An Adaptive Oversampling Technique for Imbalanced Datasets

Shaukat Ali Shahee and Usha Ananthakumar(✉)

Indian Institute of Technology Bombay, Mumbai 400076, India
shaukatali.shahee@iitb.ac.in, usha@som.iitb.ac.in

Abstract. Class imbalance is one of the challenging problems in classification domain of data mining. This is particularly so because of the inability of the classifiers in classifying minority examples correctly when data is imbalanced. Further, the performance of the classifiers gets deteriorated due to the presence of imbalance within class in addition to between class imbalance. Though class imbalance has been well addressed in literature, not enough attention has been given to within class imbalance. In this paper, we propose a method that can adaptively handle both between-class and within-class imbalance simultaneously and also that can take into account the spread of the data in the feature space. We validate our approach using 12 publicly available datasets and compare the classification performance with other existing oversampling techniques. The experimental results demonstrate that the proposed method is statistically superior to other methods in terms of various accuracy measures.

Keywords: Classification · Imbalanced dataset · Oversampling
Model based clustering · Lowner John ellipsoid

1 Introduction

In data mining literature, class imbalance problem is considered to be quite challenging. The problem arises when the class of interest contains a relatively lower number of examples compared to other class examples. In this study, the minority class, the class of interest is considered positive and the majority class is considered negative. Recently, several authors have addressed this problem in various real life domains including customer churn prediction [6], financial distress prediction [10], employee churn prediction [39], gene regulatory network reconstruction [7] and information retrieval and filtering [35]. Previous studies have shown that applying classifiers directly to imbalance dataset results in poor performance [34,41,43]. One of the possible reasons for the poor performance is skewed class distribution because of which the classification error gets dominated by the majority class. Another kind of imbalance is referred to as *within-class* imbalance which pertains to the state where a class composes of different number of sub-clusters (sub-concepts) and these sub-clusters in turn, containing different number of examples.

P. Perner (Ed.): ICDM 2018, LNAI 10933, pp. 1–16, 2018.
https://doi.org/10.1007/978-3-319-95786-9_1

In addition to class imbalance, *small disjuncts*, *lack of density*, *overlapping between classes* and *noisy examples* also deteriorate the performance of the classifiers [2,28–30,36]. The *between-class* imbalance along with *within-class* imbalance is an instance of *problem of small disjuncts* [26]. Literature presents different ways of handling class imbalance such as data preprocessing, algorithmic based, cost-based methods and ensemble of classifier sampling methods [12,17]. Though no method is superior in handling all imbalanced problems, sampling based methods have shown great capability as they attempt to improve data distribution rather than the classifier [3,8,23,42]. Sampling method is a preprocessing technique that modifies the imbalanced data to a balanced data using some mechanism. This is generally carried out by either increasing the minority class examples called as oversampling or by decreasing the majority examples, referred to as undersampling [4,13]. It is not advisable to undersample the majority class examples if minority class has complete rarity [40]. The current literature available on simultaneous *between-class* imbalance and *within-class* imbalance is limited.

In this paper, an adaptive method for handling between class imbalance and within class imbalance simultaneously based on an oversampling technique is proposed. It also factors in the scatter of data for improving the accuracy of both the classes on the test set. Removing between class imbalance and within class imbalance simultaneously helps the classifier to give equal importance to all the sub-clusters, and adaptively increasing the size of sub-clusters handles the randomness in the dataset. Generally, classifier minimizes the total error, and removal of between class imbalance and within class imbalance helps the classifier in giving equal weight to all the sub-clusters irrespective of the classes thus resulting in increased accuracy of both the classes. Neural network is one such classifier and is being used in this study. The proposed method is validated on publicly available data sets and compared with well known existing oversampling techniques. Section 2 discusses the proposed method and analysis on publicly available data sets is presented in Sect. 3. Finally, Sect. 4 concludes the paper with future work.

2 An Adaptive Oversampling Technique

The approach in this proposed method is to oversample the examples in such a way that it helps the classifier in increasing the classification accuracy on the test set.

The proposed method is based on two challenging aspects faced by the classifiers in case of imbalanced data sets. First one is the case of the loss function, where the majority class dominates the minority class and thus eventually, minimization of the loss function is largely due to minimization of the majority class. Because of this, the decision boundary between the classes does not get shifted towards the minority class. Removing the between class and within class imbalance helps in removing the dominance of the majority class.

