International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-2, July 2019

CoGBUS- Center of Gravity Based Under Sampling Method for Imbalanced Data Classification

Shidha M V, T Mahalekshmi, Sabu M K

Abstract: ***Learning of class imbalanced data becomes a challenging issue in the machine learning community as all classification algorithms are designed to work for balanced datasets. Several methods are available to tackle this issue, among which the resampling techniques- undersampling and oversampling are more flexible and versatile. This paper introduces a new concept for undersampling based on Center of Gravity principle which helps to reduce the excess instances of majority class. This work is suited for binary class problems. The proposed technique –CoGBUS- overcomes the class imbalance problem and brings best results in the study. We take F-Score, GMean and ROC for the performance evaluation of the method.***

Index Terms**: *Center of Gravity, F-Score, GMean, ROC, undersampling.***

I. INTRODUCTION

The class imbalance problems refer the classification tasks in which one class of data outnumbers the other classes. This data skewness behavior affects badly in the prediction of rare class data. Many real world problems, their class distribution belong to imbalanced sets and more importantly, prediction accuracy is crucial for the rare class events. Such problems include medical diagnoses for rare diseases, fraud detection in banking sectors, Protein-ligand affinity in drug discovery process etc.[1]-[5]. There is a wide variety of strategies have been proposed to solve the issues regarding imbalanced datasets. The internal level strategies deal with the design of new classification algorithms or the modification of existing ones so as to capable for dealing imbalance factor [6], [7]. The external level approaches use data balancing techniques such as undersampling and oversampling to solve the imbalance problem [8], [9]. Hybrid methods which are a combination of internal and external strategies are also available [10], [11] to work with imbalanced data. This paper uses undersampling concept that reduces the size of majority class samples to almost equal or nearer to the count of minority class. The proposed method aims at eliminating both class imbalance issue and the generation of noisy samples. The content of study is organized in the following fashion. In section 2, related work is summarized and some popular undersampling methods are specified. Section 3 describes the proposed undersampling technique and its working principle in a detailed manner. Section 4 discloses the datasets and the performance criteria for their evaluation. The experimental results are discussed in section 5 which is followed by section 6 for giving conclusion.

Revised Manuscript Received on 30 July 2019.
***** Correspondence Author
Shidha M V*, Research Scholar, Bharathiar University, Coimbatore, India.
T Mahalekshmi, Professor & Principal, SNIT, Kollam, India.
Sabu M K, Associate Professor, Department of Computer Applications, CUSAT, India.

II. RELATED WORKS

The ultimate objective of any resampling technique is the improvement over the classification results produced by the data which has not been resampled. Application of resampling techniques converts an imbalanced dataset to balanced dataset before it is subjected to classification process. This is achieved either by reducing excess majority class samples viz. undersampling or by generating more minority class samples by the process- oversampling. Several types of undersampling and oversampling strategies are made available [12]. Of these two resampling strategies, undersampling is a better choice over oversampling [3], [13] as oversampling increases the likelihood of overfitting during the model construction process. Nevertheless the undersampling strategy leads to the elimination of useful data in the majority class. To open up a new thought in undersampling procedure, we propose an attractive method based on center of gravity concept. This method finds representatives for a set of majority class samples which are later replaced by these new representatives. Thus a reduction in size made possible and in fact the process becomes equivalent to undersampling. This new strategy helps to reduce the risk of losing relevant information often caused by undersampling procedure. A brief description on some undersampling strategies is narrated below. RUS- Random Undersampling is the simplest undersampling strategy that will do the balancing at high speed [14]. The reduction of samples is done by removing majority class samples which are selected at random. Several issues noticed during the processing of RUS. Firstly, random selection of samples gives no guarantee on choosing less relevant samples. Sometimes important samples are also picked up for removal due to its random selection behaviour. Secondly, no consistency showed in the output during the execution of same dataset. Each time when the rebalancing process worked out on the same dataset, output varies depends on the samples chosen.

Since it is not reliable, we do not consider RUS for the performance comparison of our proposed work. CUS-Cluster based Under Sampling technique uses the idea of cluster centroids to represent the cluster. This overcomes the issues caused by random under sampling strategy. Each cluster centroid which is based on the mean of similar data in the same cluster is derived using K-Means algorithm [15] could be used to represent the data in the whole group. This reduces the size of majority class.

