International Journal of Computational Intelligence Research
ISSN 0973-1873 Volume 15, Number 1 (2019), pp. 11-32

http://www.ripublication.com

A Novel Algorithm for Class Imbalance Learning on Big Data Using Under Sampling Technique

Dr. Mohammad Imran[1]
Assistant Professor in Department of CSE,
Muffakham Jah College of Engineering and Technology,
Banjara Hills, Hyderabad-500034, India.

Ms. Shama Kouser[2]
Lecturer in Department of Computer Science,
Jazan University, Jazan, Kingdom of Saudi Arabia.

Mr. Maradana Durga Venkata Prasad[3]
Research Scholar (Regd No:1260316406),
Department of Computer Science and Engineering,
GITAM Deemed to be University, Visakhapatnam,
Andhra Pradesh, India.

Abstract

Classifiers are trained with datasets of imbalanced class distributions, imbalance big data is an important problem in data mining. Imbalance in the data occurs when the number of examples representing the class of interest is much lower than the ones of the other classes. The presence of Imbalance datasets in many real-world applications has brought along a growth of attention from researchers.

By introducing the characteristics of the imbalanced dataset scenario in classification, presenting the specific metrics for evaluating performance in class imbalanced learning and enumerating the proposed solutions. In particular, we will describe preprocessing, and ensemble techniques, carrying out an experimental study to contrast these approaches.

In this paper we propose an under sampling algorithm for big data and will carry out a detailed discussion on the main issues related to using data intrinsic characteristics in this classification problem. Finally, we introduce several approaches and recommendations to address these problems in conjunction with imbalanced data, and we will show some experimental examples on the behavior of the learning algorithms on data with such intrinsic characteristics.

Keywords: Imbalanced big dataset, sampling, noisy data, preprocessing ensemble techniques, Under sampling, cost-sensitive learning.

1. INTRODUCTION

In many supervised learning applications, there is a significant difference between the probabilities with which an example belongs to the different classes of the classification problem. This situation is known as the class imbalance problem [1] [2] [3]. Class imbalance problem is common in many real problems from telecommunications, World Wide Web, financial and accounting, ecology, biology, medicine. Furthermore, it is worth to point out that the minority class is usually the one that has the highest interest from a learning point of view and it also implies a great cost when it is not well classified[4].

The drawback with imbalanced datasets is that standard classification algorithms are often biased towards the majority class (known as the ‘‘negative’’ class) and therefore there is a higher misclassification rate for the minority class instances (known as the ‘‘positive’’ class), many solutions such as bagging, boosting and hybrid based approaches have been proposed to deal with this problem, both for standard learning algorithms and for ensemble techniques[5].

The methods of dealing with class imbalance problem can be categorized into three major groups:

1) Data Sampling: In which the training instances are modified in such a way to produce a more or less balanced class distribution that allow classifiers to perform in a similar manner to standard classification [6,7].
2) Algorithmic Modification: This procedure is oriented towards the adaptation of base learning methods to be more assimilate to class imbalance issues [8].
3) Cost-sensitive learning: This type of solutions incorporate approaches at the datalevel, at the algorithmic level, or at both levels combined, considering higher costs for the misclassification of examples of the positive class with respect to the negative class, and therefore, trying to minimize higher cost errors.[9,10]

Most of the studies on the behavior of several standard classifiers in imbalance domains have shown that significant loss of performance is mainly due to the skewed class distribution, given by the imbalance ratio (IR), defined as the ratio of the number of instances in the majority class to the number of examples in the minority class [11,12].

$$\mathrm{IR} = \frac{|\mathrm{support(A)} - \mathrm{support(B)}|}{\mathrm{support(A)} + \mathrm{support(B)} - \mathrm{support}(\mathrm{A} \cup \mathrm{B})}$$

In much simpler representation IR can also be given as

$$\mathrm{IR} = \frac{\{\text{number of instances in the majority class}\}}{\{\text{number of instances in the minority class}\}}$$

While some people might consider these both uninteresting, others might want to know about this. To differentiate between the two situations, we can look at Imbalance Ratio where 0 is perfectly balanced and 1 is very skewed. There are several investigations which also suggest that there are other factors that contribute to such performance degradation [13].

This paper is organized as follows. First, Section 2 presents the problem of imbalanced datasets, introducing its features and the metrics employed in this context. Section 3 describes the diverse preprocessing, cost-sensitive learning and ensemble methodologies that have been proposed to deal with this problem. Next, we develop an experimental study for contrasting the behavior of these approaches in Section 4. Section 5 is devoted to analyzing and discussing the aforementioned problems associated with data intrinsic characteristics. Finally, Section 6 summarizes and concludes the work.

2. IMBALANCED DATASETS IN CLASSIFICATION

In this section, we first introduce the problem of imbalanced datasets and then we present the evaluation metrics for this type of classification problem, which differ from usual measures in classification.

2.1 The problem of imbalanced datasets

In the classification problem the scenario of imbalanced datasets appears frequently. The main property of this type of classification problem is that the examples of one class significantly outnumber the examples of the other one [14,15]. In most cases, the imbalanced class problem is associated to binary classification, but the multi-class problem often occurs and since there can be several minority classes, it is more difficult to solve [16,17].