Another challenge faced by the classifiers is the accuracy of the classifiers on the test set. Due to the randomness of data, if the test example lies in the

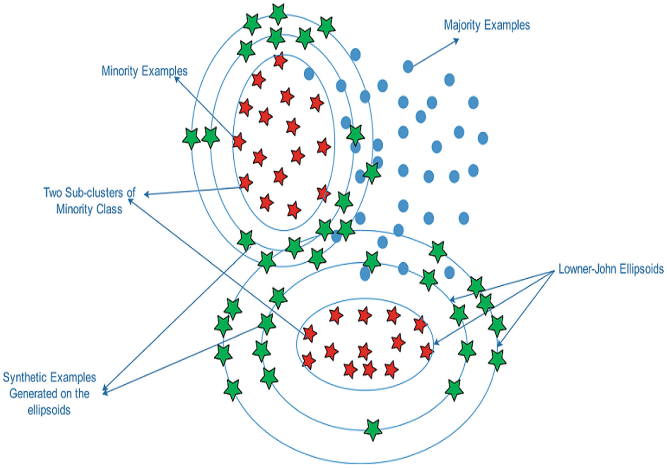
Majority Examples
Minority Examples
Two Sub-clusters of Minority Class
Lowner-John Ellipsoids
Synthetic Examples Generated on the ellipsoids

Fig. 1. Synthetic minority class examples generation on the peripheral of Lowner John ellipsoids

outskirts of the sub-clusters, there is a need to adjust the decision boundary to minimize misclassification. This is achieved by expanding the size of the sub-cluster in order to cope with such test examples. Now the question is, what is the surface of the sub-clusters and how far the sub-clusters should be expanded. To answer this, we use minimum volume ellipsoid that contains the dataset known as *Lowner John ellipsoid* [33]. We adaptively increase the size of the ellipsoid and synthetic examples are generated on the surface of the ellipsoid. One such instance is shown in Fig. 1 where minority class examples are denoted by stars and majority class examples by circle.

In the proposed method, the first step is data cleaning where the noisy examples are removed from the dataset as this helps in reducing the oversampling of noisy examples. After data cleaning, the concept is detected by using model based clustering and the boundary of each of the clusters is determined by *Lowner John ellipsoid*. Subsequently, the number of examples to be oversampled is determined based on the complexity of sub-clusters and synthetic data are generated on the peripheral of the ellipsoid. Following section elaborates the proposed method in detail.

2.1 Data Cleaning

In data cleaning process, the proposed method removes the noisy examples in the dataset. An example is considered as noisy if it is surrounded by all the examples of other class as defined in [3]. The number of examples is taken to be 5 in this study as also being considered in other studies including [3,32].

2.2 Locating Sub-clusters

Model based clustering [16] is used with respect to minority class to identify the sub-clusters (or sub-concepts) present in the dataset. We have used MCLUST [15] for implementing the model based clustering. MCLUST is a R package that implements the combination of hierarchical agglomerative clustering, Expectation Maximization (EM) and Bayesian Information criterion (BIC) for comprehensive cluster analysis.

2.3 Structure of Sub-clusters

The structure of sub-clusters can be obtained using eigenvalues and eigenvector. Eigenvectors gives the shape of sub-cluster and size is given by eigenvalues. Let $X = \{x_1, x_2, \ldots, x_m\}$ be a dataset having m examples and n features. Let the mean vector of X be μ and the covariance matrix computed by $\Sigma = E[(X - \mu)(X - \mu)^T]$. The eigenvalues ($\lambda$) and eigenvectors v of the covariance matrix Σ are found such that $\Sigma v = \lambda v$.

2.4 Identifying the Boundary of Sub-clusters

For each of the sub-clusters, *Lowner-John ellipsoid* is obtained as given by [33]. This is a minimum volume ellipsoid that contains the convex hull of $C = \{x_1, x_2, \ldots, x_m\} \subseteq R^n$. The general equation of ellipsoid is

$$\varepsilon = \{v| ||Av + b||_2 \leq 1\} \tag{1}$$

We assume that $A \in S^n_{++}$ is a positive definite matrix where the volume of ε is proportional to $detA^{-1}$. The problem of computing the minimum volume ellipsoid containing C can be expressed as

$$\begin{array}{ll} \text{minimize} & logdetA^{-1} \\ \text{subject to} & ||Ax_i + b||_2 \leq 1, \; i = 1, \ldots, m. \end{array} \tag{2}$$

We use *CVX* [21], a Matlab-based modeling system for solving this optimization problem.

2.5 Synthetic Data Generation

The synthetic data generation is based on the following three steps

1. In the first step, the proposed method determines the number of examples to be oversampled per cluster. The number of minority class examples to be oversampled is computed using Eq. (3).

$$N = TC0 - TC1 \tag{3}$$

where N is the number of minority class examples to be oversampled, $TC0$ is the total number of examples of majority class *class 0* and $TC1$ is the total number of examples of *class 1.*

