PROBABILITY BASED CLUSTER EXPANSION OVERSAMPLING TECHNIQUE FOR IMBALANCED DATA

Shaukat Ali Shahee and Usha Ananthakumar

Shailesh J. Mehta School of Management,
Indian Institute of Technology Bombay, Mumbai, India

ABSTRACT

In many applications of data mining, class imbalance is noticed when examples in one class are overrepresented. Traditional classifiers result in poor accuracy of the minority class due to the class imbalance. Further, the presence of within class imbalance where classes are composed of multiple sub-concepts with different number of examples also affect the performance of classifier. In this paper, we propose an oversampling technique that handles between class and within class imbalance simultaneously and also takes into consideration the generalization ability in data space. The proposed method is based on two steps- performing Model Based Clustering with respect to classes to identify the sub-concepts; and then computing the separating hyperplane based on equal posterior probability between the classes. The proposed method is tested on 10 publicly available data sets and the result shows that the proposed method is statistically superior to other existing oversampling methods.

KEYWORDS

Supervised learning, Class Imbalance, Oversampling, Posterior Distribution

1. INTRODUCTION

Class imbalance is one of the most challenging problems in Data Mining [1]. It refers to data sets where one class is under represented compared to another class also referred to as between class imbalance. This phenomenon is commonly seen in many real life applications like fault detection, fraud detection, anomaly detection, medical diagnosis [2][3][4]. Traditional classifiers applied to such imbalanced data fail to classify minority class examples correctly due to its bias towards majority class [5][6][7]. Owing to the large number of potential applications of class imbalance, various methods have been proposed in the literature to address this problem. These methods can be classified into four categories: Sampling based methods, Cost-sensitive learning, kernel-based learning and active learning. Though various approaches exist in literature to handle class imbalance problem, sampling based methods have shown great potential as they attempt to improve data distribution rather than the classifier [8][9][10][11]. Liu et al. [12] have given a number of reasons on why sampling methods are preferred compared to other methods.

Sampling based method is a pre-processing technique that diminishes the class imbalance effect either by increasing the minority class examples or by decreasing the majority class examples

Dhinaharan Nagamalai et al. (Eds) : NATL, CSEA, DMDBS, Fuzzy, ITCON, NSEC, COMIT - 2018
pp. 77–90, 2018. © CS & IT-CSCP 2018 DOI : 10.5121/csit.2018.80607

[13][14]. In this study, we focus on oversampling as undersampling of majority class is not recommended when the dataset has absolute rarity of minority class [15]. In case of oversampling, the number of minority class examples is increased either by random replication of examples or by generating new synthetic examples to minimize the overfitting problem. With regard to synthetic sampling, the synthetic minority over-sampling technique (SMOTE) [8] generates synthetic minority class examples. The method first randomly selects a minority class example and then chooses its k nearest neighbours belonging to the minority class. Synthetic examples are generated between the example under consideration and the selected nearest neighbour example along the line joining them. However, while selecting the nearest neighbours of the minority class, it does not consider the majority class examples and gives equal weight to all nearest neighbours. Oversampling the examples along the line joining the considered example and the selected nearest neighbour leads to the problem of overlapping between the classes [16]. Adaptive synthetic sampling approach for imbalanced learning (ADASYN) [17] adaptively generates synthetic minority class examples based on their weighted distribution of minority class examples according to the level of difficulty in learning. This method generates more synthetic examples corresponding to hard to learn examples and less synthetic instances corresponding to easier to learn examples. Thus the method reduces the bias due to class imbalance and adaptively shifts the classification decision boundary towards hard to learn examples. The crux of the method is to identify hard to learn minority class examples and ADASYN sometimes fails to find the minority class examples that are closer to the decision boundary [9]. Majority weighted minority oversampling technique (MWMOTE) [9] is effective in selecting hard to learn minority class examples but in this method, small concepts present in minority class examples that are located far from majority class examples are not identified. For handling this problem which is also referred to as within class imbalance in literature, various cluster based methods have been proposed in literature [18][10][19]. Cluster Based Oversampling (CBO) [19] is an oversampling technique that can handle between-class imbalance and within-class imbalance simultaneously. However, this method uses random oversampling to oversample the sub-clusters and thus could result in the problem of overfitting.