Since most of the standard learning algorithms consider a balanced training set, this may generate suboptimal classification models, i.e. a good coverage of the majority examples, whereas the minority ones are misclassified frequently. Therefore, those algorithms, which obtain a good behavior in the framework of standard classification, do not necessarily achieve the best performance for imbalanced datasets [18].The imbalanced learning problem has received much attention from the machine learning community. regarding real world domains, the importance of the imbalance learning problem is growing, since it is a recurring issue in many applications. As some examples, we could mention very high resolution airbourne imagery [19], forecasting of ozone levels [20], face recognition[21], and especially medical diagnosis [22].

2.2 Evaluation in imbalanced domains

The evaluation criteria is a key factor in assessing the classification performance and guiding the classifier modeling. In a two-class problem, the confusion matrix (shown in Table 1) records the results of correctly and incorrectly recognized examples of each class.

Table 1: Confusion matrix for a two-class problem.

	Positive prediction	Negative prediction
Positive class	True Positive (TP)	False Negative (FN)
Negative class	False Positive (FP)	True Negative (TN)

Traditionally, the accuracy rate (Eq.(1),Eq.(2)) has been the most commonly used empirical measure. However, in the frame work of imbalanced datasets, accuracy is no longer a proper measure, since it does not distinguish between the number of correctly classified examples of different classes. Hence, it may lead to erroneous conclusions, i.e., a classifier achieving an accuracy of 90% in a dataset with an IR value of 9 is not accurate if it classifies all examples as negatives

The Area under Curve (AUC) measure is computed by,

$$AUC = \frac{1 + TP_{RATE} - FP_{RATE}}{2} \qquad (1)$$

[Or]

$$AUC = \frac{TP + TN}{TP + FN + FP + TN} \;\ldots\ldots\ldots\ldots\ldots(2)$$

On the other hand, in several problems we are especially interested in obtaining high performance on only one class. For example, in the diagnosis of a rare disease, one of the most important things is to know how reliable a positive diagnosis is. For such problems, the precision (or purity) metric is often adopted, which can be defined as the percentage of examples that are correctly labeled as positive: The Precision measure is computed by,

$$\text{Precision} = \frac{TP}{(TP)+(FP)} \qquad (3)$$

In statistics, the F-measure is a measure of a test's accuracy. It considers both the precision p and the recall r of the test to compute the score: p is the number of correct results divided by the number of all returned results and r is the number of correct results divided by the number of results that should have been returned. The F-measure can be interpreted as a weighted average of the precision and recall, where an F-measure reaches its best value at 1 and worst score at 0. The F-measure is the harmonic mean of precision and recall: F-measure is a measure that combines precision and recall is the harmonic mean of precision and recall, the traditional F-measure or balanced F-score. The F-measure Value is computed by,

$$F - measure = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \qquad (4)$$

To deal with class imbalance, sensitivity (or recall) and specificity have usually been adopted to monitor the classification performance on each class separately. Note that sensitivity (also called true positive rate, TP rate) is the percentage of positive examples that are correctly classified, while specificity (also referred to as true negative rate, TN rate) is defined as the proportion of negative examples that are correctly classified:

The True Positive Rate measure is computed by,

$$TruePositiveRate = \frac{TP}{(TP)+(FN)} \qquad (5)$$

The True Negative Rate measure is computed by,

$$TrueNegativeRate = \frac{TN}{(TN)+(FP)} \qquad (6)$$

2.3. *Addressing Classification with Imbalanced Data: preprocessing, cost-sensitive learning and ensemble techniques*

A large number of approaches have been proposed to deal with the class imbalance problem. These approaches can be categorized into two groups: the internal approaches that create new algorithms or modify existing ones to take the class-imbalance problem into consideration [23] and external approaches that preprocess the data in order to diminish the effect of their class imbalance [24]. Furthermore, cost-sensitive learning solutions incorporating both the data (external) and algorithmic level (internal) approaches assume higher misclassification costs for samples in the minority class and seek to minimize the high cost errors [25].Ensemble methods [26,27] are also frequently adapted to imbalanced domains, either by modifying the ensemble learning algorithm at the data-level approach to preprocess the data before the learning stage of each classifier [28,29] or by embedding a cost-sensitive framework in the ensemble learning process [30]. A complete taxonomy for ensemble methods for learning with imbalanced classes can be found on a recent review [31], which we summarize in Fig. 2

