-Boost Algorithm

In addition to the aforementioned assumptions for AdaBoost algorithm, we assume that, out of *n* training samples, n^+ samples are in minority class and n^- samples are in majority class. We let e_k^+ be the error rate for minority class of *k*th classifier and e_k^+ be the error rate for minority class of *k*th classifier. Then the geometric mean based error rate e_k, can be defined as follows:

$$e_k=\sqrt{e_k^+\cdot e_k^-},$$

$$\text{where } e_k^+=\frac{\sum_{i=1}^{n^+}w_k(i)L(C_k(x_i),y_i)}{\sum_{i=1}^{n^+}w_k(i)} \text{ and}$$

$$e_k^-=\frac{\sum_{i=1}^{n^-}w_k(i)L(C_k(x_i),y_i)}{\sum_{i=1}^{n^-}w_k(i)}$$

Accordingly, α_k which means classification accuracy of the classifier is calculated as a geometric mean based accuracy of classification accuracies of minority class and majority class.

$$\alpha_k=ln\left(\sqrt{\mu\cdot\alpha_k^+\cdot\alpha_k^-}\right),$$

$$\text{where } \alpha_k^+=\frac{1-e_k^+}{e_k} \text{ and } \alpha_k^-\frac{1-e_k^-}{e_k}$$

Note that μ is a weighting degree that controls the weight value multiplied to each instance. Following AdaBoost, the weight imposed on the samples for (k+1)th classifier is calculated as follows:

$$w_{k+1}(i) = \frac{w_k(i)\exp(-\alpha_k C_k(x_i) y_i)}{Z_k}$$

$$\text{where } Z_k = \sum_i w_k(i)\exp(-\alpha_k C_k(x_i) y_i)$$

And the final classification result for ith observation is calculated as a linear combination of ensemble results and α_k.

$$C(x_i) = sign\left(\sum_{k=1}^{K} \alpha_k C_k(x_i)\right)$$

Having an advantage of providing learning opportunity to minority class samples, various boosting algorithms based on AdaBoost are frequently applied to data imbalance problem as an alternative solution. As data imbalance is more severe, the error rate for minority class is higher whereas the error rate for majority class is lower. Since higher weights are assigned to minority class samples in the process of constructing training samples for new classifier, the new classifier will strengthen its learning for minority class. In this way, although learning algorithm is concentrated on majority class samples in the beginning stage of ensemble learning, gradually there become more learning opportunities for minority class samples. Upon such characteristic, AdaBoost has an advantage of yielding robust learning performance even under data imbalance.

However, the boosting algorithms can exhibit the over-fitting and generalization problems because they try to maximize arithmetic accuracy. The error rate of the classifier ek and the performance of the classifier, a_k, are measures based on arithmetic mean. As mentioned before, measures based on arithmetic accuracy might not be valid as a useful objective function because the objective function based on arithmetic measures tends to generate a strongly biased classification function towards majority class or class with high similarity among samples. Especially, when the boosting algorithms are applied after SMOTE algorithm, which generates a new data sample from a group of adjacent data samples weighted with their inter-distances, it will increase the inductive bias due to the increased similarity among the group of data samples and will eventually aggravate the over-fitting effects. The notion of geometric accuracy, which can consider predictive performances of both majority class and minority class, is introduced to alleviate these problems.

IV. Research Design

We collected the experimental data used for this research from a Korean commercial bank. The bankrupt companies are 500 audited manufacturing companies during year 2002 to year 2005, while the non-bankrupt companies are 2,500 audited manufacturing companies during 2002-2005. For the non-bankrupt companies, we collected 10,000 firm-year financial statements during 2001-2004. In this way, we collected a total of 10,000 financial statements based on firms-year standard, and the average bankrupt rate for the four years is about five percent, which falls in the expected range of bankruptcy rate (three to five percent) estimated by professional credit rating agencies.

As for the financial ratios for bankruptcy prediction, we collected seven thirty financial ratios, which have been usefully applied in the previous corporate bankruptcy prediction researches. The collected ratios are divided into seven financial ratio groups including profitability, debt coverage, leverage, capital structure, liquidity, activity, and size. Consequently, the seven final input variables, each of which has the highest AUROC in each group, are selected.

Variance information factor (VIF) analysis is performed to check for multicollinearity among the seven financial ratios. Table 1 shows the AUROC and VIF of the seven final input variables. We can see that the chosen variables do not exhibit any substantial multicollinearity because all the VIFs are below four.