NearMiss- This strategy selects samples from the majority class whose average distance of the *k* nearest samples of the minority class is small [16]. This means that NearMiss give priority for those majority class samples which are closer to minority groups for elimination. In this paper, we use CUS and NearMiss for the performance comparison of CoGBUS, the proposed undersampling strategy.

III. PROPOSED WORK

A. *The Theme Behind*

A new approach for undersampling has been proposed in this study. This is based on the Center of Gravity (CoG) line method [17] and is useful for dimensionality reduction when size of data is very large. Since the effort is for undersampling which is a process of reducing the count of majority class data to balance its count with the minority class objects, the concept of CoG has applied to majority class objects. It generates synthetic samples for a group of majority samples and later these sets of data will be replaced by the newly generated samples. COG is an imaginary point around which the center of an object's weight lies. This makes possible for the points in the plane can be separable by a line based on their CoG. This line is a virtual line which lies in the middle of equally weighted and distinct set of points.

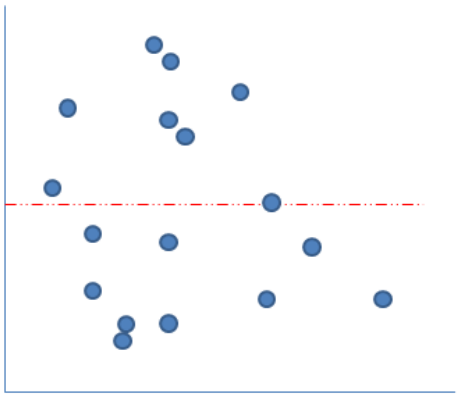

Fig1: Centre of Gravity line representation

The CoG line of a set of points in the plane is positioned in such a way that the sum of all perpendicular distances from the points to this line is zero.

B. *Basic Terms and Terminology*

This work focuses on the binary classification for imbalanced datasets. Obviously, the dataset under consideration has two class values [1,0]. We assume that the dataset consists of *p* number of features. The variables *pos* and *neg* denote the count of minority class samples (class value 1) and majority class samples (class value 0) respectively. The imbalance ratio of the dataset is represented by *ir* which is obtained by *ir* = neg / pos. According to CoG method, a CoG line is created among the samples of same class, based on the distance between these samples. Since this strategy is usually applied on cluster of samples, we make subgroups in class 0 samples. Each subgroup G_z, consists of *ir* number of samples. The number of subgroups $G_1,G_2,G_3,\ldots$, Gm of majority class is determined by the value *m= group_count* which is obtained by *neg* / *ir*. Each group is processed and a new feature vector is synthesised for them based on the distance values between the samples in a group. These differences are added together and the features of minimum sum are selected as the features of new CoG vector. This value is calculated using the following formula. This process is repeated for every sample in each subgroup.

$$\sum_{j=1}^{ir} \left| \boldsymbol{d}\left(ai(X), ai(samplej)\right) \right|, \text{ i= 1,2,….,p}$$

where *X* is a sample in the current subgroup G_z, z = 1,2,…., *group_count*. Initially, *X* is set as the first instance of current subgroup. The distance between *X* and other samples are calculated and a sum is generated for each attribute. This process is repeated by changing *X* value, so that *ir* number of sum vectors are generated for a group. Then a CoG vector is derived for the group from the least sum.

That is,

$$X_{\text{ind}} = Index\left(Min\left(\sum_{j=1}^{ir} \left| \boldsymbol{d}\left(ai(X), ai(samplej)\right) \right|\right)\right)$$

$$ai\_new = ai(Sample\_X_{\text{ind}})$$

$$CoG(G_z) = [a_1\_new, a_2\_new, \ldots\ldots, a_p\_new]$$

where X_{ind} gives the index number of sample *X* that provides least sum on the i^{th} feature and CoG (G_z) is the new feature vector obtained for the subgroup G_z with *p* number of features. As an example, if a dataset consists of 470 instances in which *neg* holds the count 400 and *pos* has 70. The imbalance ratio, *ir* is 5, so that 400 majority samples are divided into 80 subgroups, with 5 samples in each. Thereafter CoGBUS is applied to the groups, representative CoG feature vectors are generated. In this case, the eighty groups provide 80 new synthetic vectors. These are combined with 70 minority class samples results a well-balanced dataset.

