International Journal of Advanced Computational Engineering and Networking, ISSN: 2320-2106, Volume-5, Issue-4, Aprl.-2017
http://iraj.in

ARTIFICIAL ACTIVATION SYSTEM THE ENZYMATIC MODEL FOR CLASSIFICATION OF IMBALANCED DATA

ANITA KUSHWAHA

Computer Science and Engineering Department Birla Institute of Technology, MESRA, Ranchi
E-mail: a.kushwaha@bitmesra.ac.in

Abstract- Imbalanced Dataset is a very common problem in classification of data. In supervised learning many techniques have been developed to tackle the problem of imbalanced training sets. Such techniques have been divided into two groups: at algorithm level and at the data level. Data level groups emphasized are those that try to balance the training sets by reducing the larger class through elimination of samples or increasing the smaller ones by constructing new samples known as Under sampling and Over sampling respectively. This paper proposes a new hybrid method for the classification of imbalanced datasets through construction of new samples using the Synthetic Minority Over sampling technique together with the application of a new technique Enzyme-computation called Artificial Activation System. The proposed method Enzyme-computation has been comparatively studied, validated and supported by an experimental study and shows good results.

Keywords- Imbalanced Datasets, Oversampling, Under Sampling, rough set theory, Enzyme-computation model

I. INTRODUCTION

With the increasing amounts of data being generated by businesses and researchers there is a need for very fast, accurate and robust algorithms for data analysis of imbalanced datasets. Improvements in databases technology, computing performance, artificial intelligence have contributed to the development of intelligent data analysis. The primary aim of data mining is to discover patterns in data that lead to better understanding of the data generating process and to useful predictions and the pattern analysis. A new classification algorithm called Enzyme computation based on Enzymes is proposed in this study.

Enzymes are the basic machinery that catalyses the chemical reactions occurring in the living cells but remain unaltered in the substrate-product combination pair. Enzymes usually require activation energy which is the energy necessary to form transition state between reactants and products. They are protein macro-molecules which consist of chains of amino-acids ranging in length from one hundred to several thousands. These chains fold upon themselves and interact with other proteins to form a wide variety of structures. The structure and any subsequent modification of these amino-acids ultimately govern the enzyme function. Knowledge about the functioning of Enzyme which is essentially proteins (generally) is essential in the understanding of biological processes. Enzymes itself had beenclassified into various functional classes based on thetypes of reaction they catalyze. But alternative classification methods may need to be explored as class imbalance becomes a common problem in classification. The imbalanced data-set problem in classification domains occur when the number of instances that represents one class is much larger than the other classes. The minority class is usually more interesting from the point of view of the learning task.[6].There are many situations in which imbalance occurs between classes such as satellite image classification[19] risk management [17], medical applications[20] and so on, as a result this problem has been identified as a current challenge in Data mining[21].Recently support vector machine(SVM) has been used for predicting protein protein interaction [22], protein fold recognition, and protein structure prediction. Instead of directly analyzing sequences and doing classification on the physicochemical properties of a protein generated from sequence ,a new approach to data classification based on the enzymatic model called Artificial Activation System (AAS) which is a classification technique based on Enzymes is developed. Classification algorithms often achieve high accuracy with the majority class whereas with the minority class quite the opposite occurs. In imbalanced training sets, original knowledge often focuses on the minority class whereas many classifiers consider the less frequent data to be rarities or noise, focusing exclusively on the results of the global measures. [23, 24, 25]

Many techniques for dealing with class imbalance have arisen as a result of research and are grouped into two categories [6]: those at the level of the learning algorithm and those that modify data distribution (data level). This work introduces a new hybrid proposal for carrying out oversampling via SMOTE and under sampling over the synthetic instances for highly imbalance data-sets based on Enzyme-computation algorithm. Leonor Michaelis and Maud Menten published a set of equations believed to govern enzyme kinetics based on the concept of enzyme forming a noncovalent complex with its substrate before catalyzing the reaction and then dissociating from the product.