TABLE 1. THE RESULT OF VARIANCE INFLATION FACTOR ANALYSIS ON THE CHOSEN VARIABLES.

Variables	AUROC	**VIF**
Ordinary income to total assets	51.7	1.36
EBITDA to Interest expenses	51.2	2.11
Total debt to total assets	50.9	1.77
Retained earning to total assets	52.5	2.53
Cash ratio	45.5	1.34
Inventory to sales	30.5	1.59

V. Research Results

Sequential minimal optimization (SMO) is used as a SVM base classifier and the radial basis function (RBF) is used as as a kernel function. There are two parameters in RBF kernels: acceptable error C and kernel parameter δ^2. We made up various configurations of the two parameters: varying C from 1 to 250, and δ^2 from 1 to 200.

We prepared samples through two stages. At the first stage, we chose samples from the total of 10,500 cases, with the ratio of bankrupt companies to normal companies as 1:1(A), 1:3(B), 1:5(C), 1:10(D), and 1:20(E). Then we set 60% of each of them as training samples, and the rest 40% of each of them as test samples. Table 2 shows these configurations of samples. We repeated these steps of the first stage fifty times to generate fifty training sample sets and fifty test sample sets for each of the five configurations (A, B, C, D, and E).

TABLE 2. CONFIGURATIONS OF IMBALANCED DATA SAMPLES

Set		**Training**			**Validation**		
		Bankrupt	***Normal***	***Total***	***Bankrupt***	***Normal***	***Total***
A	1:1	300	300	600	200	200	400
B	1:3	300	900	1,200	200	600	800
C	1:5	300	1,500	1,800	200	1,000	1,200
D	1:10	300	3,000	3,300	200	2,000	2,200
E	1:20	300	6,000	6,300	200	4,000	4.200

At the second stage, we used SMOTE algorithm, where k is set to five, to generate new bankrupt companies, so that we obtained the number of bankrupt companies same with that of normal companies. Table 3 shows these configurations of samples. We repeated the same sampling process fifty times to generate fifty training sample sets and fifty test sample sets for each of four configurations (B, C, D, and E).

TABLE 3. CONFIGURATIONS OF BALANCED DATA SAMPLES.

Set		Training			Validation		
		Bankrupt	*Normal*	*Total*	*Bankrupt*	*Normal*	*Total*
A	1:1	300	300	600	200	200	400
B	1:3	900	900	1,200	200	600	800
C	1:5	1,500	1,500	1,800	200	1,000	1,200
D	1:10	3,000	3,000	3,300	200	2,000	2,200
E	1:20	6,000	6,000	6,300	200	4,000	4.200

A. Experimental Results in Imbalanced Data

Table 4 shows the results of average accuracy of fifty validations. In case of AdaBoost, as the data imbalance is getting severe, arithmetic accuracy over total samples is steadily increased due to the high accuracy over samples of majority class, while the arithmetic accuracy for minority class is dramatically reduced, and thereby geometric accuracy over total samples is gradually decreased. In particular, average accuracy for minority class of sample groups C, D, and E is 7%, 4.5%, and 3.5%, respectively. It indicates that the classification for minority class is meaningless. Those results are caused by arithmetic error and accuracy calculation of AdaBoost.

Comparing to AdaBoost, however, GM-Boost shows stable arithmetic accuracy for minority class and geometric accuracy over total samples. T-test is performed to analyze the difference of geometric accuracy between both boosting algorithms for the five configurations (A, B, C, D, and E). The results of T-test show that the prediction accuracy between two training algorithms for sample group A is significantly different at 5% level and for sample group B, C, D, and E is different at 1% level, respectively. The difference in geometric accuracies becomes higher, as the data imbalance becomes more severe.

B. Experimental Results in Balanced Data

We apply the final sampled sets generated from SMOTE to AdaBoost and GM-Boost experiments. Table 5 shows the results of average accuracy of fifty validations. As noted, the higher is the proportion of new generated samples in minority class, the higher is the similarity among minority class samples. SVM, the base classifier of AdaBoost, will tend to learn focusing on minority samples with high similarity because this strategy is helpful maximizing arithmetic accuracy. Boosting algorithms also try to modify the weight of each instance based on misclassification, but do not try to balance majority class error and minority class error. This problem leads to over-fitting problem and deteriorates the performance of SMOTEBoost in the perspectives of generalization and prediction for novel samples.