It then computes the complexity of sub-clusters based on the number of danger zone examples. An example is called a danger zone example or a borderline example if an example under consideration is surrounded by more than 50% examples of other class as also being considered in other studies including [23]. That is, if k is the number of nearest neighbors under consideration, an example being a danger zone example implies $k/2 \leq z < k$ where z is the number of other class examples among the k nearest neighbor examples. For example, Fig. 2 shows two sub-clusters of minority class having 4 and 2 danger zone examples. In this study, we consider $k = 5$ as in [3]. Let $c_1, c_2, c_3, \ldots, c_q$ be the number of danger zone examples present in the sub-clusters $1, 2, \ldots, q$ respectively. The number of examples to be oversampled in the sub-cluster i is given by

$$n_i = \frac{c_i * N}{\sum_{i=1}^{q} c_i} \tag{4}$$

2. Having determined the number of examples to be oversampled, the next task is to weigh the danger zone examples in accordance with the direction of the ellipsoid and its distance from the centroid. These weights are computed with respect to the eigenvectors of the variance-covariance matrix of the dataset. For example, consider Fig. 3 where A and B denote the danger zone examples. Here we compute the inner product between danger zone examples A and B with the eigenvectors *Evec1* and *EVec2* that form acute angles with the danger zone examples. The weight of A, $W(A)$ is computed as

$$W(A) = \langle A, EVec1 \rangle + \langle A, Evec2 \rangle \tag{5}$$

Similarly the weight of B, $W(B)$ is computed as

$$W(B) = \langle B, EVec1 \rangle + \langle B, Evec2 \rangle \tag{6}$$

Thus, when data is n dimensional, the total weight of the b_k^{th} danger zone example w_k is

$$w_k = \sum_{i=1}^{n} \langle b_k, e_i \rangle \tag{7}$$

where e_i is the eigenvector.

3. In each of the sub-clusters, synthetic examples are generated on the *Lowner John ellipsoid* by linear extrapolation of the selected danger zone example where the selection of danger zone example is carried out with respect to the weights obtained in step 2. Here

$$P(b_k) = \frac{w_k}{\sum_{i=1}^{c_i} w_i} \tag{8}$$

where $P(b_k)$ is the probability of selecting danger zone example b_k and w_k is the weight of k^{th} danger zone example present in the sub-cluster c_i. The selected danger zone example is extrapolated and a synthetic example is generated on the *Lowner John ellipsoid* at the point of intersection of the

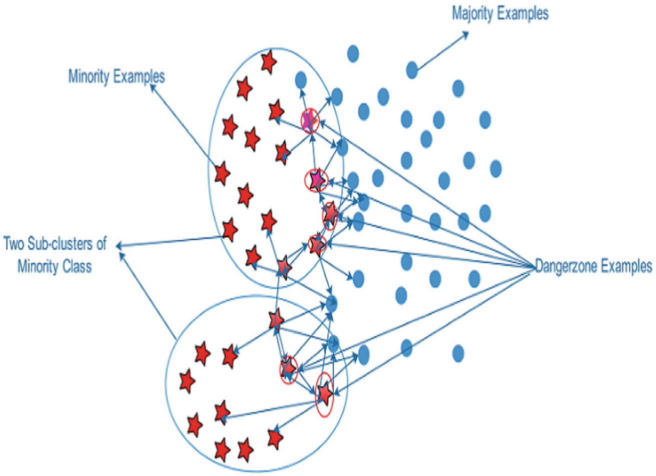

Fig. 2. Illustration of danger zone examples of minority class sub-clusters

extrapolated vector with *Lowner John ellipsoid*. Let the centroid of the ellipsoid be $center = -A^{-1} * b$ and if b_k is the danger zone example selected based on the probability distribution given by Eq. (8), the vector $v = b_k - center$ is extrapolated by 'r' units to intersect with the ellipsoid and the synthetic example s_t thus generated is given by

$$s_t = center + \frac{(r + C) * v}{\|v\|} \tag{9}$$

where C controls the expansion of the ellipsoid.

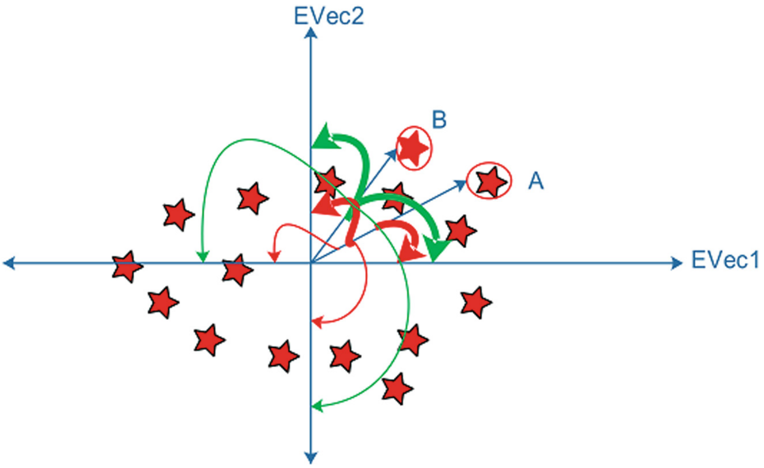

Fig. 3. Illustration of danger zone examples A & B of minority class forming acute angle with eigenvector in bold line

The whole procedure of the algorithm is explained in Algorithm 1.