Further, though class imbalance has been studied well in literature, the simultaneous presence of between class imbalance and within class imbalance has not been addressed enough. In this paper, we propose a method that can reasonably handle between class imbalance and within class imbalance simultaneously. It is an oversampling approach and also considers the generalization ability of the classifier. We have validated our proposed method on publicly available data sets using neural network and compared with existing oversampling techniques that rely on spatial location of minority class examples in the Euclidean feature space.

The remainder of the paper is divided into three sections. Section 2 discuses the proposed method and its various components. Analysis on various real life data sets is presented in Section 3. Finally, Section 4 concludes the paper with future work.

2. THE PROPOSED METHOD

The main objective of the proposed method is in enabling the classifier to give equal importance to all the sub-clusters of the minority class that would have been otherwise lacking due to skewed distribution of the classes. The other objective is to increase the generalization ability of the classifier on the test dataset. Generally, the classifier tries to minimize the total error and when the class distributions are not balanced, minimization of total error gets dominated by

minimization of error due to majority class. Neural network is one such classifier that minimizes the total error. The first objective is achieved by removal of between class and within class imbalance as it helps the classifier in giving equal importance to all the sub clusters. The second objective is realized by enlarging the data space of the sub-clusters as it increases the generalization ability of the classifier on test set.

In the proposed method, the first step is to normalize the input dataset between [0, 1] and then to remove the noisy examples from the dataset. A noisy example is identified based on K-Nearest Neighbour (KNN) of the considered example. In our method, we consider an example to be noisy if it is surrounded by 5 examples of the other class as also being considered in other studies including [9]. Removal of noisy examples helps in reducing the oversampling of noisy examples. After the removal of noisy examples, the concept present in data is detected using model based clustering. The boundary of the sub-clusters is computed based on the equal posterior probability of the classes. Subsequently, the number of examples to be oversampled is determined. Following subsections elaborate the proposed method in detail.

2.1. Locating each sub-concept

Model based clustering is used with respect to the classes to identify the sub-clusters (or sub-concepts) present in the dataset [20]. Model based clustering assumes that data are generated by a mixture of probability distributions in which each component corresponds to a different cluster. We have used MCLUST [21] for implementing the model based clustering. MCLUST is an R package that implements the combination of hierarchical agglomerative clustering, Expectation Maximization (EM) and the Bayesian Information criterion (BIC) for comprehensive cluster analysis.

2.2. Locating the separating hyperplane between the sub-clusters

Model based clustering assumes that data comes from mixture of underlying probability distributions in which each component represents a different group or cluster. In general it considers mixture of multivariate normal distributions. In computing the separating hyperplane between sub-clusters of two classes, the majority class sub-cluster is identified on the basis of the nearest neighbour examples of the minority class sub-cluster. A separating hyperplane is then computed between these two sub-clusters where the posterior probability between these two classes are considered equal. We have

$$p(y = 1|x) = p(y = 0|x) \quad \text{..... (1)}$$

which is same as

$$p(x|y = 1)p(y = 1) = p(x|y = 0)p(y = 0) \text{ ..... (2)}$$

As oversampling handles between class and within class imbalance thus making prior probability equal, equation (2) reduces to

$$p(x|y = 1) = p(x|y = 0) \text{ ..... (3)}$$

Since we assume that distribution is multivariate normal, the final equation of separating hyperplane is

$$(\mu_1 - \mu_2)^T \sum^{-1} x = (\mu_1 - \mu_2)^T \sum^{-1} (\mu_1 + \mu_2)/2 \quad \text{..... (4)}$$

where $x \epsilon R^n$, n is the number of features of the dataset; μ_1 is the mean of the minority class sub-cluster and μ_2 and $\sum$ are respectively the mean and covariance of the majority class sub-cluster.

After computing the separating hyperplane between the sub-clusters, we expand the size of sub-clusters till the boundary of the region given by the hyperplane while maintaining the same structure as shown in Figure 1.