In our case, since data set E has the higher proportion of new generated samples and the higher similarity among data samples than any other data sets, it is likely to show the lower prediction performance for novel samples. Hence, while the accuracy of AdaBoost for majority class samples consistently lies on the interval between 0.750 and 0.830, its accuracy for minority class samples becomes lower as the degree of data imbalance is higher. Thus, arithmetic accuracy of AdaBoost stably lies between 0.750 and 0.798, but geometric accuracy continues to deprecate from 0.797 to 0.734. On the contrary, GM-Boost, that employs geometric accuracy, systematically avoids this over-fitting problem because it considers both accuracies of majority class category and minority class category. Consequently, GM-Boost exhibits more robustness and generalization than AdaBoost does for novel test samples. T-test is performed to compare the prediction accuracy between AdaBoost and GM-Boost for the five configurations (A, B, C, D, and E). The results show that significant difference between two algorithms in classification accuracies for all configurations except the configuration A.

TABLE 4. PREDICTION ACCURACY AND THE T-TEST FOR THE FIVE CONFIGURATIONS OF IMBALANCED DATA SAMPLES

Set	SVM				GM-Boost				t-value
	Majority	Minority	Arithmetic	Geometric	Majority	Minority	Arithmetic	Geometric	
A	0.820	0.755	0.788	0.787	0.820	0.780	0.800	0.800	1.851*
B	0.960	0.330	0.803	0.563	0.893	0.630	0.828	0.750	2.435**
C	0.990	0.070	0.837	0.263	0.891	0.610	0.844	0.737	2.704**
D	0.999	0.045	0.912	0.212	0.916	0.505	0.879	0.680	3.291**
E	0.998	0.035	0.952	0.187	0.912	0.420	0.889	0.619	3.557**

** and * represent significance levels at 1% and 5%, respectively.

TABLE 5. PREDICTION ACCURACY AND THE T-TEST FOR THE FIVE CONFIGURATIONS OF BALANCED DATA SAMPLES

Set	AdaBoost				GM-Boost				t-value
	Majority	Minority	Arithmetic	Geometric	Majority	Minority	Arithmetic	Geometric	
A	0.830	0.765	0.798	0.797	0.820	0.780	0.800	0.800	0.152
B	0.750	0.750	0.750	0.750	0.747	0.770	0.753	0.758	1.852*
C	0.775	0.720	0.766	0.747	0.808	0.745	0.798	0.776**	2.438**
D	0.755	0.720	0.751	0.737	0.775	0.765	0.774	0.770*	3.257***
E	0.775	0.695	0.771	0.734	0.776	0.785	0.776	0.780*	3.997***

***, **, and * represent significance levels at 1%, 5%, and 10%, respectively.

VI. Conclusion and Future Research Directions

Data imbalance problem has received a lot of attention in machine learning community because it is one of the causes that degrade the performance of classifiers or predictors. In our research, we proposed GM-Boost algorithm to resolve data imbalance problem. The proposed GM-Boost algorithm is applied to bankruptcy prediction task to verify the performance of GM-Boost algorithm. At the first stage, five sample groups are constructed according to different data balance rates (1:1, 1:3, 1:5, 1:10, and 1:20) and classification experiments using AdaBoost and GM-Boost are performed against those imbalanced data sets. At the second stage, SMOTE algorithm is used to generate new bankrupt company data sets and the newly sampled sets is applied to AdaBoost and GM-Boost experiments for the performance verification of GM-Boost in balanced data. Experimental results show that GM-Boost has the advantages of high prediction power and robust learning capability in imbalanced data distribution as well as balanced data distribution.

We expect the following future researches to be conducted to cope with the limitations of GM-Boost. Firstly, boosting algorithms have drawbacks that degrade classification accuracy when outliers are included in the learning samples or when there is high correlation between the classifiers in the ensemble. Various methods have been proposed to compensate these shortcomings [4,5,20], and we plan to conduct researches to develop algorithms coupled with those methods. Secondly, the ensemble algorithm we propose in this research is a modification of a boosting algorithm to solve data imbalance problem. However, it can be possible to solve the data imbalance problem by combining our results with SVM kernel management [11,26], so we anticipate future researches in this direction.

