International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161
©2021 INPRESSCO®, All Rights Reserved
Available at http://inpressco.com/category/ijcet

Research Article

Data Balancing Technique for Multi-Class Imbalanced Problems

Deore Mrunalee C and J.R. Mankar

Department of Computer Engineering K. K. Wagh Institute of Engineering & Research, Nashik, Maharashtra

Received 10 Nov 2020, Accepted 10 Dec 2020, Available online 01 Feb 2021, **Special Issue-8 (Feb 2021)**

Abstract

The imbalanced dataset contains skewed distribution of data. Such data distribution generates difficulties for machine learning algorithms. These algorithms also fail to generate accurate results in case of data imbalance, overlapping of class boundaries and hybrid datasets. Various techniques proposed in a literature to balance a dataset using oversampling or under sampling methods. The study of these techniques is done independently. A little work has been done with the combined study of these two techniques. The proposed system focuses on the study and implementation of oversampling and under-sampling together to balance a dataset. The technique is generalized for hybrid datasets. Cluster based under sampling approach is used followed by the Mahalanobis Distancebased Over-sampling technique. The data will be tested on multiple hybrid datasets and classification accuracy using C4.5 algorithm will be evaluated. The accuracy results will be compared with the individual oversampling and under sampling approach.

Keywords: *Oversampling, under sampling, hybrid dataset, Mahalanobis distance, cluster based under sampling, Imbalance data, Classification*

Introduction

Lot of applications generates supervised data. i.e. data is divided in number of classes depending on its feature values. The number of records (instances) varies from class to class. If there is a large difference between numbers of instances for each class in a dataset, then such dataset is called as imbalanced dataset. The class contains less number of instances is called as minority class whereas class having greater number of instances as compared to other classes is called as majority class. If the dataset has two classes and there is large difference in number of instances in each class then one class is called as majority class and the other one is called as minority class. In multiclass dataset, more than one class can be labeled as minority class. Variety of applications generates such imbalanced data. The application includes disease diagnosis, activity recognition, fraud detection, protein fold etc. The performance of machine learning algorithm degrades in case of imbalanced data. The main aim of the study of imbalanced data handling is to improve the accuracy of machine learning algorithm for minority class without hampering the accuracy of majority class. The imbalanced data handling solution is classified in two main categories:

1. Data level:

The data level solution is the preprocessing task. This is applied before machine learning algorithms. Initially data is balanced and then it is given to the machine learning algorithm. The data level solution has two techniques for data balancing:

A: oversampling:

In oversampling the number of instances of minority class is increases by adding few dummy instances. The dummy instances are created by analyzing the existing instances in that class.

b: Undersampling:

In undersampling the majority class instance count is reduced with respect to the minority class instance count.

The oversampling technique may lead to overfitting and overgeneralization problem whereas undersampling may face problem of information loss and may mislead the classification technique.

2. Algorithmic level:

In the algorithmic level data balancing is virtually done in machine learning algorithms. In learning phase the balancing can be done using adjusting the threshold value or adjusting the probabilistic estimate. It can also be done using one class learning and ensemble learning. The binary class imbalanced problem can be handled using resampling or undersampling using data level solution or the classifier threshold can be shifted towards minority class using algorithmic level. But the same solution cannot be directly applied for multiclass imbalanced problem because: the relation among classes is not obvious and the classes boundaries may overlap. Following section elaborates the work done in the domain of data balancing. In the section 3, analysis of existing systems and problem formulation is elaborated. In section IV, new system is proposed followed by its implementation details and result analysis techniques. At the end conclusion is stated.