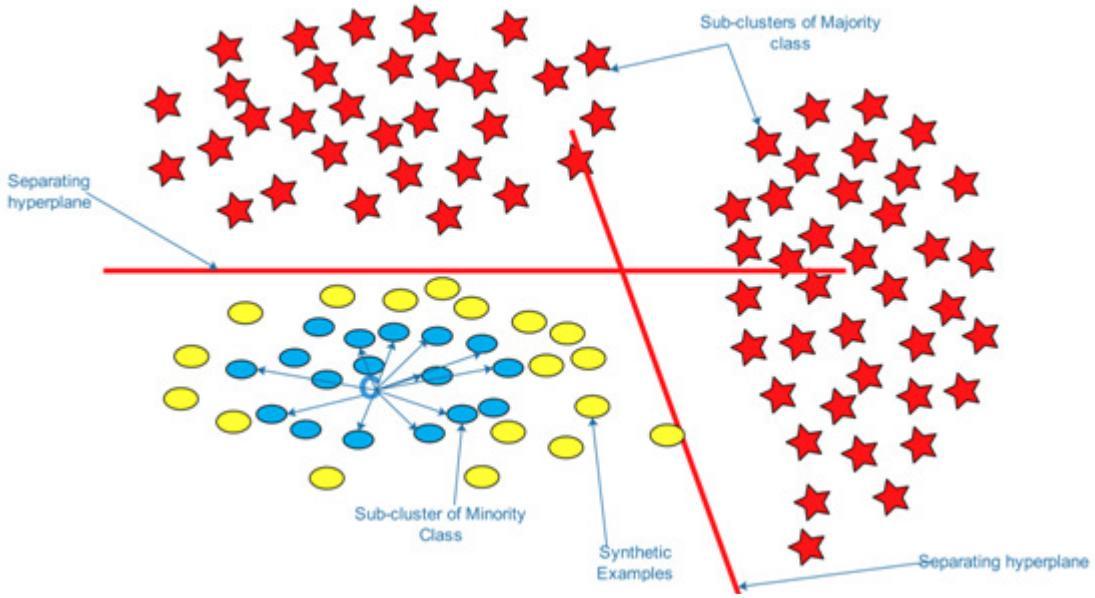

Figure 1. Illustration of synthetic examples being generated in the enclosed region of separating hyperplane

Maintaining the same structure can be done by multiplying the data points by a scalar quantity. The angle between the two vectors are defined as

$$cos\theta = < v_1, v_2 > / ||v_1|| ||v_2|| \quad \text{..... (5)}$$

If we multiply the vectors by a scalar α

$$cos\theta = < \alpha v_1, \alpha v_2 > / ||\alpha v_1|| ||\alpha v_2|| \quad \text{..... (6)}$$

As $cos\theta$ does not change, multiplying the vectors by a scalar does not change the structure of the data. The scalar α is the minimum of the perpendicular distances from the centroid of the sub-cluster to the neighbouring separating hyperplanes. After computing α, the synthetic example is generated by randomly selecting the minority class example u and extrapolating that example by using the following equation

$$Synthetic\ Example = C + (u - C)\alpha \quad \text{.....(7)}$$

In a situation where all the minority class examples of the sub-clusters lie outside the enclosing hyperplane of the sub-cluster as shown in Figure 2, synthetic examples have been generated inside the enclosing region of hyperplane using the following steps.

1. Let C be the centroid of the sub-cluster and vector u lie outside the region.
2. Substituting
   $x = C + t(u - C)$ in equation (4), we get the t value.
3. Generate the uniform number between [0-t]
4. $Synthetic\ Example = C + t(u - C)$ ..... (8)