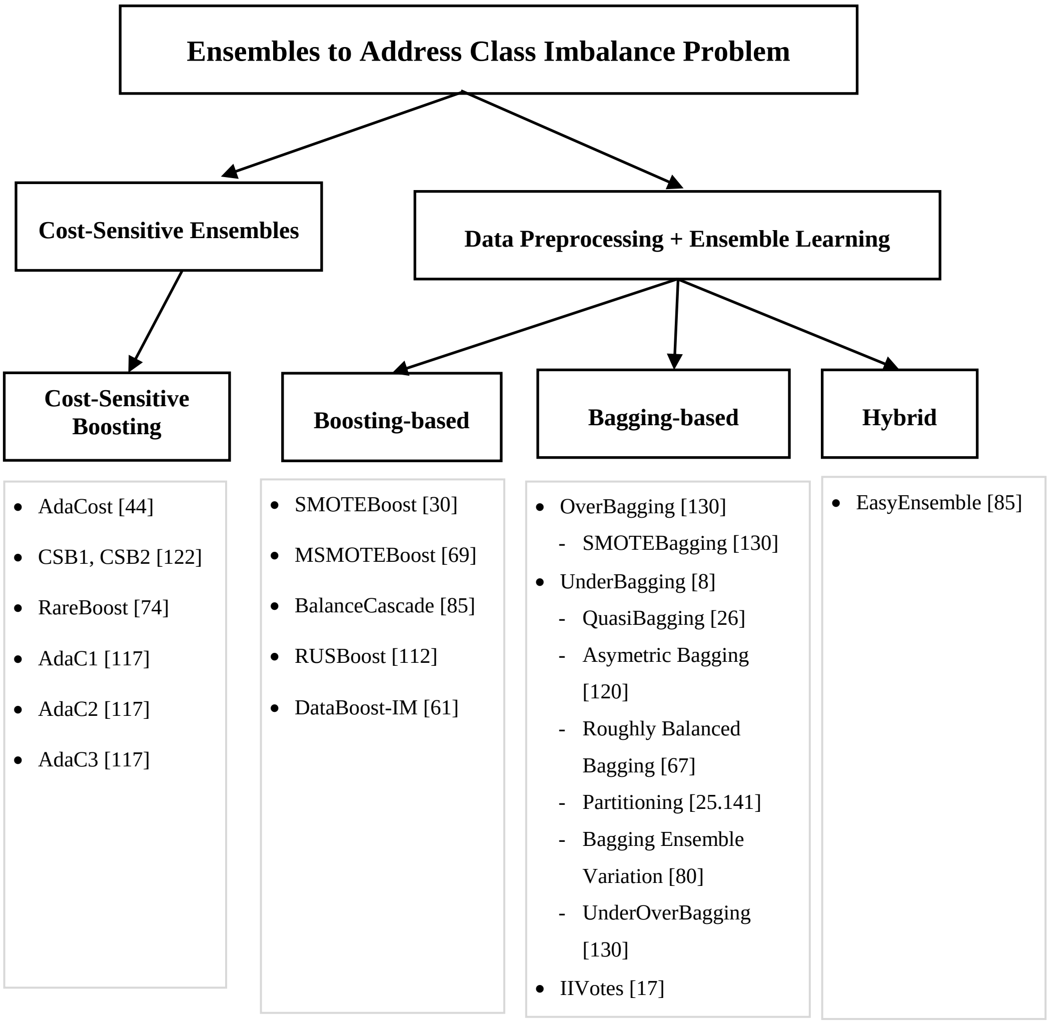

Fig. 2. Galar et al.'s proposed taxonomy for ensembles to address class imbalance problem. (See above-mentioned references for further information).

3. FRAMEWORK OF USIBD ALGORITHM

The proposed USIBD algorithm is summarized as below.

Algorithm: Under Sampled Imbalance Big Data(USIBD)

```
Algorithm: New Predictive Model
Input: D – Data Partition,
A – Attribute List
Output: A Decision Tree
Procedure:

Processing Phase:
Step 1. Take the class imbalance data and divide it into majority and minority sub sets. Let the minority subset be P € pi (i = 1,2,..., pnum) and majority subset be N € ni(i = 1,2,..., nnum).

Selection Phase
Step 1: begin
Step 2: k ← 0,j←1.
Step 3: Apply CFS on subset N,
Step 4: Find Fj from N, k= number of features extracted in CFS
Step 5: repeat
Step 6: k=k+1
Step 7: Select the range for weak or noises instances of Fj.
Step 8: Remove ranges of weak attributes and form a set of major class examples Nstrong
Step 9: Until j = k
Step 10: Form a new dataset using P and Nstrong
Step 11:End

Building Predictive Model:
1. Create a node N
2. If samples in N are of same class, C then
3. return N as a leaf node and mark class C;
4. If A is empty then
5. return N as a leaf node and mark with majority class;
6. else
7. apply C4.5
8. endif
9. endif
10. Return N
```

The algorithm Under Sampled Imbalance Big Data (USIBD) learning is a unique framework, which performs under sampling by following a strategic approach of removing the instances from the majority subset. Under sampling can help improve run time and storage problems by reducing the number of training data samples when the training data set is huge.

These limitations are uniquely addressed in our proposal such as: under sampling can discard potentially useful information which could be important for building rule classifiers. The sample chosen by random under sampling may be a biased sample. It will not be an accurate representative of the population and thereby, resulting in inaccurate results with the actual test data set.

In different scenarios, an aim of under sampling is to balance class distributions. The process of eliminating majority instances depending upon unique properties of the datasets can be extended for different percentages.

Our proposed method consists of two steps. In the first step, we construct an influence space around a test point *p*. In the second step a rank difference based outlier score is assigned on the basis of this influence space.

3.1. Influence space construction

Influence space depicts a region with significantly high reverse density in the locality of a point under consideration.

If the localities of the neighbours within the influence space are denser with respect to the locality of the concerned point, then a high value of outliernesss core will be assigned to it. For an entire dataset, number of neighbours in the influence space is kept fixed. As the distance is increased from the target point, more number of neighbours gets included in its surroundings result

In given different values of radius R, with successive addition of neighboring points, a set of reverse densities is obtained for each point at varying depths (number of neighboring points). The average reverse density *R* for each depth is determined next. Note that we have considered the depth and not the distance around the neighbours to handle situations where there is empty space (no neighboring point is present) surrounding a given point. To avoid random fluctuations, the variation in the average reverse density with respect to depth has been smoothed using a Gaussian kernel.