C. *Design View*

The following methodology has been used to conduct the prediction of class labels and evaluates the impact of CoGBUS in handling imbalanced datasets.

Step1: Input an imbalanced dataset- DF

Step2: Apply preprocessing to convert categorical attributes to numerical attributes.

Step3: Group the samples in DF into two- a set of class '1' samples, DF1 and a set of class '0' samples, DF0.

Step4: Apply COGBUS to samples in DF0.

Step5: CoGBUS returns CoG feature vectors based on the count of subgroups in DF0.

Step6: Merge the new representative samples of class0 obtained in step5 with DF1.

Step7: Split the resultant dataset to Train set and Test set in the ratio 70:30.

Step8: Use SVM classifier to learn and fit the train data.
Step9: Use the derived model to make prediction for the test set data.
Step10: Evaluate the impact of CoGBUS resampling process in the classification of positive samples using F-Score, GMean and ROC values.
Schematic representation of this methodology has shown in Figure 2. The pseudo code of CoGBUS is given below.

ALGORITHM: CoGBUS
Input: *DF*- Imbalanced dataset
pos- count of class '1' (positive) samples
neg- count of class '0' (negative) samples
p- no. of features
Method:
Begin
// Step1: Split the samples in *DF*
DF1← {x/ x ε *DF* and class(x) =1}
DF0← {x/ x ε *DF* and class(x) =0}
//Step2:Determine imbalance ratio *ir* and the No. of subgroups made with *DF0*, *group_count*
ir← neg/pos
group_count← neg/ir
//Step3:Initialize 2-D lists *min_sum*, *index_no*, and *cog_vector* with dimension [*group_count*][*p*] and *sum_att* with order [*ir*][*p*]
sum_att[][]←0
min_sum[][]←0
index_no[][]←0
cog_vector[][]←0
//Step4: For each subgroup *Gz* (z = 1,2,…., *group_count*) in DF0
Compute CoG feature vector:
a) Find sum of distance between the samples in *Gz* and *X*. Initially *X*= sample1
sum_att[][]← $\sum_{j=1}^{ir} \left| d\left(ai(X), ai(samplej)\right) \right|$, i= 1,2,….,p
b) Repeat step (a) by changing *X* to sample$_2$ until it becomes sample$_{ir}$
c) Find minimum of sums generated for each *X*
min_sum[][]←$Min(\sum_{j=1}^{ir} \left| d\left(ai(X), ai(samplej)\right) \right|)$
d) Store index_no of *X* which provides minimum sum for each attribute a_i, i=1,2,..,p
index_no[][]←index_no(*X*)
e) Derive CoG feature vectors using values of a_i, i=1,2,..,p
cog_vector[][]←*DF*0[index_no][i]

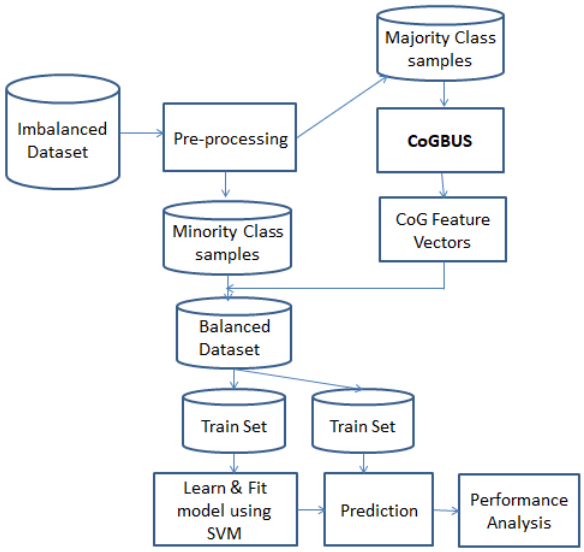

Fig 2: Process flowchart

In step 1 of the algorithm, total samples of selected binary class dataset are splitted into two sub datasets, DF1 of class 1 samples and DF0 of class 0 samples. Step2 calculates imbalance ratio and the number of subgroups needed with DF0. Data structures required for different intermediate results are defined in step3. Step4 describes the actions required for generating CoG vectors. Finally a 2-D list of CoG vectors is returned as output.