References

[1] L. Bruzzone, and S. B. Serpico, “Classification of imbalanced remote-sensing data by neural networks,” Pattern Recognition Letters 18(11-13), 1997, pp. 1323–1328

[2] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, “SMOTE: synthetic minority oversampling techniques,” Journal of Artificial Intelligence Research, 16, 2002, pp. 321-357.

[3] N. Chawla, A. Lazarevic, L. Hall, and K. Bowyer, “SMOTEBoost: improving prediction of the minority class in boosting,” 7th European conference on principles and practice of knowledge discovery in databases. Cavtat-Dubrovnik, Croatia, 2003, pp. 107-119.

[4] T. M. Cover, and J. A. Thomas, “Element of information theory”, John Wiley & Sons, 1991.

[5] G. A. Darbellay, “An estimator of the mutual information based on a criterion for independence,” Computational Statistics and Data Analysis, 32, 1999, pp. 1-17.

[6] C. Elkan, “The foundation of cost-sensitive learning,” Proceedings of the 17th International Joint Conference on Artificial Intelligence, Seattle, WA, 2001, pp. 973-978.

[7] T. Fawcett, “An introduction to ROC analysis,” Pattern Recognition Letters, 27, 2006, pp. 861-874.

[8] T. Fawcett, and F. Provost, “Adaptive fraud detection. Data Mining and Knowledge discovery,” 1(3) , 1997, pp. 291-316.

[9] Y. Freund, and R. E. Schapire, “A decision theoretic generalization of online learning and an application to boosting,” Journal of Computer and System Science, 55(1) , 1997, pp. 119-139.

[10] X. Hong, “A kernel-based two-class classifier for imbalanced data sets,” IEEE Transactions on neural networks, 18(1) , 2007, pp. 28-40.

[11] N. Japkowicz, and S. Stephen, “The class imbalance problem: a systematic study,” Intelligent Data Analysis, 6(5) , 2002, pp. 429-250.

[12] P. Kang, and S. Cho, “EUS SVMs: Ensemble of Under-Sampled SVMs for Data Imbalance Problems,” ICONIP 2006, Part I, LNCS 4232, pp 837-846.

[13] S. Kotsiantis, D. Tzelepis, E. Kounmanakos, and V. Tampakas, “Selective costing voting for bankruptcy prediction,” International Journal of Knowledge-based and Intelligent Engineering Systems, 11, 2006, pp. 115-127, 2007.

[14] M. Kubat, R. Holte, and S. Matwin, “Learning when Negative example abound,” Proceedings of the 9th European Conference on Machine Learning, ECML’97, 1997.

[15] M. Kubat, and S. Matwin, “Addressing the curse of imbalanced training sets: one-sided selection,” Proceedings of the Fourteenth International Conference on Machine Learning, 1997, pp. 179-186.

[16] M. Kubat, R. Holte, and S. Matwin, “Machine Learning for the detection of oil spills in satellite radar images,” Machine Learning 30(2), 1998, pp. 195-215

[17] Y. S. Kwon, I. G. Han, and K. C. Lee, “Ordinal Pairwise Partitioning (OPP) approach to neural networks training in bond rating,” Intelligent Systems in Accounting, Finance and Management, 6, 1997, .pp. 23-40.

[18] J. Laurikkala, “Instance-based data reduction for improved identification of difficult small classes,” Intelligent Data Analysis, 6(4), 2002, pp. 311-322.

[19] T. T. Maia, A. P. Braga, and A. F. Carvalho, “Hybrid classification algorithms based on boosting and support vector machines,” Kybernetes, 37(9), 2008, pp. 1469-1491.

[20] F. Provost, and T. Fawcett, “Robust classification for imprecise environments,” Machine Learning, 42, 2001, pp. 203-231.

[21] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, “RUSBoost: Improving classification performance when training data is skewed,” 19th International Conference on Pattern Recognition, 2008, pp. 1-4.

[22] H. J. Shin, and S. Z. Cho, “Response Modeling with Support Vector Machine,” Expert Systems with Applications 30(4), 1997, pp. 746–760

[23] B. X. Wang, and N. Japkowicz, “Boosting support vector machines for imbalanced data sets,” Knowledge and Information Systems, Knowledge and Information Systems, 25, 2009, pp. 1-20.

[24] G. Wu, and E. Chang, “Adaptive feature-space conformal transformation for imbalanced data learning,” Proceedings of the 20th International Conference on Machine Learning, 2003