In this smoothing process, an optimal width for the kernel optimal is determined using better estimation of the significant density fluctuation around the neighbor points. We deem the first most significant peak in this smoothed kernel probability density function as the limit of the influence space. The peak has been determined using the un decimated value.

3.2. Outlier score

In the second part of our proposed algorithm we have used a rank difference based score for ranking of the outliers. The positive value of the rank difference ($R-k$) signifies the high concentration of the neighbours around the training point q than that of the test point p. The negative and zero value respectively signify a lower or same concentration of the training points around q than that of p. Thus the outlierness of the test point depends directly on the excess population of the neighbourhood space of q with respect to the test point p, i.e., on the rank difference ($R-k$). Secondly, it also depends inversely on its own forward density.

4. EXPERIMENTAL SETUP AND RESULTS

In order to compare the performance of the different proposed methods in this research, 15 standard imbalanced datasets from the UCI Machine Learning Repository [31] are used in the experiments. According to the scope of the work, all the datasets represent two-class domains are considered. Thus, the collections of datasets provide a good combination of real world sampling of class imbalance problems with wide range and are used by several researchers and academicians.

The classifier's future performance can't be simply measures by accuracy on the training dataset. If so, 100 % accuracy can be obtained in most of the cases. In real time application, one of the ways of measuring the performance is to divide the dataset into two subsets. One subset is used for training of classifier and other is used for validation. This approach is called is called as hold-out method. In hold-out approach a well noted problem exits, that is the performance of hold-out method largely depends upon the division of original dataset. This problem is called as variance.

The solution to the above problem is to design by an approach known as Cross-Validation. The cross validation technique is a standard technique for generating reliable and accurate results and it has been used by many researchers and academicians in machine learning.

We used tenfold cross validation (CV) in all our experiments to estimate AUC, Precision, F-measure, TP Rate and TN Rate. A k fold CV experiment consists of the following steps.

1. Randomly dividing the data into k equal sized disjoint partitions.
2. For each partition, build a decision tree using all data outside the partition, and test the tree on the data in the partition.
3. Sum the number of correct classifications of the k trees and divide by the total number of instances to compute the classification accuracy. Report this accuracy and the average size of the k trees.

Usually the number of folds in the cross-validation is set to ten. This number has been found empirically to be a good choice and this idea is supported using a theoretical result by many prominent researchers.

This work uses ten times stratified ten-fold cross-validation. When randomizing the original data each time, the seed is set differently before the data is divided into ten parts. Thus, each time a model is built on different data and classification is also based on different data. This reduces variance further.

The framework for 10 fold cross validation is shown in Figure 1. Each entry in all the experiments is the results of ten 10 fold CV experiments: i.e., the result of tests that used 100 models. Each of the ten 10 fold cross validations used a different random partitioning of the data. Each entry in the tables reports the average AUC, Precision, F-measure, TP Rate and TN Rate. Good results should have high values for AUC, Precision, F-measure, TP Rate and TN Rate.

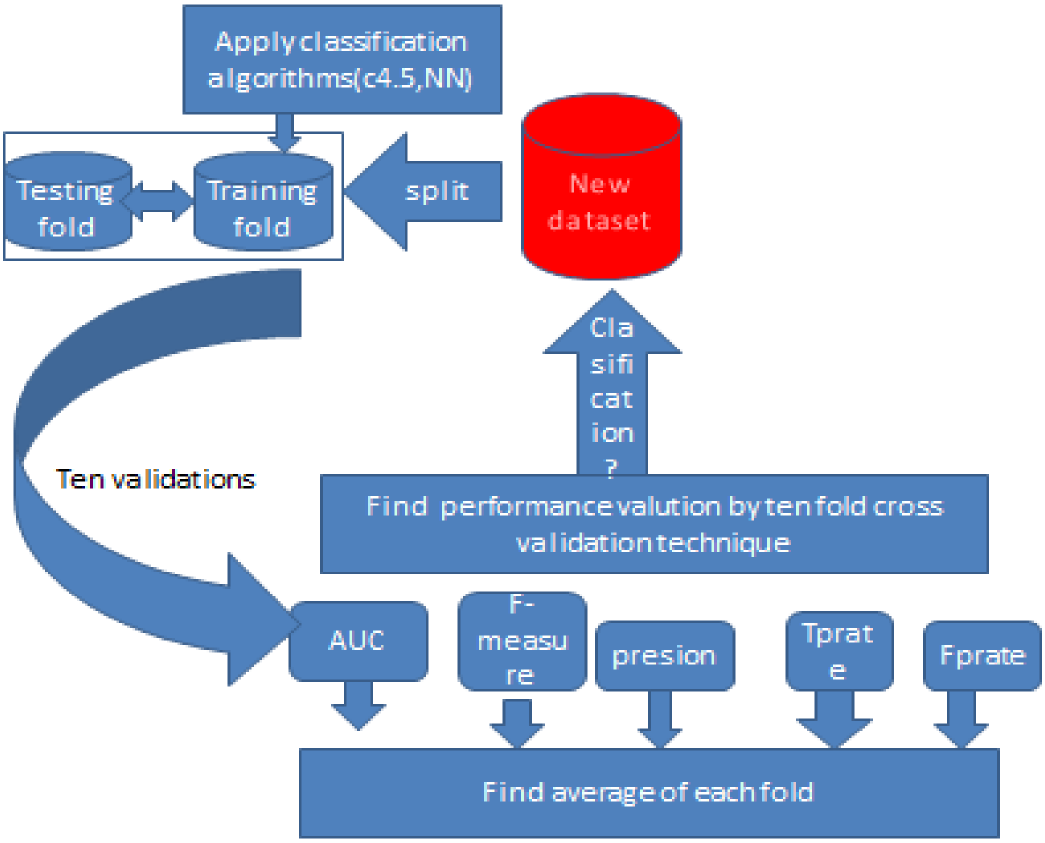
Apply classification algorithms(c4.5,NN)
Testing fold
Training fold
split
New dataset
Clasification?
Ten validations
Find performance valution by ten fold cross validation technique
AUC
F-measure
presion
Tprate
Fprate
Find average of each fold