Algorithm 1. An Adaptive Oversampling Technique for Imbalanced Data sets

```
Input: Training dataset: $S = \{X_i, y_i\}, i = 1, ..., m$; $X_i \in R^n$ and $y_i \in \{0, 1\}$ Positive
    class: $S^+ = \{X_i^+, y_i^+\}, i = 1, .., m^+$; Negative class: $S^- = \{X_i^-, y_i^-\}, i = 1, ..., m^-$;
    $S = S^+ \cup S^-$; $m = m^+ + m^-$ and No. of examples to be oversampled: $N = m^- - m^+$
Output: Oversampled Dataset
 1: Clean the training set
 2: Apply Model-Based clustering on $S^+$, return $\{smin_1, .....smin_q\}$ sub-clusters.
 3: for each minority sub-cluster $smin_i$, $1 \leq i \leq q$ do
 4:    $B_i \leftarrow$ DangerzoneExample($smin_i$) //Return list of danger zone examples
 5: end for
 6: for each minority sub-cluster $smin_i$, $1 \leq i \leq q$ do
 7:    $n_i = \frac{length(B_i)*N}{\sum_{n=1}^{q} length(B_i)}$ for $i = 1, ..., q$ // No of examples to oversample in sub-
       cluster i
 8: end for
 9: for $i = 1$ to $q$ do
10:    $\mu_i \leftarrow$ MEAN($smin_i$)
11:    $\Sigma_i \leftarrow$ COV($smin_i$)
12:    Compute the Lowner John ellipsoids of $smin_i$ as given in Subsect. 2.4 gives $A$
       and $b$
13:    The eigenvectors $v_1, ..., v_n$ and eigenvalues $\lambda_1, ...\lambda_n$ of the covariance matrix $\Sigma_i$
       of dataset in sub-clusters $smin_i$ is computed by $\Sigma v_i = \lambda_i v$
14:    for $j = 1$ to $length(B_i)$ do
15:       $b_j \leftarrow B_i[j]$
16:       $w_j = 0$
17:       for $t = 1$ to $n$ do
18:          $weight = \langle b_j, v_t \rangle$
19:          if $weight \geq 0$ then
20:             $w_j = w_j + weight$
21:          end if
22:       end for
23:    end for
24:    $p(b_j) = \frac{w_j}{\sum_{n=1}^{length(B_i)} w_n}$ // Compute the prob. dist of danger zone examples
25:    $NewSyntheticExample = \Phi$
26:    for $t = 1$ to $n_i$ do
27:       Select the danger zone example based $b_i$ based on step 24
28:       Synthetic example $s_t$ has been generated as given in equation (9)
29:       $NewSyntheticExample = NewSyntheticExample \cup \{s_t\}$
30:    end for
31:    $oversample_i = smin_i \cup NewSyntheticExample$
32: end for
33: $Oversampled\ Dataset = \bigcup_{i=1}^{q} oversample_i$
```

3 Experiments

3.1 Data Sets

We evaluate the proposed method on 12 publicly available datasets which have skewed class distribution available on the KEEL dataset [1] repository. As *yeast* and *pageblocks* data sets have multiple classes, we have suitably transformed the data sets to two classes to meet our needs of binary class problem. In case of *yeast* dataset, it has 1484 examples and 10 classes {MIT, NUC, CYT, ME1, ME2, ME3, EXC, VAC, POX, ERL}. We choose ME3 as the minority class and the remaining are combined to form the majority class. In case of *pageblocks* dataset, it has 548 examples and 5 classes {1, 2, 3, 4, 5}. We choose *1* as majority class and the rest as the minority class. Minority class is chosen in both the data sets in such a way that it contains reasonable number of examples to identify the presence of sub-concepts and also to maintain the imbalance with respect to the majority class. The rest of the data sets were taken as they are. Table 1 represents the characteristics of various data sets used in the analysis.

Table 1. The data sets

Data sets	Total exp	Minority exp	Majority exp	No. attribute
glass1	214	76	138	9
pima	768	268	500	8
glass0	214	70	144	9
yeast1	1484	429	1055	8
vehicle2	846	218	628	18
ecoli1	336	77	259	7
yeast	1484	163	1321	8
glass6	214	29	185	9
yeast3	1484	163	1321	8
yeast-0-5-6-7-9_vs_4	528	51	477	8
yeast-0-2-5-7-9_vs_3-6-8	1004	99	905	8
pageblocks	548	56	492	10

3.2 Assessment Metrics

Traditionally, performance of classifiers is evaluated based on the *accuracy* and *error rate* as defined in (10). However, in case of the imbalanced dataset, the accuracy measure is not appropriate as it does not differentiate misclassification between the classes. Many studies address this shortcoming of accuracy measure with regard to imbalanced dataset [9,14,20,31,37]. To deal with class imbalance, various metric measures have been proposed in the literature that is based on the confusion matrix shown in Table 2.