IV. DATASETS

A total of 7 imbalanced datasets are considered for the study. They are accessed from KEEL dataset repository available at https://sci2s.ugr.es/keel/imbalanced.php. Binary class data are taken for this work. In the case of multi class datasets, they are converted to binary class data by merging the classes together to form one majority class and one minority class. The characteristics of the selected datasets are given in Table I. The imbalance ratio, *ir* varies from 2 to 39 and three of them viz. Yeast, Winequality, Abalone are highly imbalanced with a ratio of 28, 29 and 39 respectively. SVM classifier has been used to learn and fit the data [18]. Usually performance metrics like average accuracy misleading the assessment in the case of imbalanced class problems [19]. Two basic measurements are Precision (TP/[TP+FP]) and Recall (TP/[TP+FN]). Normally we need to have high confidence that observations predicted as positive are actually positive (high precision) and a high detection rate of positives (high recall). But often precision and recall share an inverse relationship; another measurement based on both called F-Score is being used. It is evaluated by the equation (2*precision*recall)/(precision+ recall). GMean is another effective measurement for imbalanced class problems which is obtained by SQRT (recall_+ve * recall_-ve). A well-accepted ranking measure of imbalanced dataset is AUC- Area Under ROC Curve which exhibits the trade-offs between true positive and false positive error rates [20]. However to determine a general ranking among the undersamplers, the above measures F-Score, GMean and ROC have been used in this work.

TABLEI: Dataset Characteristics

Dataset	# Samples *neg+pos*	# Attributes (*p*)	# Majority class samples (*neg*)	# Minority class samples (*pos*)	Imbalance Ratio (*ir*)
Yeast4	1484	9	1433	51	28.1
Thoracic	470	17	400	70	5.7
Wisconsin	683	10	444	239	1.9
Vowel	988	14	898	90	9.9
Winequality	1599	12	1546	53	29.2
Abalone17	2339	9	2280	58	39.3
Diabetes	768	9	500	268	1.9

V. RESULTS AND DISCUSSIONS

The experiment is carried out using Python's Jupyter Notebook. In the selected datasets, all categorical attributes have been changed to numerical attributes using LabelEncoder facility of panda's dataframe. Each imbalanced dataset has subjected to CoGBUS in order to make them balanced. Table II shows the count of minority and majority samples in the case of unbalanced state as well as balanced state which is obtained through CoGBUS undersampling strategy.

TableII: Change in count of negative samples by CoGBUS

Dataset	Unbalanced data distribution			Balanced data distribution After CoGBUS		
	+'ve	-'ve	Total	+'ve	-'ve	Total
Yeast4	51	1433	1484	51	51	102
Wisconsin	239	444	683	239	222	461
Vowel	90	898	988	90	99	189
Thoracic	70	400	470	70	80	150
Winequality	53	1546	1599	53	53	106
Abalone17	58	2280	2339	58	58	116
Diabetes	268	500	768	268	250	518

Fig 3.a shows the data distribution of Diabetes dataset before the resampling process and Fig 3.b exhibits the change in distribution after CoGBUS. Initially, it has a total of 768 samples in which majority class contains 500 instances and 268 found with minority class. The number of total samples are reduced to 518 by CoGBUS in which minority class samples exist as same as before, but the 500 majority class samples are reduced to 250.

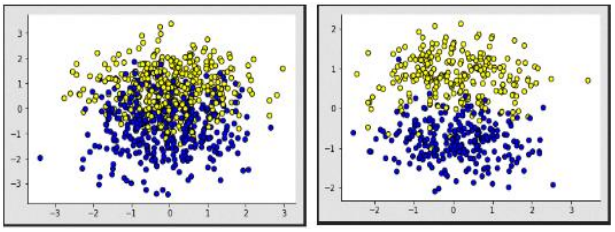

Fig 3: Data Distribution of diabetes a) Unsampled b) After CoGBUS

Three metrics useful for imbalanced datasets- F-Score, GMean, ROC- are used for the evaluation of the classifier accuracy. To assess the performance of new principle, CoGBUS, it is compared with two popular undersampling strategies CUS and NearMiss. Table III shows the F-Score values obtained for seven selected datasets with their unbalanced state and balanced state obtained through CoGBUS, CUS and NearMiss. Figure 4 is its bar chart representation.