Figure 1: Frame work for 10 Fold Cross Validation

A two-tailed corrected resampled paired t-test is used in this thesis to determine whether the results of the cross-validation show that there is a difference between the two algorithms is significant or not. Difference in accuracy is considered significant when the p-value is less than 0.05 (confidence level is greater than 95%). In discussion of results, if one algorithm is stated to be better or worse than another then it is significantly better or worse at the 0.05 level.

We performed the implementation of our new algorithms within the Weka[32] environment on windows XP with 2Duo CPU running on 2.53 GHz PC with 2.0 GB of RAM. Weka is a widely used data mining toolkit used in machine learning and it has been put into practice by many researchers and academicians.

Datasets:

The proposed methods are experimented using twelve benchmark real-world imbalanced dataset from the UCI machine learning repository. Table 1 summarizes the data selected in this study and shows, for each data set, the number of examples (#Ex.), number of attributes (#Atts.), class name of each class (minority and majority) and IR of the dataset for all the 15 UCI dataset. We downloaded these data sets in format of ARFF (Attribute-Relation File Format) from main web site of Weka. From the Table 1 it is clearly evident that all the 20 UCI datasets have IR value which indicates that these datasets are of highly imbalanced.

Table 2 The UCI datasets and their properties

S.No.	Dataset	Inst.	Missing values	Numeric. attributes	Nominal attributes	Classes	IR
1.	Anneal	898	no	6	32	5	6.90
2.	Anneal.ORIG	898	yes	6	32	5	6.90
3.	Arrhythmia	452	yes	206	73	13	1.56
4.	Audiology	226	yes	0	69	24	2.85
5.	Autos	205	yes	15	10	6	2.09
6.	Balance-scale	625	no	4	0	3	5.87
7.	Breast-cancer	286	yes	0	9	2	2.36
8.	Breast-w	699	yes	9	0	2	2.36

S.No.	Dataset	Inst.	Missing values	Numeric. attributes	Nominal attributes	Classes	IR
10.	Car	1728	yes	7	18	61	1.56
11.	Colic-h	368	yes	7	15	2	1.70
12.	Colic-h.ORIG	368	yes	7	15	2	1.96
13.	Credit-a	690	yes	6	9	2	1.24
14.	Credit-g	1000	no	7	13	2	2.33
15.	Pima diabetes	768	no	8	0	2	1.86
16.	Ecoli	336	no	7	0	8	1.70
17.	Glass	214	no	9	0	6	2.62
18.	Heart-c	303	yes	6	7	2	1.77
19.	Heart-h	294	yes	6	7	2	1.56
20.	Heart-statlog	270	no	13	0	2	1.25
21.	Hepatitis	155	yes	6	13	12	3.84
22.	Hypothyroid	3772	yes	7	22	4	17.94
25.	Ionosphere	351	no	34	0	2	17.65
26.	Iris	150	no	4	0	3	1.00

S.No.	Dataset	Inst.	Missing values	Numeric. attributes	Nominal attributes	Classes	IR
27.	Kr-vs-kp	3196	no	0	36	2	1.09
28.	Labor	57	yes	8	8	2	1.85
29.	Letter	20000	no	16	0	26	1.02
30.	Lympho	148	no	3	15	4	2.03
31.	Mfeat	2000	no	217	9	0	1.31
32.	Mushroom	8124	yes	0	22	2	1.43
33.	Nursery	12960	no	9	13	17	1.03
34.	Optdigits	5620	no	64	0	10	1.45
35	Page-blocks	5473	no	11	0	2	14.93
36.	Pendigits	10992	no	16	0	10	1.82
37.	Primary-tumor	339	yes	0	17	21	1.52
38.	Segment	2310	no	19	0	7	1.56
39.	Sick	3772	yes	7	22	2	1.73
40.	Sonar	208	no	60	0	2	1.14
41.	Soybean	683	yes	0	35	19	1.11

S.No.	Dataset	Inst.	Missing values	Numeric. attributes	Nominal attributes	Classes	IR
42.	Splice	3190	no	0	61	3	2.15
43.	Vehicle	846	no	18	0	4	1.32
44.	Vote	435	yes	0	16	2	1.45
45.	Vowel	990	no	10	3	11	1.00
46.	Waveform	5000	no	41	0	3	1.05
47.	Zoo	101	no	1	16	7	1.32

In this present work, we used various popular and effective criteria for validating proposed algorithms. It is now well known that error rate is not an appropriate evaluation criterion when there is class imbalance or unequal costs. To assess the classification results we count the number of true positive (TP), true negative (TN), false positive (FP) (actually negative, but classified as positive) and false negative (FN) (actually positive, but classified as negative) examples. There are many complex and appropriate metrics which are used in practical domain for evaluation of imbalance datasets.