Table 2. Confusion matrix

True class			
Classifier output		p	n
	P	TP	FP
	N	FN	TN

$$\begin{aligned} Accuracy &= \frac{TP + TN}{TP + FN + FP + TN} \\ Error\ rate &= 1 - Accuracy \end{aligned} \tag{10}$$

These *confusion matrix* based measures described by [25] for imbalanced learning problem are *precision*, *recall*, *F-measure* and *G-mean*. These measures are defined as

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

$$F\text{-}Measure = \frac{(1 + \beta^2) Recall * Precision}{\beta^2 * Recall + Precision} \tag{13}$$

Here β is a non-negative parameter that controls the influence of precision and recall. In this study, we set $\beta = 1$ implying that precision and recall are equally important.

$$G\text{-}Mean = \sqrt{\frac{TP}{TP + FN} \frac{TN}{TN + FP}} \tag{14}$$

Another popular technique for evaluation of classifiers under imbalance domain is the Receiving Operating Characteristic (ROC) curve [37]. ROC curve is a graphical representation of the performance of the classifier by plotting *TP rates* versus *FP rates* over possible threshold values. The TP rates and FP rates are defined as

$$\text{TP rate} = \frac{TP}{TP + FN} \tag{15}$$

$$\text{FP rate} = \frac{FP}{FP + TN} \tag{16}$$

The quantitative representation of a ROC curve is the area under this curve and is called AUC [5,27]. For the purpose of evaluation, we use AUC measure as it is independent of the distribution of positive class and negative class examples and hence this metric is not overwhelmed by the majority class examples. Apart from this, we have also considered $F\text{-}Measure$ for both minority and majority class and $G\text{-}Mean$ for comparative purposes.

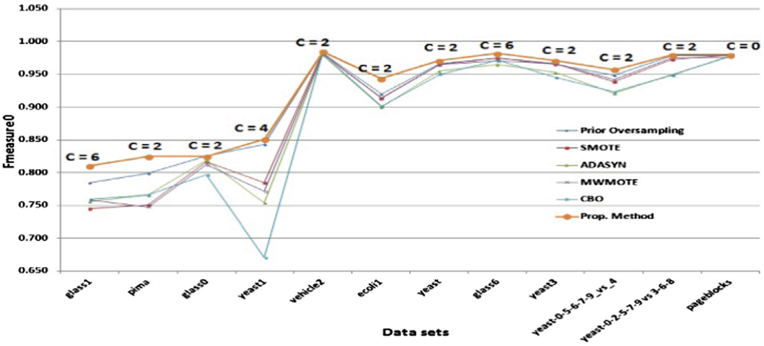

Fig. 4. Results of *F-measure* of majority class for various methods with the best one being highlighted.

3.3 Experimental Settings

In this work, we have used the feed-forward neural network with backpropagation. The structure of the network is such that it has input layers with the number of neurons being equal to the number of features. The number of neurons in the output layer is one as it is a binary classification problem. The number of neurons in the hidden layer is the average of the number of features and number of classes [22]. The activation function used at each neuron is the sigmoid function with learning rate 0.3.

We compare our proposed method with well known existing oversampling methods such as *SMOTE* [8], *ADASYN* [24], *MWMOTE* [3] and *CBO* [30]. We use default parameter settings for these oversampling techniques. In case of *SMOTE* [8], the number of nearest neighbor k is set to 5. In case of *ADASYN* [24], the number of nearest neighbor k is 5 and desired level of balance is 1. In case of *MWMOTE* [3], the number of neighbors used for predicting noisy minority class examples is $k1 = 5$, the number of nearest neighbors used to find majority class examples is $k2 = 3$, the percentage of original minority class examples used in generating synthetic examples is $k3 = |Smin|/2$, the number of clusters in the method is $Cp = 3$ and smoothing and rescaling values of different scaling factors are $Cf(th) = 5$ and $CMAX = 2$ respectively.

3.4 Results

The results of 12 data sets for metric measures *F-measure* of majority and minority class, G-mean and AUC are shown in Figs. 4, 5, 6 and 7. It is enough to show *F-measure* rather than explicitly showing *Precision* and *Recall* because *F-measure* integrates *Precision* and *Recall*. We used 5-fold stratified cross-validation technique that runs 5 independent times and average of this is presented in Figs. 4, 5, 6 and 7. In 5-fold stratified cross-validation technique, a dataset is divided into 5 folds having an equal proportion of the classes. Among the 5 folds, one fold is considered as the test set and the remaining 4 folds are

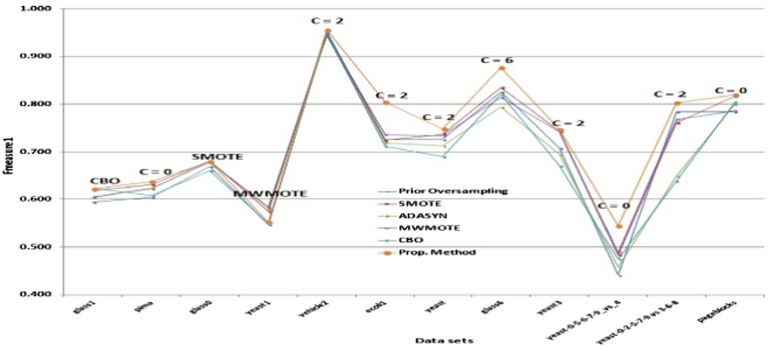

Fig. 5. Results of *F-measure* of minority class for various methods with the best one being highlighted.