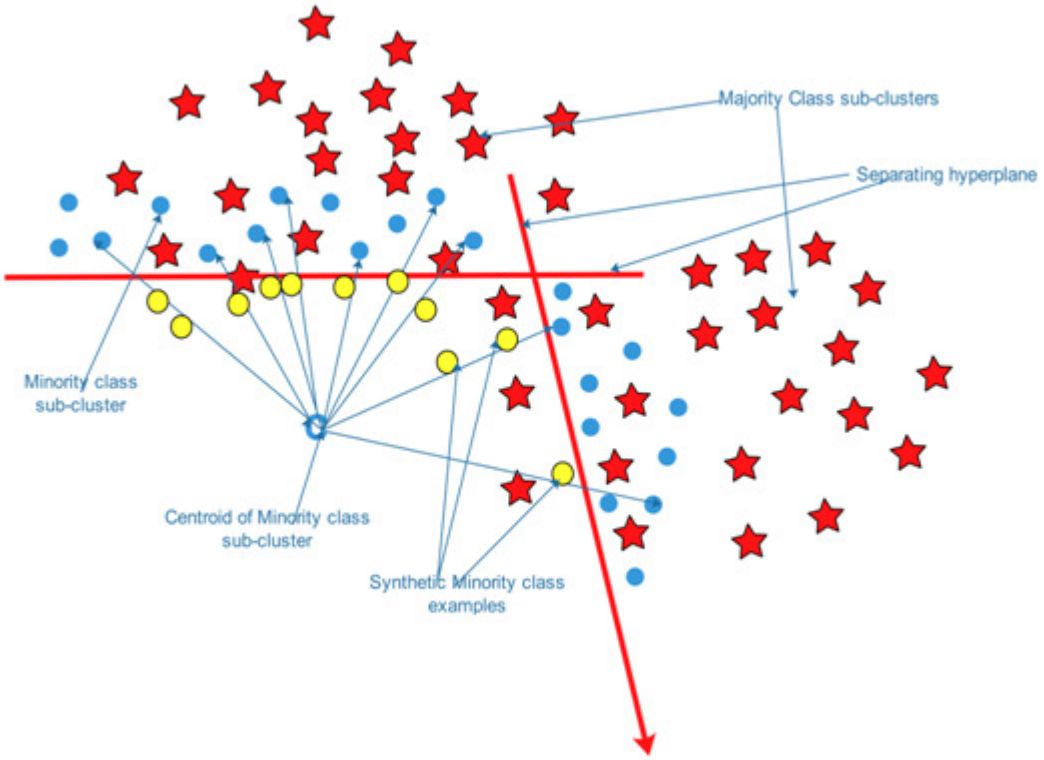

Figure 2. Illustration of synthetic examples being generated when all examples of the sub-cluster lie outside the enclosing hyperplane

2.3. Number of examples to be oversampled

After computing the enclosed region of the sub-clusters given by hyperplane, the method then computes the number of minority class examples to be oversampled. For this, we compute

$$T = Tsmaj/q \quad \text{..... (9)}$$

where T*smaj* is the total number of majority class examples and q is the total number of sub-clusters of minority class. The number of examples to be oversampled in the i^{th} sub-cluster of minority class is given by

$$t_i = T - a_i \quad \text{..... (10)}$$

where a_i is the number of examples already present in the i^{th} sub-cluster.

2.4. Algorithm

Input: Training dataset:

$$S = \{X_i, y_i\}, i = [1 - m];\ X_i \in R^n \ and \ y_i \in \{0,1\}$$

with Tsmaj = Total no of majority class example and Tsmin = Total no of minority class examples.

Output: Oversampled Dataset

1. Remove the noisy examples from the dataset
2. Applying model based clustering with respect to the classes gives
3. A set of *q* minority class sub-clusters {$smin_1$ .....$smin_q$}
4. A set of *r* majority class sub-clusters {$smaj_1$ ..... $smaj_r$ }
5. **For** each minority class sub-cluster {$smin_1$ .....$smin_q$}
6. **For** each majority class sub-cluster {$smaj_1$ ..... $smaj_r$ }
7. Compute the separating hyperplane between the sub-cluster as explained in section 2.2.
8. **EndFor**
9. **EndFor**

10. # Oversampling the minority class sub-clusters
11. T = Tsmaj/q
12. **For** each of the minority class subclusters {$smin_1$ . . . $smin_q$}
13. a_i = size ($smin_i$)
14. t_i = T - a_i
15. **If** $smin_i$ lies completely outside the enclosed region
16. **then** Generate synthetic examples using equation (8) of section 2.2
17. **Else**
18. **While** (t_i >0)
19. S = sample ($smin_i$, t_i) # Select t_i examples from $smin_i$
20. Let s_i be the number of examples lying inside the enclosed region. Generate synthetic examples using equation (7) as explained in section 2.2
21. t_i = t_i - $|s_i|$
22. **EndWhile**
23. **EndElse**
24. **EndIf**
25. **EndFor**

3. COMPARATIVE ANALYSIS

In this section, we evaluate the performance of our proposed method and compare its performance with SMOTE [10], ADASYN [12], MWMOTE [13] and CBO [16]. The proposed method is evaluated on 10 publicly available data sets from KEEL [22] dataset repository. The data sets considered in this study are listed in Table 1.