In this work, we use AUC, Precision, F-measure, TP Rate and TN Rate as performance evaluation measures. Let us define a few well known and widely used measures:

Receiver Operating Characteristic (ROC) curve is the recent evaluation metric used for classifiers dealing with imbalanced data study. This ROC curve can be used for projecting results depending upon the user perspective with different combinations of basic components such as true positives, false positives, true negatives and false negatives. The summary of the ROC curve can be given as the area under it, which is known as Area Under Curve (AUC). AUC can be computed simple as the micro average of TP rate and TN rate when only single run is available from the classifier.

5. UNDER SAMPLED IMBALANCE BIG DATA APPROACH

Table 5.1 Summary of tenfold cross validation performance for AUC on all the datasets

Datasets	**C4.5**	**USIBD**
anneal	**0.931±0.164●**	**0.938±0.166**
car	**0.981±0.011○**	**0.919±0.080**
cmc	**0.691±0.049●**	**0.692±0.048**
kr-vs-kp	**0.998±0.003○**	**0.998±0.002**
letter	**0.985±0.011○**	**0.983±0.012**
mfeat	**0.967±0.036●**	**0.969±0.030**
mushroom	**1.000±0.000**	**1.000±0.000**
nursery	**1.000±0.000**	**1.000±0.000**

○ Empty dot indicates the loss of USIBD.

● Bold dot indicates the win of USIBD;

Table 5.2 Summary of tenfold cross validation performance for Precision on all the datasets

Datasets	**C4.5**	**USIBD**
anneal	**0.505±0.500●**	**0.660±0.454**
car	**0.972±0.016○**	**0.923±0.131**
cmc	**0.606±0.051●**	**0.613±0.048**
kr-vs-kp	**0.994±0.006●**	**0.995±0.006**
letter	**0.952±0.028●**	**0.953±0.022**
mfeat	**0.921±0.077●**	**0.935±0.065**
mushroom	**1.000±0.000**	**1.000±0.000**
nursery	**1.000±0.000○**	**0.400±0.492**

○ Empty dot indicates the loss of USIBD.

● Bold dot indicates the win of USIBD;

Table 5.3 Summary of tenfold cross validation performance for Recall on all the datasets

Datasets	**C4.5**	**USIBD**
anneal	**0.510±0.502●**	**0.700±0.461**
car	**0.962±0.018○**	**0.771±0.176**
cmc	**0.617±0.063○**	**0.614±0.068**
kr-vs-kp	**0.995±0.005○**	**0.994±0.007**
letter	**0.965±0.023○**	**0.961±0.024**
mfeat	**0.925±0.080●**	**0.938±0.062**
mushroom	**1.000±0.000**	**1.000±0.000**
nursery	**1.000±0.000○**	**0.400±0.492**

○ Empty dot indicates the loss of USIBD.

● Bold dot indicates the win of USIBD;