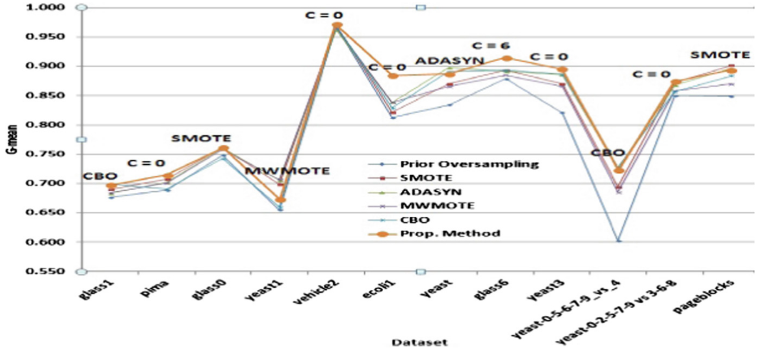

Fig. 6. Results of *G-mean* for various methods with the best one being highlighted.

combined and considered as the training set. Oversampling is carried out only on the training set and not on the test set in order to obtain unbiased estimates of the model for future prediction.

Figure 4 shows the results of *F-measure* of majority class. It is clear from the figure that the proposed method outperforms the other oversampling methods for different values of C. In this study, we consider $C \in \{0, 2, 4, 6\}$ where C controls the expansion of the ellipsoid. $C = 0$ gives the minimum volume *Lowner-John ellipsoid* and $C = 2$ means the size of ellipsoid increases by 2 units. The results of *Fmeasure1* is shown in Fig. 5. From the figure it is clear that the proposed method outperforms the other methods except in case of data sets *glass1, glass0* and *yeast1* where *CBO, SMOTE* and *MWMOTE* perform slightly better. Similarly, the results in case of *G-mean* and *AUC* are shown in Figs. 6 and 7 respectively. The method yielding the best result is highlighted in all the figures.

To compare the proposed method with other oversampling methods, we carried out non-parametric tests as suggested in the literature [11,18,19]. Wilcoxon

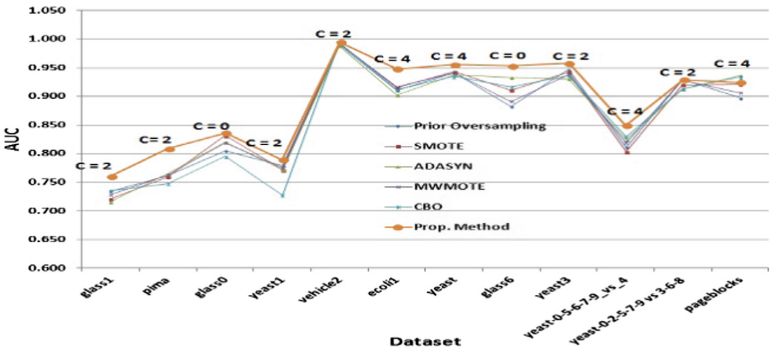

Fig. 7. Results of AUC for various methods with the best one being highlighted.

Table 3. Summary of Wilcoxon signed rank test between our proposed method and other methods

Methods	Proposed method	Metric measure	Hypothesis
Prior oversampling	p_value = 0.003204	F-measure of majority	H_0 rejected
	p_value = 0.002516	F-measure of minority	H_0 rejected
	p_value = 0.0004883	G-mean	H_0 rejected
	p_value = 0.003857	AUC	H_0 rejected
SMOTE	p_value = 0.002516	F-measure of majority	H_0 rejected
	p_value = 0.02061	F-measure of minority	H_0 rejected
	p_value = 0.07733	G-mean	Fail to reject H_0
	p_value = 0.0004883	AUC	H_0 rejected
ADASYN	p_value = 0.0004883	F-measure of majority	H_0 rejected
	p_value = 0.009766	F-measure of minority	H_0 rejected
	p_value = 0.2298	G-mean	Fail to reject H_0
	p_value = 0.004164	AUC	H_0 rejected
MWMOTE	p_value = 0.002478	F-measure of majority	H_0 rejected
	p_value = 0.01344	F-measure of minority	H_0 rejected
	p_value = 0.02531	G-mean	H_0 rejected
	p_value = 0.003857	AUC	H_0 rejected
CBO	p_value = 0.0004883	F-measure of majority	H_0 rejected
	p_value = 0.0009766	F-measure of minority	H_0 rejected
	p_value = 0.01669	G-mean	H_0 rejected
	p_value = 0.001465	AUC	H_0 rejected

signed-rank non-parametric test [38] is carried out on *F-measure of majority class, F-measure of minority class, G-Mean and AUC.* The null and alternative hypothesis are as follows:

H_0: The median difference is zero
H_1: The median difference is positive.