3.1. Data sets

As this study is about binary classification problem, we have made modifications on yeast dataset as this is multiclass dataset, and the rest of the data sets were taken as it is. In case of yeast dataset, it has 10 classes {MIT, NUC, CYT, ME1, ME2, ME3, EXC, VAC, POX, ERL}.We chose ME3 as the minority class and the remaining classes were combined to form the majority class thus making it an imbalanced dataset. Table 1 represents the characteristics of various data sets used in this study.

Table 1. The Data sets.

Data sets	Total Examples	No. Minority Example	No Majority Exp	Attributes
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
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

3.2. Assessment metrics

Traditionally, performance of the classifier is based on accuracy and error measure that is defined as follows

$$Accuracy = \frac{(TP + TN)}{Total\ Examples}$$

$$Error\ rate = 1 - Accuracy$$

where TP is the number of positive examples classified correctly and TN is the number of negative class examples classified correctly. However, in case of imbalanced datasets, this accuracy measure overestimates the performance of the classifier as this measure could be high even when all or most of the minority class examples are misclassified. To deal with this problem, Haibo [11] proposed various alternative metrics 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

Some of the alternative measures are precision, recall, F-measure and G-mean defined as

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

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

$$F - Measure = \frac{(1+\beta^2) \times \mathrm{Re}\,call \times \mathrm{Pr}\,ecision}{\beta^2 \times \mathrm{Re}\,cal + \mathrm{Pr}\,ecision}$$

$$G - Mean = \sqrt{\frac{TP}{TP + FN} * \frac{TN}{TN + FP}}$$

Here β is a non-negative parameter that controls the influence between precision and recall. In this study we set β = 1 implying that both are of equal importance. Another widely used accuracy measure is Receiving Operating Characteristic (ROC) curve that gives a graphical representation of classifier performance. The area under this curve is known as AUC measure. Since F-measure combines precision and recall, we provide F-measure, G-mean and AUC in this study.

3.3. Experimental setting

In this study, we use feed forward neural network with back propagation in order to evaluate the performance of the proposed method. This particular classifier is chosen in this study as it is one of the classifiers that minimizes the total error and the present algorithm by way of removing the between class and within class imbalance is expected to result in improved performance of the classifier. In the parameter setting, the number of input neurons being used is equal to the number of features and sigmoid function is being used as the activation function with learning rate 0.3. The number of hidden layers being used is one and the number of neurons it contains is equal to (number of features + classes)/2 [23].

Further, in SMOTE, the number of nearest neighbours used is set to 5. In case of ADASYN, we use k = 5 and balance level = 1. In case of MWMOTE, we set the parameters as k1 = 5, k2 = 3, k3 = |sminl/2, cp = 3 and cf(th) = 5.

3.4. Results

The results of the 10 data sets are shown in Table 3 where in each measure, the maximum value has been highlighted. Stratified 5-fold cross validation technique was carried out where oversampling was carried out only in the training data containing four of the folds and the fifth fold was used as the test set. The model was trained on the oversampled training set and applied on the test data set in order to obtain an unbiased estimate of the model. This process was replicated five times and its average is presented in Table 3.

Results in Table 3 show that the proposed method performs better than the other methods in most of the data sets. It can be observed that the AUC value of the proposed method is better than other oversampling methods except ***yeast1*** dataset.

Table 3. F-Measure, G-Mean and AUC for 10 data sets.