Related Work

The multiclass imbalanced problem is converted in to two class data imbalance problem by converting the problem as multiple two class sub problems. The widely used two class imbalance handling strategies are: one-versus-one(OVO) and one –versus-all(OVA). Z.-L. Zhang[4] proposes an ensemble learning based on OVA and OVO. Simply duplicating the instances in minority class may create overfitting problem whereas randomly deleing the samples of majority class may lose some important information. Chuanxia Jian, et. al. [5] proposes a new method called as different contribution sampling method (DCS). It works on contributions of the support vectors (SVs) and nonsupport vectors in classification technique. This technique uses biased support vector machine (B-SVM) method to find SVs and the NSVs from imbalanced data. SMOTE is widely used technique to balance the class distribution. It is similar to interpolation. In this technique initially K nearest neighbors are selected from minority class and then calculate the difference between them. The new samples are generated within the difference range. For this the difference value is multiplied by the random value between [0,1] and the generated value is added to the original minority class sample. SMOTE technique fails to preserve the class covariance structure. It increases the overlapping between various classes and disturbs the class boundaries[10]. Das et al. [6] proposes an oversampling technique based on joint probability distribution of data attributes. It uses Gibbs sampling for minority class sample generation process. Unlike SMOTE this strategy focuses on individual class properties and mutual relation among multiple classes. This increases the classification accuracy and the resultant dataset preserves the class covariance structure. Lin et al. [8] proposes a neural network based oversampling technique. A dynamic sampling procedure DyS technique uses train multilayer perceptrons (MLP). It iteratively selects instances from dataset based on the probability and generate final training set for MLP for multiclass imbalanced classification. Abdi and Hashemi [3] proposes a Mahalanobis distance based oversampling technique. This is a distance based oversampling technique. In this technique one class with highest samples is treated as majority class and all other classes are treated as minority class. The new samples are generated for each minority class equal to the number of instances in majority class. This is a good sampling technique for multiclass imbalanced problem with overlapped class structure. It preserves the class co-variance structure. This MDO technique is applicable for only numeric dataset.

Xuebing Yang, et. al[1] proposes an Adaptive Mahalanobis Distance-based Over-sampling AMDO technique. This is an extended version of MDO. This AMDO works on hybrid dataset. This technique balances the dataset based on Imbalanced Ratio(IR). For hybrid dataset learning it uses Heterogeneous Value Difference Metric distance (HVDM) and Generalized Singular Value Decomposition(GSVD) . Using HVDM it calculates the K2 nearest neighbor and using GSVD it transform the minority class samples to the PC space. The new samples are generated using Mahalanobis distance. The technique preserves the class covariance structure.

Unlike oversampling undersampling is also used to balance the dataset. Wei-Chao Lin, et.,al. [2] proposes a clustering based undersampling technique. In this technique majority class instances are clustered and form each cluster only a single representative instance is selected. The representative instance is preserved and all other instances in a cluster are deleted. The representative of cluster is selected using 2 strategies: 1 is the cluster centroid and 2: nearest neighbor of cluster center. The c4.5 classifier is used to evaluate the classification accuracy.

Analysis and Problem Formulation

There are various techniques for data balancing these techniques are mainly classified as: data level and algorithmic level solutions. The data level solutions are applied as a preprocessing step. It is mainly categorized in 2 sections: oversampling and undersampling. These techniques are studied independently. There is need to develop an ensemble approach system for data balancing which combines the oversampling and undersampling strategies.

Proposed Methodology

A. Architecture

The imbalanced dataset and metadata of data is input to the system. The imbalanced dataset contains numeric as well as nominal attributes. The system generates balanced dataset using ensemble learning approach. In ensemble learning system implements undersampling and oversampling techniques on the dataset and dataset is balanced. The accuracy of dataset is calculated using c4.5 classifier. Fig. 1 shows the architecture of the system..

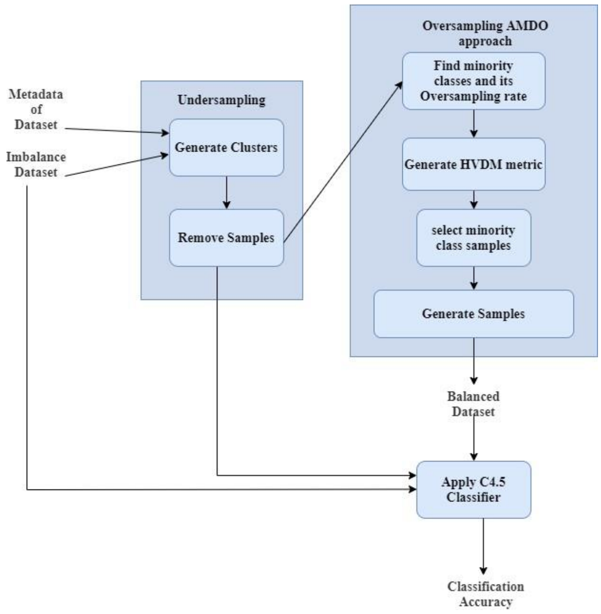

Figure 1 : System Architecture

A. *System Working*

Initially the metadata of dataset is extracted. The metadata information includes number of attributes, class attribute and data type of each attribute in the dataset. It also includes the class distribution. By analyzing the class distribution structure majority and minority classes are selected. Initially the maximum number of instances class is treated as majority class and all other classes are treated as minority class.