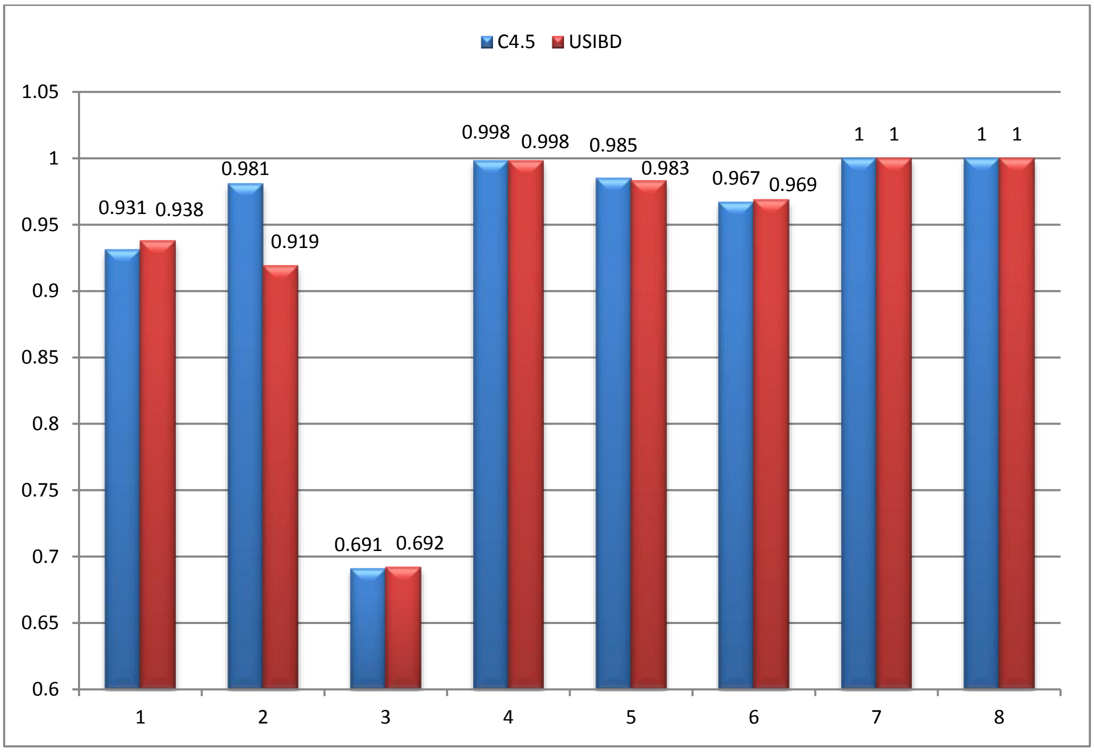

Fig. 5.1 Trends of USIBD v/s C4.5 on imbalance Big dataset

Table 5.4 Summary of tenfold cross validation performance for F-measure on all the datasets

Datasets	**C4.5**	**USIBD**
anneal	**0.507±0.500 ●**	**0.673±0.452**
car	**0.967±0.011○**	**0.827±0.135**
cmc	**0.610±0.049 ●**	**0.612±0.048**
kr-vs-kp	**0.995±0.004○**	**0.994±0.004**
letter	**0.958±0.021○**	**0.957±0.017**
mfeat	**0.921±0.069 ●**	**0.935±0.053**
mushroom	**1.000±0.000**	**1.000±0.000**
nursery	**1.000±0.000○**	**0.400±0.492**

○ Empty dot indicates the loss of USIBD.

● Bold dot indicates the win of USIBD;

6. CONCLUSIONS AND FUTURE WORK

In this work, we presented a set of novel contributions algorithm for decision trees. The proposed algorithm mimics human learning approach. We posited that by applying human learning in machine spaces will lead to an improved performance due to dynamic planning. To test this hypothesis we ran experiments on widely available datasets from UCI. In our future work, we will apply our research to more learning tasks, especially high dimensional feature learning tasks

REFERENCES

[1] N.V. Chawla, N. Japkowicz, A. Kotcz, Editorial: special issue on learning from imbalanced data sets, SIGKDD Explorations 6 (1) (2004) 1–6.

[2] H. He, E.A. Garcia, Learning from imbalanced data, IEEE Transactions on Knowledge and Data Engineering 21 (9) (2009) 1263–1284.

[3] Y. Sun, A.K.C. Wong, M.S. Kamel, Classification of imbalanced data: a review, International Journal of Pattern Recognition and Artificial Intelligence 23 (4) (2009) 687–719.

[4] C. Elkan, The foundations of cost–sensitive learning, in: Proceedings of the 17th IEEE International Joint Conference on Artificial Intelligence (IJCAI'01), 2001, pp. 973–978.

[5] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, F. Herrera, A review on ensembles for class imbalance problem: bagging, boosting and hybrid based approaches, IEEE Transactions on Systems, Man, and Cybernetics – part C: Applications and Reviews 42 (4) (2012) 463–484.

[6] N.V. Chawla, K.W.Bowyer, L.O.Hall, W.P.Kegelmeyer, SMOTE: synthetic minority over-sampling technique, Journal of Artificial Intelligent Research 16 (2002) 321–357.

[7] 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, SIGKDD Explorations 6 (1) (2004) 20–29.

[8] B. Zadrozny, C. Elkan, Learning and making decisions when costs and probabilities are both unknown, in: Proceedings of the 7th International Conference on Knowledge Discovery and Data Mining (KDD'01), 2001, pp. 204–213.

[9] 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.

[10] B. Zadrozny, J. Langford, N. Abe, Cost–sensitive learning by cost–proportionate example weighting, in: Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM’03), 2003, pp. 435–442.

[11] V. García, J.S. Sánchez, R.A. Mollineda, On the effectiveness of preprocessing methods when dealing with different levels of class imbalance, Knowledge Based Systems 25 (1) (2012) 13–21.

[12] A. Orriols-Puig, E. Bernadó-Mansilla, Evolutionary rule-based systems for imbalanced datasets, Soft Computing 13 (3) (2009) 213–225.

[13] N. Japkowicz, S. Stephen, The class imbalance problem: a systematic study, Intelligent Data Analysis Journal 6 (5) (2002) 429–450.

[14] H. He, E.A. Garcia, Learning from imbalanced data, IEEE Transactions on Knowledge and Data Engineering 21 (9) (2009) 1263–1284.

[15] Y. Sun, A.K.C. Wong, M.S. Kamel, Classification of imbalanced data: a review, International Journal of Pattern Recognition and Artificial Intelligence 23 (4) (2009) 687–719.

[16] A. Fernández, V. López, M. Galar, M.J. del Jesus, F. Herrera, Analysing the classification of imbalanced data-sets with multiple classes: binarization techniques and ad-hoc approaches, Knowledge-Based Systems 42 (2013) 97–110.

[17] M. Lin, K. Tang, X. Yao, Dynamic sampling approach to training neural networks for multiclass imbalance classification, IEEE Transactions on Neural
Networks and Learning Systems 24 (4) (2013) 647–660.