Data	**Method**	**F-Measure of Majority Class**	**F-Measure of Minority Class**	**G-Mean**	**AUC**
glass1	SMOTE	0.745	0.619	0.690	0.721
	ADASYN	0.757	0.606	0.683	0.717
	MWMOTE	0.759	0.605	0.684	0.728
	CBO	0.760	**0.624**	**0.699**	0.736
	Prop. Method	**0.803**	0.616	0.694	**0.767**
Pima	SMOTE	0.750	0.631	0.707	0.759
	ADASYN	0.766	0.622	0.703	0.765
	MWMOTE	0.748	0.623	0.701	0.763
	CBO	0.766	0.608	0.691	0.748
	Prop. Method	**0.814**	**0.639**	**0.716**	**0.795**
glass0	SMOTE	0.817	0.681	0.762	0.832
	ADASYN	0.819	0.680	0.761	0.820
	MWMOTE	0.812	0.678	0.758	0.819
	CBO	0.797	0.661	0.743	0.795
	Prop. Method	**0.842**	**0.713**	**0.785**	**0.847**
yeast1	SMOTE	0.785	0.575	0.699	0.772
	ADASYN	0.754	0.582	0.705	0.772
	MWMOTE	0.772	**0.584**	**0.707**	**0.776**
	CBO	0.670	0.550	0.660	0.727
	Prop. Method	**0.808**	0.572	0.692	0.770
vehicle2	SMOTE	0.982	0.950	0.970	0.993
	ADASYN	0.980	0.942	0.963	0.989
	MWMOTE	0.981	0.946	0.967	0.993
	CBO	0.982	0.949	0.968	**0.994**
	Prop. Method	**0.985**	**0.957**	**0.974**	**0.994**

ecoli1	SMOTE ADASYN MWMOTE CBO **Prop. Method**	0.913 0.900 0.914 0.901 **0.938**	0.723 0.719 0.736 0.711 **0.786**	0.822 0.839 0.839 0.829 **0.854**	0.916 0.903 0.916 0.911 **0.937**
Yeast	SMOTE ADASYN MWMOTE CBO **Prop. Method**	0.965 0.955 0.965 0.950 **0.967**	0.737 0.712 0.734 0.689 **0.752**	0.870 **0.898** 0.866 0.892 0.881	0.943 0.938 0.941 0.935 **0.959**
yeast3	SMOTE ADASYN MWMOTE CBO **Prop. Method** left, ..	0.966 0.952 0.966 0.945 **0.968**	0.743 0.695 0.742 0.671 **0.759**	0.870 0.886 0.866 **0.887** 0.878	0.943 0.930 0.938 0.936 **0.951**
yeast-0-5-6-7-9 vs 4	SMOTE ADASYN MWMOTE CBO **Prop. Method**	0.939 0.921 0.942 0.923 **0.948**	0.484 0.458 0.489 0.475 **0.506**	0.694 0.725 0.685 **0.728** 0.684	0.804 0.824 0.819 0.830 **0.851**
yeast-0-2-5-7-9 vs 3-6-8	SMOTE ADASYN MWMOTE CBO **Prop. Metho**	0.972 0.949 0.975 0.948 **0.976**	0.760 0.649 0.768 0.638 **0.787**	0.873 0.868 0.858 0.857 **0.882**	0.920 0.913 0.929 0.913 **0.933**

To test the statistical difference between the proposed method and other existing oversampling methods, we have performed Wilcoxon signed-rank non-parametric test [24] on the metric measures F-measure of minority and majority class, G-mean and AUC. The null and alternative hypotheses are as follows:

H0: The median difference is zero.

H1: The median difference is positive.

The test statistic of the Wilcoxon Signed Rank Test is defined as W = min(W+, W-) where W+ is the sum of the positive ranks and W- is the sum of the negative ranks. As 10 data sets have been used to carry out the test, the W value at a significance of 0.05 should be less than or equal to 10 to reject the null hypothesis. The details of Wilcoxon Signed Rank Test for AUC measure between the proposed method and MWMOTE is given in Table 4. As we can see from this table that W+ = 52, W- = 3, W = min(W+, W-) => W = 3, we reject the null hypothesis and conclude that the proposed method is better than MWMOTE in terms of AUC measure.

Table 4. Wilcoxon Signed Rank Test of AUC between the proposed method and MWMOTE.