The imbalance ratio IR[9] ratio is calculated to check whether dataset is imbalanced or not. The IR ratio is calculated as:

$$IR = \frac{N_{maj}}{N_{min}} \qquad (1)$$

Where
Nmaj = number of instances in majority class
Nmin = number of instances in the smallest class

If the IR ratio of dataset is greater than 1.5 then the dataset is treated as imbalanced dataset. Ensemble learning approach is proposed and number of instances in dataset is deleted from majority class and number of samples are added to minority class till the IR ratio is less than or equal to the 1.5.

In ensemble approach initially undersampling is applied. In undersampling the number of instances from majority class is deleted using clustering approach. In clustering approach, the clusters of majority class instances are created using Kprototype algorithm. The cluster count is set to the number of instances in second highest class in a dataset.

The instances are reduced to the cluster count and the cluster centroids are selected as a representative instance of that cluster [9]. After reducing the samples of majority class the oversampling is applied on minority classes. The top classes having highest number of instances are majority classes and all other classes are treated as minority classes. The oversampling rate of each class is calculated using Partially Balanced Resampling algorithm. The dataset has numeric as well as nominal attributes. For further calculations the nominal attributes are converted to the numeric attributes. The Heterogeneous Value Difference MetricThe instances are reduced to the cluster count and the cluster centroids are selected as a representative instance of that cluster [9]. After reducing the samples of majority class the oversampling is applied on minority classes. The top classes having highest number of instances are majority classes and all other classes are treated as minority classes. The oversampling rate of each class is calculated using Partially Balanced Resampling algorithm. The dataset has numeric as well as nominal attributes. For further calculations the nominal attributes are converted to the numeric attributes. The Heterogeneous Value Difference Metric distance (HVDM) metric [10] is used for all nominal attributes in a dataset. The HVDM metric is calculated as:

$$HVDM\ (x,y) = \sqrt{\sum_{a=1}^{m} d_a{}^2(x_a,y_a)} \qquad (2)$$

m = number of attributes da(x, y) = distance between two values of attribute a from 2. vectors x and y. This can be calculated as: 3. da(x, y) = 1 if x and y is unknown 4. da(x, y) = normalized-vdma(x,y) if a is nominal 5. da(x, y) = normalized-diffa(x,y) if a is linear 6. The normalized VDM distance (normalized-vdma) is is calculated as:

$$= \sum_{c=1}^{C} \left| \frac{N_{a,x,c}}{N_{a,x}} - \frac{N_{a,y,c}}{N_{a,y}} \right|^q \qquad (3)$$

Where,
$N_{a,x}$ = Number of instances having x value for attribute a
$N_{a,x,c}$ = Number of instances having x value for attribute a and class c
C = Total number of classes
q = Constant value (this value is set to 1 or 2)
Normalized Difference Distance(normalized-diffa) is calculated as:

$$\text{Normalized} - \text{diff}_a(x,y) = \frac{|x-y|}{4\sigma_a} \qquad (4)$$

Where, σ is the standard deviation value of attribute a.

For minority classes new instances are generated based on the Mahalanobis distance. Using Mahalanobis distance the new instance is created within the eclipse contours. he overlapping entries of instances among multiple classes. After finding HVDM metric, the K2 nearest neighbor of each instance in minority class is calculated. Based on the highest nearest neighbors count, the representative instances are selected from minority class. These instances are further used in sample generation process. The selected instances are then transformed to PC space using Generalized Singular Value Decomposition. GSVD[11]. GSVD transforms the mixed type data in PC space. This Mahalanobis distance creates new samples within the eclipse contours. This preserves the class co-variance structure and reduces the class overlapping. The new samples are generated by solving the following equation.

$$\frac{x1^2}{\alpha V1}+\frac{x2^2}{\alpha V2}+\cdots+\frac{xd^2}{\alpha Vd}=1 \qquad (5)$$

Where,
V is the vector coefficient. It is extracted from covariance matrix generated with the help of GSVD. α is constant value. And x1,s2,..xd are instance values for ddimensional dataset.