TableIII: F-Score values

Dataset	F-Score values			
	Unsampled	CoGBUS	CUS	NearMiss
Yeast4	0.00	**0.89**	0.76	0.60
Wisconsin	0.94	**0.96**	0.96	0.95
Vowel	0.83	**1.00**	0.98	1.00
Thoracic	0.00	0.63	**0.68**	0.59
Winequality	0.00	0.61	**0.77**	0.69
Abalone17	0.00	0.73	**0.79**	0.67
Diabetes	0.62	**0.77**	0.66	0.73

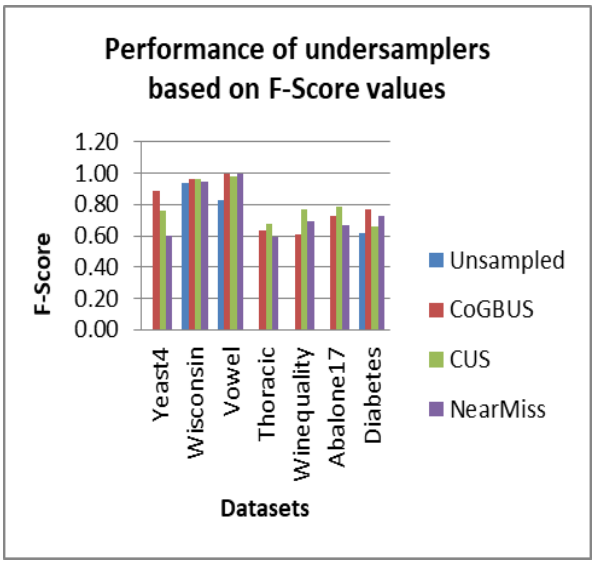

Fig4: Comparison of CoGBUS based on F-Score values

CoGBUS performs best for four datasets. It is just behind CUS for the other three datasets.

TableIV: GMean values

Dataset	GMean values			
	Unsampled	CoGBUS	CUS	NearMiss
Yeast4	0.00	**0.89**	0.77	0.61
Wisconsin	0.95	**0.96**	0.96	0.95
Vowel	0.91	**1.00**	0.98	1.00
Thoracic	0.00	0.68	**0.69**	0.64
Winequality	0.00	0.66	**0.78**	0.73
Abalone17	0.00	0.74	**0.79**	0.71
Diabetes	0.70	**0.77**	0.66	0.74

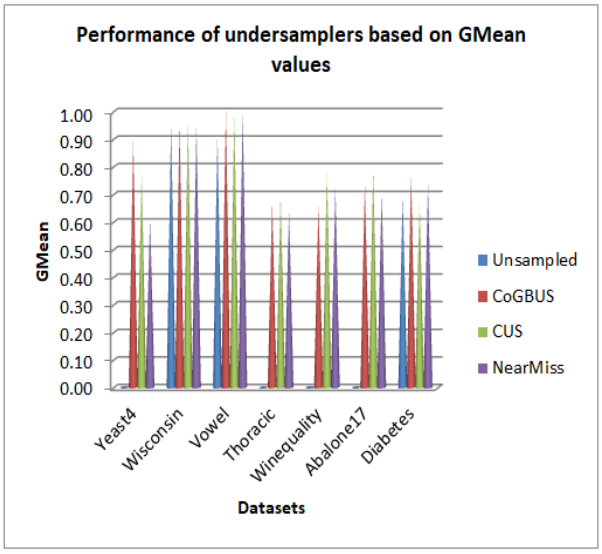

Fig5: Comparison of CoGBUS based on GMean values

TableV: ROC values

Dataset	AUC_ROC values			
	Unsampled	CoGBUS	CUS	NearMiss
Yeast4	0.500	**0.900**	0.777	0.615
Wisconsin	0.956	0.957	**0.965**	0.951
Vowel	0.915	**1.000**	0.983	1.000
Thoracic	0.500	**0.716**	0.691	0.659
Winequality	0.500	0.719	**0.781**	0.750
Abalone17	0.500	0.745	**0.799**	0.737
Diabetes	0.719	**0.774**	0.665	0.739

With the case of GMean and ROC also, CoGBUS proves that it is a reliable method for doing undersampling process. It outperforms over CUS and NearMiss for 4 datasets. Refer figures 5 and 6, Tables IV and V for the proof of values.