Dataset	Proposed method	MWMOTE	Difference	Rank
glass1	0.767	0.728	0.039	10
Pima	0.795	0.763	0.032	8.5
glass0	0.847	0.819	0.028	7
yeast1	0.770	0.776	-0.006	3
vehicle2	0.994	0.993	0.001	1
ecoli1	0.937	0.916	0.021	6
Yeast	0.959	0.941	0.018	5
yeast3	0.951	0.938	0.013	4
yeast-0-5-6-7-9 vs 4	0.851	0.819	0.032	8.5
yeast-0-2-5-7-9 vs 3-6-8	0.933	0.929	0.004	2
W+ = 52, W- = 3, W = min(52, 3) = 3				

For space consideration, we present just the summary of Wilcoxon Signed Rank Test between the proposed method and other oversampling methods for various metric measures in Table 5. From this table, it can be seen that the proposed method is statistically significantly better than the other oversampling methods in terms of AUC and F-measure of both majority and minority class, although in case of G-mean, the proposed method does not seem to outperform the other oversampling methods. Though it is desirable that any algorithm performs well on all the measures, as stated in [25], AUC is a measure that is not sensitive to the distribution of the two classes thus making it suitable as a performance measure for the imbalanced problem.

Table 5. Summary of Wilcoxon signed rank test between our proposed method and other methods

Method	Proposed Method	Metric Measure
SMOTE	W+ = 55, W- = 0, W = 0	F-Measure of Majority class
	W+ = 52, W- = 3, W = 3	F-Measur of Minority class
	W+ = 36, W- = 19, W = 19	G-mean
	W+ = 53, W- = 2, W = 2	AUC
ADASYN	W+ = 55, W- = 0, W = 0	F-Measure of Majority class
	W+ = 53.5, W- = 1.5, W = 1.5	F-Measur of Minority class
	W+ = 32.5, W- = 22.5, W = 22.5	G-mean
	W+ = 54, W- = 1, W = 1	AUC
MWMOTE	W+ = 55, W- = 0, W = 0	F-Measure of Majority class
	W+ = 53, W- = 3, W = 3	F-Measur of Minority class
	W+ = 47.5, W- = 7.5, W = 7.5	G-mean
	W+ = 52, W- = 3, W = 3	AUC
CBO	W+ = 55, W- = 0, W = 0	F-Measure of Majority class
	W+ = 53.5, W- = 1.5, W = 1.5	F-Measur of Minority class
	W+ = 40, W- = 15, W = 15	G-mean
	W+ = 55, W- = 0, W = 0	AUC

4. CONCLUSION

In this paper, we have proposed a method that can handle between class imbalance and within class imbalance simultaneously. The proposed method applies model based clustering with respect to each of the classes to identify the sub-concepts present in the dataset. Then it computes the separating hyperplane that satisfies the equal posterior probability between the sub-concepts.

It then generates the synthetic examples while maintaining the structure of the original dataset in the enclosed region given by the hyperplane thus increasing the generalization accuracy of the classifier.

The proposed method has been evaluated on 10 publicly available data sets and the results clearly show that the proposed method increases the accuracy of the classifier. However, the limitation of the proposed method is that it gets influenced by the nearest majority class sub-clusters in the expansion of the minority sub-clusters which could be extended as future work. Another possible extension could be in modifying the computation of separating hyperplane by including majority class clusters that are located far from minority class clusters.

REFERENCES

[1] Q. Yang et al., "10 Challenging Problems in Data Mining Research," Int. J. Inf. Technol. Decis. Mak., vol. 5, no. 4, pp. 597–604, 2006.

[2] M. Galar, A. Fernández, E. Barrenechea, and F. Herrera, "EUSBoost : Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling," Pattern Recognit., vol. 46, no. 12, pp. 3460–3471, 2013.

[3] S. García and F. Herrera, "Evolutionary Undersampling for Classification with Imbalanced Datasets: Proposals and Taxonomy," Evol. Comput., vol. 17, no. 3, pp. 275–306, 2009.

[4] S. Vajda and G. A. Fink, "Strategies for training robust neural network based digit recognizers on unbalanced data sets," Proc. - 12th Int. Conf. Front. Handwrit. Recognition, ICFHR 2010, no. November 2010, pp. 148–153, 2010.

[5] S. Maldonado and J. Lopez, "Imbalanced data classification using second-order cone programming support vector machines," Pattern Recognit., vol. 47, no. 5, pp. 2070–2079, 2014.

[6] D. J. Yu, J. Hu, Z. M. Tang, H. Bin Shen, J. Yang, and J. Y. Yang, "Improving protein-ATP binding residues prediction by boosting SVMs with random under-sampling," Neurocomputing, vol. 104, pp. 180–190, 2013.