The generated new samples are then transformed to the original space and new samples are added to minority class. The IR ratio of dataset is calculated again and Resampling process is executed again if IR ratio is greater than 1.5. The AMDO: Adaptive Mahalanobis Distance-based Over-sampling algorithm describes the detailed process of Oversampling. The C4.5 classifier is used for classification accuracy evaluation. C. Algorithms Algorithm 1: AMDO: Adaptive Mahalanobis Distance-based Over-sampling Input: S: Imbalanced dataset, A: Metadata of all attributes

K1, K2: System constants Output: S': updated dataset Processing:

```
1. Initialize: c: set of classes , p1: set of numeric attributes , p2: set of nominal attributes, m: distinct values of nominal attribute, D: Class distribution , nmaj : number of samples in majority class
2. Calculate Orate for all minority Cn-1 classes using algorithm2
3. For x =1 to cn-1
4. Sx = Read samples of class x
5. For each sample i in Sx
6. Find K2 nearest neighbors using HVDM matrix
7. Num(i) = number of nearest neighbors of selected sample i
8. Weight(i) = num(i)/ K2
9. If Num(i)<K1
10. Remove i from Sx
11. End
12. End
13. Obtains Xsup = transform nominal attributes to m distinct attributes
14. Calculate $\mu_s$:Mean and σs : standard deviation of Xsup
15. $Xsup = \frac{Xsup-\mu_s}{\sigma s}$ , Normalize sample set
16. Generate matrix N and X from Xsup
16. Update Xsup = N1/2XsupM1/2
17. Compute V via GSVD of Xsup using N and M
18. Obtain diagonal vector of coefficients of matrix V
19. If Orate if class >0 then
20. For i=0 to Orate
21. Choose random sample from Xsup
22. Compute α using Mahalanobis distance function
23. Generate p11 + m positive random numbers
24. r1,..rp11+m and make them sum to one
25. for k= 1 : p11 + m
26. calculate Xk $=\frac{Xk^2}{\alpha Vk}$
27. end
28. Add Xk to Xsnew
29. End
30. End
31. Xsnew = σs(N-1/2XsnewVTM(-1/2) + μ)
32. End
33. Update S' = S+Xsnew
34. Return S'
```

Algorithm 2: Partially Balanced Resampling
Input: c: set of classes
D : distribution of class Output: Orate of cn-1 classes
Processing:

```
1. Initialize N: number of attributes
2. Initialize nmin = number of samples in minority class, ncmin = number of samples in smallest minority class, nmax = number of samples in majority class
3. Calculate Maxit = $\frac{(n_{max}(cn-1))}{n_{min}}$
4. Initialize D' = D, N' = N
5. For i = 1 to maxit
6. Ntemp = current size of nmin
7. Find minimum rato Pmin= ntemp/ N'
8. If P$^{min}$ <=$\frac{2}{3c-1}$
9. Ntemp = Ntemp+ncmin
10. end
11. update D' and N'
12. end
13. calculate Orate = D' - D
14. return Orate for all minority classes
```

D. Mathematical Model

The system S can be defined as:
S= {I, O, P} where
I = {I1, I2}, Set of Inputs
I1 = imbalanced dataset
I2 = feature count
O = {O1,O2 }, Set of Outputs
O2= Balanced Dataset
O3 = Accuracy using C4.5
F= {F1, F2, F3, F4, F5, F6, F7, F8, F9, F10, F11, F12, F13, F14,
F15}, Set of Functions
F1 = Upload dataset
F2 = create clusters using k means
F3= Remove samples
F4 = Find Orate of minority class
F4 = Generate HVDM as the metric
F5 =Choose minority class samples
F6 = Find nearest neighbor

F7 = Select nearest neighbor
F8 = Select Neighbors
F9= Normalize sample set
F10= PC space transformation using GSVD
F11 = Find vector coefficient
F12 = generate sample using MDO-oversampling
F13 = Transform samples to Original space
F14 = Generate ensemble method for data balancing
F15 = Apply C4.5 classifier

Result and Analysis

The system is implemented in java using jdk 1.8 using Netbeans 8.2 IDE. For implementation and testing the windows 10 system is used with 4gb ram with i5 processor.

A. *Dataset:*

UCI[12] and KEEL[13] benchmark data sets datasets are used for system testing. Following table 1 represents the detailed description of dataset.