This test computes the difference in the respective measure between the proposed method and the method compared with it and ranks the absolute differences. Let $W+$ be the sum of the ranks with positive differences and $W-$ be the sum of the ranks with negative differences. The test statistic is defined as $W = min(W+, W-)$. Since there are 12 data sets, the *W value* should be less than 17 (critical value) at a significance level of 0.05 to reject H_0 [38]. Table 3 shows the *p-values* of test statistics of Wilcoxon signed-rank test.

The statistical tests indicate that the proposed method statistically outperforms the other methods in terms of *AUC* and *F-measure* of both minority and majority class, although in case of *G-mean* measure, the proposed method does not seem to outperform *SMOTE* and *ADASYN*. Since we use *AUC* for comparison purpose, it can be inferred that our proposed method is superior to other oversampling methods.

4 Conclusion

In this paper, we propose an oversampling method that adaptively handles between class imbalance and within class imbalance simultaneously. The method identifies the concepts present in the data set using model based clustering and then eliminates the between class and within class imbalance simultaneously by oversampling the sub-clusters where the number of examples to be oversampled is determined based on the complexity of the sub-clusters. The method focuses on improving the test accuracy by adaptively expanding the size of sub-clusters in order to cope with unseen test data. 12 publicly available data sets were analyzed and the results show that the proposed method outperforms the other methods in terms of different performance measures such as *F-measure* of both the majority and minority class and AUC.

The work could be extended by testing the performance of the proposed method on highly imbalanced data sets. Further, in our current study, we have expanded the size of clusters uniformly. This could be extended by incorporating the complexity of the surrounding sub-clusters in order to adaptively expand the size of various sub-clusters. This may reduce the possibility of overlapping with other class sub-clusters resulting in increase of classification accuracy.

References

1. Alcalá, J., Fernández, A., Luengo, J., Derrac, J., García, S., Sánchez, L., Herrera, F.: Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. J. Multiple-Valued Log. Soft Comput. **17**(2–3), 255–287 (2010)
2. Alshomrani, S., Bawakid, A., Shim, S.O., Fernández, A., Herrera, F.: A proposal for evolutionary fuzzy systems using feature weighting: dealing with overlapping in imbalanced datasets. Knowl.-Based Syst. **73**, 1–17 (2015)
3. Barua, S., Islam, M.M., Yao, X., Murase, K.: Mwmote-majority weighted minority oversampling technique for imbalanced data set learning. IEEE Trans. Knowl. Data Eng. **26**(2), 405–425 (2014)
4. Batista, G.E., Prati, R.C., Monard, M.C.: A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD Explor. Newsl. **6**(1), 20–29 (2004)
5. Bradley, A.P.: The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recogn. **30**(7), 1145–1159 (1997)
6. Burez, J., Van den Poel, D.: Handling class imbalance in customer churn prediction. Expert Syst. Appl. **36**(3), 4626–4636 (2009)
7. Ceci, M., Pio, G., Kuzmanovski, V., Džeroski, S.: Semi-supervised multi-view learning for gene network reconstruction. PLoS ONE **10**(12), e0144031 (2015)
8. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
9. Chawla, N.V., Lazarevic, A., Hall, L.O., Bowyer, K.W.: SMOTEBoost: improving prediction of the minority class in boosting. In: Lavrač, N., Gamberger, D., Todorovski, L., Blockeel, H. (eds.) PKDD 2003. LNCS (LNAI), vol. 2838, pp. 107–119. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-39804-2_12
10. Cleofas-Sánchez, L., García, V., Marqués, A., Sánchez, J.S.: Financial distress prediction using the hybrid associative memory with translation. Appl. Soft Comput. **44**, 144–152 (2016)
11. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. **7**(Jan), 1–30 (2006)
12. Díez-Pastor, J.F., Rodríguez, J.J., García-Osorio, C.I., Kuncheva, L.I.: Diversity techniques improve the performance of the best imbalance learning ensembles. Inf. Sci. **325**, 98–117 (2015)
13. Estabrooks, A., Jo, T., Japkowicz, N.: A multiple resampling method for learning from imbalanced data sets. Comput. Intell. **20**(1), 18–36 (2004)
14. Fawcett, T.: ROC graphs: notes and practical considerations for researchers. Mach. Learn. **31**(1), 1–38 (2004)
15. Fraley, C., Raftery, A.E.: MCLUST: software for model-based cluster analysis. J. Classif. **16**(2), 297–306 (1999)
16. Fraley, C., Raftery, A.E.: Model-based clustering, discriminant analysis, and density estimation. J. Am. Stat. Assoc. **97**(458), 611–631 (2002)
17. Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., Herrera, F.: A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **42**(4), 463–484 (2012)
18. García, S., Fernández, A., Luengo, J., Herrera, F.: Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: experimental analysis of power. Inf. Sci. **180**(10), 2044–2064 (2010)