[7] C. Y. Yang, J. S. Yang, and J. J. Wang, "Margin calibration in SVM class-imbalanced learning," Neurocomputing, vol. 73, no. 1–3, pp. 397–411, 2009.

[8] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," J. Artif. Intell. Res., pp. 321–357, 2002.

[9] S. Barua, M. M. Islam, X. Yao, and K. Murase, "MWMOTE--majority weighted minority oversampling technique for imbalanced data set learning," Knowl. Data Eng. IEEE Trans., vol. 26, no. 2, pp. 405–425, 2014.

[10] S.-J. Yen and Y.-S. Lee, "Cluster-based under-sampling approaches for imbalanced data distributions," Expert Syst. Appl., vol. 36, no. 3, pp. 5718–5727, 2009.

[11] H. Han, W. Wang, and B. Mao, "Borderline-SMOTE : A New Over-Sampling Method in," pp. 878–887, 2005.

[12] A. Liu, J. Ghosh, and C. E. Martin, "Generative Oversampling for Mining Imbalanced Datasets," Int. Conf. data Min., pp. 66–72, 2007.

[13] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data," SIGKDD Explor. Newsl., vol. 6, no. 1, pp. 20–29, 2004.

[14] A. Estabrooks, T. Jo, and N. Japkowicz, "A multiple resampling method for learning from imbalanced data sets," Comput. Intell., vol. 20, no. 1, pp. 18–36, 2004.

[15] G. M. Weiss, "Mining with Rarity: A Unifying Framework," SIGKDD Explor., vol. 6, no. 1, pp. 7–19, 2004.

[16] H. He and E. A. Garcia, "Learning from imbalanced data," IEEE Trans. Knowl. Data Eng., vol. 21, no. 9, pp. 1263–1284, 2009.

[17] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in Proceedings of the International Joint Conference on Neural Networks, 2008, no. 3, pp. 1322–1328.

[18] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "DBSMOTE: Density-based synthetic minority over-sampling technique," Appl. Intell., vol. 36, no. 3, pp. 664–684, 2012.

[19] T. Jo and N. Japkowicz, "Class imbalances versus small disjuncts," ACM Sigkdd Explor. Newsl., vol. 6, no. 1, pp. 40–49, 2004.

[20] C. Fraley and a E. Raftery, "Model-based clustering, discriminant analysis, and density estimation," J. Am. Stat. Assoc., vol. 97, no. 458, pp. 611–631, 2002.

[21] C. Fraley and A. E. Raftery, "MCLUST: Software for model-based cluster analysis," J. Classif., vol. 16, no. 2, pp. 297–306, 1999.

[22] J. Alcalá-Fdez et al., "KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework," J. Mult. Log. Soft Comput., vol. 17, no. 2–3, pp. 255–287, 2011.

[23] H. Guo and H. L. Viktor, "Boosting with Data Generation: Improving the Classification of Hard to Learn Examples.," Iea/Aie, vol. 3029, pp. 1082–1091, 2004.

[24] A. Richardson, "Nonparametric Statistics for Non-Statisticians: A Step-by-Step Approach by Gregory W. Corder, Dale I. Foreman," Int. Stat. Rev., vol. 78, no. 3, pp. 451–452, 2010.

[25] I. Nekooeimehr and S. K. Lai-Yuen, "Adaptive semi-unsupervised weighted oversampling (A-SUWO) for imbalanced datasets," Expert Syst. Appl., vol. 46, pp. 405–416, 2016.

AUTHORS

Shaukat Ali Shahee received the Bachelor's degree in Mathematics honors from Patna University and Master's degree in computer application from the West Bengal University of Technology, India. He is currently pursuing PhD at Shailesh J. Mehta School of Management, Indian Institute of Technology Bombay. His research interests include data mining, machine learning and applied statistics.

Prof. Usha Ananthakumar received PhD degree in Statistics from Indian Institute of Technology Bombay, India. She is currently a Professor at Shailesh J. Mehta School of Management, Indian Institute of Technology Bombay. Her research interests include data mining, machine learning, applied Statistics and multivariate data analysis.