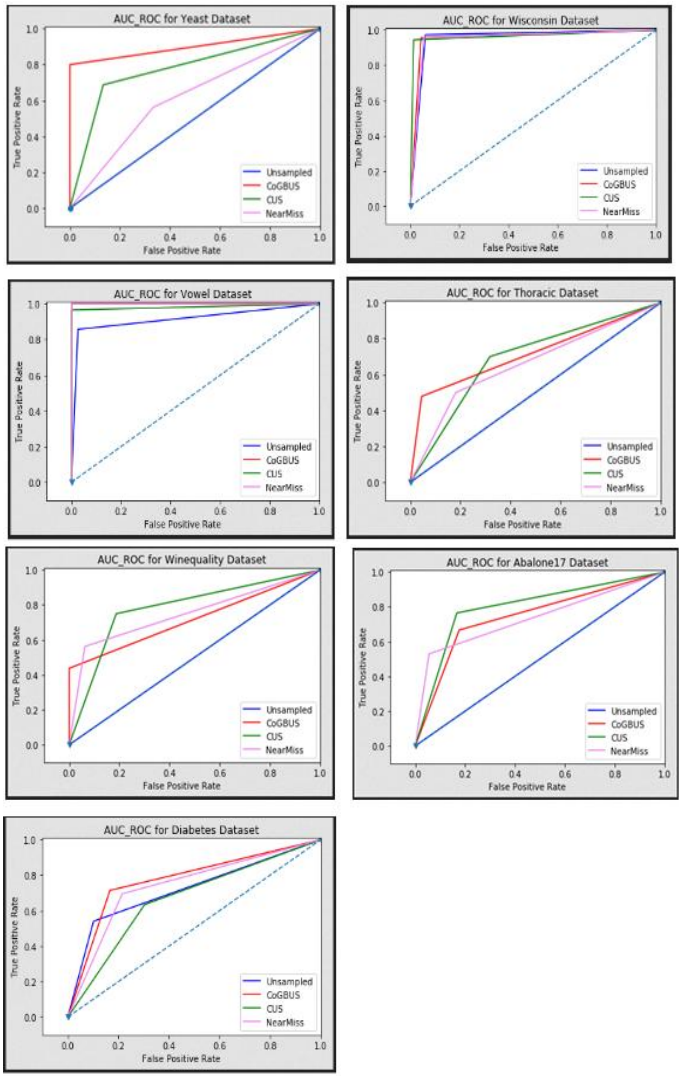

Fig6: Comparison of CoGBUS based on ROC values

VI. CONCLUSION

This study has proposed a new undersampling concept that rebalances an imbalanced dataset based on the concept of Center of Gravity principle. This is achieved by setting a CoG line for a set of points in such a way that the sum of all perpendicular distances from the points to this line is zero. Experimental results prove that CoGBUS offers a powerful resampling strategy in comparison with existing popular undersampling algorithms. We succeeded to eliminate the information loss problem often happens with undersampling by introducing derived samples based on CoG. This method is advisable when the imbalance ratio is greater than 2.