[18] A. Fernandez, S. García, J. Luengo, E. Bernadó-Mansilla, F. Herrera, Genetics-based machine learning for rule induction: state of the art, taxonomy and comparative study, IEEE Transactions on Evolutionary Computation 14 (6) (2010) 913–941.

[19] X. Chen, T. Fang, H. Huo, D. Li, Graph-based feature selection for object oriented classification in VHR airborne imagery, IEEE Transactions on Geoscience and Remote Sensing 49 (1) (2011) 353–365.

[20] C.-H. Tsai, L.-C. Chang, H.-C. Chiang, Forecasting of ozone episode days by cost-sensitive neural network methods, Science of the Total Environment 407 (6) (2009) 2124–2135.

[21] N. Kwak, Feature extraction for classification problems and its application to face recognition, Pattern Recognition 41 (5) (2008) 1718–1734.

[22] R. Batuwita, V. Palade, microPred: effective classification of pre-miRNAs for human miRNA gene prediction, Bioinformatics 25 (8) (2009) 989–995.

[23] R. Barandela, J.S. Sánchez, V. García, E. Rangel, Strategies for learning in class imbalance problems, Pattern Recognition 36 (3) (2003) 849–851.

[24] A. Estabrooks, T. Jo, N. Japkowicz, A multiple resampling method for learning from imbalanced data sets, Computational Intelligence 20 (1) (2004) 18– 36.

[25] R. Batuwita, V. Palade, Class imbalance learning methods for support vector machines, in: H. He, Y. Ma (Eds.), Imbalanced Learning: Foundations, Algorithms, and Applications, Wiley, 2013, pp. 83–96.

[26] R. Polikar, Ensemble based systems in decision making, IEEE Circuits and Systems Magazine 6 (3) (2006) 21–45.

[27] L. Rokach, Ensemble-based classifiers, Artificial Intelligence Review 33 (1) (2010) 1–39.

[28] J. Błaszczyń ski, M. Deckert, J. Stefanowski, S. Wilk, Integrating selective pre-processing of imbalanced data with ivotes ensemble, in: M. Szczuka, M. Kryszkiewicz, S. Ramanna, R. Jensen, Q. Hu (Eds.), Rough Sets and Current Trends in Computing, LNSC, vol. 6086, Springer, Berlin/Heidelberg, 2010,pp. 148–157.

[29] N.V. Chawla, A. Lazarevic, L.O. Hall, K.W. Bowyer, SMOTEBoost: Improving prediction of the minority class in boosting, in: Proceedings of 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'03), 2003, pp. 107–119.

[30] W. Fan, S.J. Stolfo, J. Zhang, P.K. Chan, Adacost: misclassification cost-sensitive boosting, in: Proceedings of the 16th International Conference on Machine Learning (ICML'96), Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999, pp. 97–105.

[31] A. Asuncion D. Newman. (2007). *UCI Repository of Machine Learning Database* (School of Information and Computer Science), Irvine, CA: Univ. of California [Online]. Available:http://www.ics.uci.edu /~mlearn/MLRepository.html.

[32] Witten, I.H. and Frank, E. (2005) Data Mining: Practical machine learning tools and techniques. 2nd edition Morgan Kaufmann, San Francisco.

AUTHOR'S DETAILS

Dr.Mohammad Imran received his B.Tech (CSE) in 2006 and M.Tech (CSE) in 2008 from JNTU, Hyderabad, His Research interests include Big Data Analytics, Artificial Intelligence, Class Imbalance Learning, Ensemble learning, Machine Learning and Data mining.He completed his Ph.D(CSE) in the department of Computer Science and Engineering, Rayalaseema University, Kurnool-518007, Andhra Pradesh. He is currently working as an Associate Professor in Department of CSE, Muffakham Jah College of Engineering and Technology, Banjara Hills, Hyderabad-500034, India. You can reach him at imran.quba@gmail.com.

Ms.Shama Kouser received her B.Engg (ECE) from Osmania University and M.Tech (DECS) in 2010 from Jawaharlal Nehru Technological University, Hyderabad, her research interests include Big Data Analytics, IoT, Security & Privacy, Machine Learning.She is currently working as a lecturer in Department of Computer Science, Jazan University, Jazan, Kingdom of Saudi Arabia.

Mr. Maradana Durga Venkata Prasad received his B.TECH (Computer Science and Information Technology) in 2008 from JNTU, Hyderabad and M.Tech. (Software Engineering) in 2010 from Jawaharlal Nehru Technological University, Kakinada, He is a Research Scholar with Regd No:1260316406 in the department of Computer Science and Engineering, Gandhi Institute Of Technology And Management (GITAM) Deemed to be University,Visakhapatnam,Andhra Pradesh, INDIA His Research interests include Clustering in Data Mining ,Big Data Analytics, Artificial Intelligence, Class Imbalance Learning, Ensemble learning, Machine Learning and Data mining.He is currently working as an Assistant Professor in Department of Information Technology, Muffakham Jah College of Engineering and Technology, Banjara Hills, Hyderabad-500034,Telangana,INDIA. He is also an industrial trainee where he teaches programming languages. He is the author of several research papers in the area of Software Engineering.