Table 1 : Dataset Description

Sr. No .	Dataset	No. of Classes	No. of instanc es	No. of attribut es	Class Distribution
1.	Balance	3	625	4	288/49/288
2.	Dermat ology	6	358	34	111/60/71/4 8/48/20
3.	Thyroid	3	7,200	21	166/368/6,6 66
4.	Contrac eptive	3	1,473	9	629/333/511

B. *Performance Measures*

1. F-measure:

Using C4.5 classifier the accuracy of sample generation process is evaluated. The accuracy of original dataset and reduced dataset is compared. The f-measure is calculated as: F-measure = $\frac{precison*recall}{precision+recall}$ (6) Where precision is calculated as:

Precision = $\frac{Actual\ Instances\ \cap Correctly\ classified\ Instances}{correctly\ classified\ Instances}$ (7)

And recall is calculated as:

Recall = $\frac{Actual\ Instances\ \cap classified\ Instances}{Actual\ Instances}$ (8)

2. Evaluation Time:

Execution Time for data balancing process is evaluated.

C. Implementation Status:

The system is partially implemented. Oversampling rate is determined using Partially Balanced Resampling algorithm.. Following table shows the results on various dataset with respect to oversampling rate for each class and time required for processing.

Table 1 : Oversampling rate Evaluation

Sr. No.	Dataset	Oversampling rate	Time for Processing (In milliseconds)
1.	Balance	0/147/0	46
2.	dermatology	0/0/0/0/0/40	77
3.	Thyroid	323/ 306/ 0	136
4.	Contraceptive	0/333/0	138

Conclusions

In this research work, the system generates normalized dataset using ensemble learning approach. In ensemble learning cluster based undersampling and Mahalanobis Distance based oversampling technique are used collectively to balance a dataset. The system also works on hybrid dataset. After generating the balanced dataset the classification accuracy of dataset is evaluated using C4.5 classification algorithm. In future system can be implemented with different techniques such as hellinger distance based oversampling, Knn based undersampling, etc.

References

[1]. Xuebing Yang , Qiuming Kuang , Wensheng Zhang, and Guoping Zhang, "AMDO: An Over-Sampling Technique for Multi-Class Imbalanced Problems", IEEE Trans. Knowl. Data Eng., vol.30, no. 9, pp. 1672 - 1685 Sept. 2018

[2]. Lin, Wei-Chao & Tsai, Chih-Fong & Hu, Ya-Han & Jhang, Jing-Shang," Clustering-based undersampling in class-imbalanced data", in Information Sciences, vol. 409, May 2017.

[3]. L. Abdi and S. Hashemi, "To combat multi-class imbalanced problems by means of over-sampling techniques," IEEE Trans. Knowl. Data Eng., vol. 28, no. 1, pp. 238-251, Jan. 2016.

[4]. Z. L. Zhang, B. Krawczyk, S. Garcia, A. Rosales-Perez, and F. Herrera, "Empowering ono-vs-one decomposition with ensemble learning for multiclass imbalanced data," Knowl.-Based Syst., vol. 106, no. C, pp. 251–263, Aug. 2016.

[5]. C. X. Jian, J. Gao, and Y.-H. Ao, "A new sampling method for classifying imbalanced data based on support vector machine ensemble,"

[6]. Nurocomputing, vol. 193, no. C, pp. 115–122, Jun. 2016. [6] B. Das, N. C. Krishnan, and D. J. Cook, "RACOG and wRACOG: Two probabilistic oversampling techniques," IEEE Trans. Knowl. Data Eng., vol. 27, no. 1, pp. 222–234, Jan. 2015.

[7]. M. Chavent, V. Kuentz-Simonet, A. Labenne, and J. Saracco, "Multivariate analysis of mixed type data:The PCAmixdata R package", Nov. 2014.

[8]. M.-L. Lin, K. Tang, and X. Yao, "Dynamic sampling approach to training neural networks for multiclass imbalance classification," IEEE Trans. Neural Netw. Learn. Syst., vol. 24, no. 4, pp. 647–660, Apr. 2013.

[9]. A. Orriols-Puig and E. Bernado-Mansilla, "Evolutionary rulebased systems for imbalanced data sets," Soft. Comput., vol. 13, no. 3, pp. 213–225, Oct. 2008.

[10]. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," J. Artif. Intell. Res., vol 16, no. 1, pp. 321–357, Jan. 2002.

[11]. D. R.Wilson and T. R.Martinez, "Improved heterogeneous distance functions," J.Artif. Intell. Res., vol. 6, no. 1, pp. 1–34, Jan. 1997. UCI dataset: http://mlr.cs.umass.edu/ml/datasets.html