REFERENCES

1. Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16:321-357

2. Kotsiantis, S., Pintelas, P., Anyfantis, D., and Karagiannopoulos, M. (2007). Robustness of learning techniques in handling class noise in imbalanced datasets.
3. Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., and Herrera, F. (2012). A review on ensembles for the class imbalance problem: Bagging, boosting, and hybrid-based approaches.
4. N. Japkowicz and S. Stephen. The class imbalance problem: A systematic study. Intelligent data analysis, 6(5):429–449, 2002.
5. G. M. Weiss and F. Provost, "The effect of class distribution on classifier learning: An empirical study," Technical Report ML-TR-44, Department of Computer Science, Rutgers University, 2001.
6. R. Barandela, J.S. Sanchez, V. Garcia, E. Rangel, Strategies for learning in class imbalance problems, Pattern Recognition 36 (3) (2003) 849–851.
7. W. Lin, J.J. Chen, Class-imbalanced classifiers for high-dimensional Briefings in Bioinformatics 14 (1) (2013) 13–26.
8. A. Estabrooks, T. Jo, N. Japkowicz, A multiple resampling method for learning from imbalanced data sets, Computational Intelligence 20(1) (2004) 18–36.
9. G.E.A.P.A. Batista, R.C. Prati, M.C. Monard, A study of the behaviour of several methods for balancing machine learning training data, SIGKDDExplorations 6 (1) (2004) 20–29.
10. P. Domingos, Metacost: a general method for making classifiers cost–sensitive, in: Proceedings of the 5th International Conference on Knowledge Discovery and Data Mining (KDD'99), 1999, pp. 155–164.
11. Rokach, Ensemble-based classifiers, Artificial Intelligence Review 33 (1) (2010) 1–39.
12. Ajinkya More, "Survey of resampling techniques for improving classification performance in unbalanced datasets".
13. J. Blaszczynski , J. Stefanowski , Neighborhood sampling in bagging for imbalanced data, Neurocomputing 150 (2015) 529–542.
14. Sei_ert, C., Khoshgoftaar, T., Van Hulse, J., Napolitano, A.: Rusboost: A hybrid approach to alleviating class imbalance. IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans 40(1), 185-197 (2010).
15. J.A. Hartigan , M.A. Wong , Algorithm AS 136: a k-means clustering algorithm, J. R. Stat. Soc., Ser. *C* 28 (1) (1979) 100–108.
16. 16. Inderjeet Mani, I Zhang. KNN approach to unbalanced data distributions: a case study involving information extraction. In proceedings of workshop on learning from imbalanced datasets, 2003.
17. Charlotte Bean, Chandra Kambhampati, Autonomous Clustering Using Rough Set Theory: Article in International Journal of Automation and Computing · January 2008.
18. Tang, Y., Zhang, Y.Q., Chawla, N.V., Krasser, S.: Modeling for highly imbalanced classification, J. latex class files. **1**(11) (2002).
19. Shidha M.V, Mahalsekshmi T, Assessing the impact of Instance Imbalance in the prediction of Class labels, in: Proceedings of the National Conference on Advanced Computing (NCAC'19) in Bharathiar University, January 2019.
20. Nitesh V. Chawla, "Data mining for imbalanced datasets: An overview".

AUTHORS PROFILE

Ms. Shidha M V is pursuing Ph.D in Bharathiar University, Coimbatore, India. She obtained her Master of Computer Applications degree from Mother Teresa Women's University, Kodaikanal in 2001 and MPhil from Vinayaka Mission University, Salem in 2009. Her area of specialization is Data mining. She has been working as Assistant Professor in Department of Computer Applications of Federal Institute of science and Technology, Angamaly, Kerala since December 2014. She has 18 years of teaching experience and had attended several national and international seminars and conferences. She also has article publications in various journals.

Dr. T Mahalekshmi has obtained her Ph.D degree in Computer Science from University of Kerala, India in 2007 and Master of Science from School of Computer Science, University of Minnesota, USA in 1985. She bagged 1st Rank in Master of Science in Mathematics from University of Kerala in 1983. Her specialization areas in research are Computational Biology and Soft Computing Algorithms. She is working as a Principal and Professor, Department of Computer Applications, Sree Narayana Institute of Technology, Kollam, Kerala since June 2007. Her writings have appeared in numerous national and international journals. She was an author for few books such as "Data Structures in C", published by PHI Learning Private Ltd, Delhi 2009 and a text book "Computer Graphics" based on the 8th Module of 'B' Level- a Master of Computer Application Level course published in 1999.

Dr. Sabu M K received the M.C.A degree from Government Engineering College, Thrissur, Kerala and Ph.D degree in Computer Science from School of Computer Sciences, Mahatma Gandhi University, Kottayam, Kerala, India. He is currently an Associate Professor in the Department of Computer Applications, Cochin University of Science and Technology (CUSAT), Kochi, India. His research interests include Data Mining, Rough Set Theory, Optimization Techniques and Machine Learning. He has numerous articles in well-known journals and chaired many conferences. He is very active as a Resource Person, Examiner, Reviewer, Visiting Faculty member and member- Board of Studies of Mahatma Gandhi University, Kottayam and several colleges.