19. Garcia, S., Herrera, F.: An extension on “statistical comparisons of classifiers over multiple data sets” for all pairwise comparisons. J. Mach. Learn. Res. **9**(Dec), 2677–2694 (2008)
20. García, V., Mollineda, R.A., Sánchez, J.S.: A bias correction function for classification performance assessment in two-class imbalanced problems. Knowl.-Based Syst. **59**, 66–74 (2014)
21. Grant, M., Boyd, S., Ye, Y.: CVX: Matlab software for disciplined convex programming (2008)
22. Guo, H., Viktor, H.L.: Boosting with data generation: improving the classification of hard to learn examples. In: Orchard, B., Yang, C., Ali, M. (eds.) IEA/AIE 2004. LNCS (LNAI), vol. 3029, pp. 1082–1091. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24677-0_111
23. Han, H., Wang, W.-Y., Mao, B.-H.: Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In: Huang, D.-S., Zhang, X.-P., Huang, G.-B. (eds.) ICIC 2005. LNCS, vol. 3644, pp. 878–887. Springer, Heidelberg (2005). https://doi.org/10.1007/11538059_91
24. He, H., Bai, Y., Garcia, E.A., Li, S.: ADASYN: adaptive synthetic sampling approach for imbalanced learning. In: IEEE International Joint Conference on Neural Networks, IJCNN 2008, (IEEE World Congress on Computational Intelligence), pp. 1322–1328. IEEE (2008)
25. He, H., Garcia, E.A.: Learning from imbalanced data. IEEE Trans. Knowl. Data Eng. **21**(9), 1263–1284 (2009)
26. Holte, R.C., Acker, L., Porter, B.W., et al.: Concept learning and the problem of small disjuncts. In: IJCAI, vol. 89, pp. 813–818. Citeseer (1989)
27. Huang, J., Ling, C.X.: Using auc and accuracy in evaluating learning algorithms. IEEE Trans. Knowl. Data Eng. **17**(3), 299–310 (2005)
28. Japkowicz, N.: Class imbalances: are we focusing on the right issue. In: Workshop on Learning from Imbalanced Data Sets II, vol. 1723, p. 63 (2003)
29. Japkowicz, N., Stephen, S.: The class imbalance problem: a systematic study. Intell. Data Anal. **6**(5), 429–449 (2002)
30. Jo, T., Japkowicz, N.: Class imbalances versus small disjuncts. ACM SIGKDD Explor. Newsl. **6**(1), 40–49 (2004)
31. Kubat, M., Matwin, S., et al.: Addressing the curse of imbalanced training sets: one-sided selection. In: ICML, vol. 97, Nashville, USA, pp. 179–186 (1997)
32. Lango, M., Stefanowski, J.: Multi-class and feature selection extensions of roughly balanced bagging for imbalanced data. J. Intell. Inf. Syst. **50**(1), 97–127 (2018)
33. Lutwak, E., Yang, D., Zhang, G.: L_p John ellipsoids. Proc. Lond. Math. Soc. **90**(2), 497–520 (2005)
34. Maldonado, S., López, J.: Imbalanced data classification using second-order cone programming support vector machines. Pattern Recogn. **47**(5), 2070–2079 (2014)
35. Piras, L., Giacinto, G.: Synthetic pattern generation for imbalanced learning in image retrieval. Pattern Recogn. Lett. **33**(16), 2198–2205 (2012)
36. Prati, R.C., Batista, G.E.A.P.A., Monard, M.C.: Class imbalances *versus* class overlapping: an analysis of a learning system behavior. In: Monroy, R., Arroyo-Figueroa, G., Sucar, L.E., Sossa, H. (eds.) MICAI 2004. LNCS (LNAI), vol. 2972, pp. 312–321. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24694-7_32
37. Provost, F.J., Fawcett, T., et al.: Analysis and visualization of classifier performance: comparison under imprecise class and cost distributions. In: KDD, vol. 97, pp. 43–48 (1997)

38. Richardson, A.: Nonparametric statistics for non-statisticians: a step-by-step approach by Gregory W. Corder, Dale I. Foreman. Int. Stat. Rev. **78**(3), 451–452 (2010)
39. Saradhi, V.V., Palshikar, G.K.: Employee churn prediction. Expert Syst. Appl. **38**(3), 1999–2006 (2011)
40. Weiss, G.M.: Mining with rarity: a unifying framework. ACM SIGKDD Explor. Newsl. **6**(1), 7–19 (2004)
41. Yang, C.Y., Yang, J.S., Wang, J.J.: Margin calibration in SVM class-imbalanced learning. Neurocomputing **73**(1), 397–411 (2009)
42. Yen, S.J., Lee, Y.S.: Cluster-based under-sampling approaches for imbalanced data distributions. Expert Syst. Appl. **36**(3), 5718–5727 (2009)
43. Yu, D.J., Hu, J., Tang, Z.M., Shen, H.B., Yang, J., Yang, J.Y.: Improving protein-atp binding residues prediction by boosting svms with random under-sampling. Neurocomputing **104**, 180–190 (2013)