This chemical scheme is shown below:

$$E + S \rightleftharpoons E.S \rightleftharpoons E + P$$

Here E is the enzyme, S is the substrate, E.S is the Enzyme-substrate combine and P is the product that is the outcome of the reaction. Usually enzymes catalyses biological reactions. Sometimes co-factors also participate in catalysis. Vitamins are the organic co-factors & co-enzymes. Minerals are inorganic cofactors. According to Enzyme commission Enzyme has an EC number, which is a hierarchical number that distinguishes enzymes by the type of chemical reaction they catalyses. The EC groups enzymes into six broad classes that include

I. Oxidoreductase(EC1)-that catalyses oxidoreduction reaction in which there is transfer of electrons from one molecule to another

$$A + B: \rightarrow A: + B$$

II. Transferase(EC2)-catalyses the transfer of chemical group from donor to acceptor or transfer of a radical (group of electrons) from one molecule to another

$$A + BX \rightarrow AX + B$$

III. hydrolases(EC3)-catalyzes the hydrolysis of various bonds or combination of molecules or Addition or removal of water molecule (breaking of C-H bonds).

$$A + H20 \rightarrow B + C$$

IV. Lyases(EC4)-enzymes that cleaves bonds o ther than hydrolysis or breaking of C-C bonds. They catalyses lysis reaction that results in double bond.

$$A \rightarrow B + C$$

V. isomerases(EC5)-catalyses geometrical or structural changes within one molecule.

$$A \rightarrow B$$

VI. Ligases(EC6)-catalyses the joining of two molecules

$$A + B \rightarrow AB$$

So we find several factors to be considered in the Enzymatic model .These are:- the Enzyme (E), the substrate(S), the electron (e), the combination of electrons, weights (input weight due to which the electrons and radicals combine), the binding strength, the active site.Moreover, this Enzyme-substrate model which we call as the Artificial Activation System (AAS) can be used for various other applications too.

In Imbalanced Datasets, the problem of data imbalance occurs when number of instances in majority class is much larger than the other classes. The minority class is usually more interesting from the point of view of learning task .Classification algorithms often achieve high accuracy with the n majority class whereas with the minority class, quite the opposite occurs. In imbalanced training sets, the original knowledge often focuses on the minority class whereas many classifiers consider less frequent data to be rarities or noise, focusing exclusively on the results of the global measure [23, 24, 25] Many techniques for dealing with class imbalance have arisen as a result of research and are grouped into two categories: those at learning algorithm (algorithm level) and those that modify data distribution (data level).

This work introduces a new hybrid proposal forv carrying out oversampling via SMOTE and under sampling over the synthetic instances for highly imbalanced datasets called Enzyme-computation method and it is based on two steps:

-building new synthetic examples of the minority class using SMOTE and

-improving the quality of these new samples through Enzyme-computation algorithm, acting over the artificial instances of the minority class created by the SMOTE algorithm.

This work introduces a new preprocessing method using SMOTE to generate synthetic examples and Enzyme-computation as a cleaning method. The elimination of any synthetic example that does not belong to lower approximation of minority class is carried out as they are considered as noisy as they are not in the boundary region and are not useful for classification. We carried out the experiments in order to show the goodness of this model in comparison with SMOTE, SMOTE_ENN, SMOTETomek Links, Borderline-SMOTE1, Borderline- SMOTE2 and Safe-Level-SMOTE using several datasets from UCI-repository with high imbalance ratio. The measure of performance is based on AUC [15] and the significance of results is supported by the statistical analysis as suggested in the literature [10, 27]

In order to do this, the paper is organized as follows. Section 2 Related works, the imbalanced dataset problem is introduced and evaluation metric used in this work is discussed, some preprocessing techniques for imbalanced datasets are introduced. Section 3 describes the Enzyme-computation algorithm. Section 4 is the experimental study that is the benchmark datasets, the statistical tests for performance comparison and the experimental analysis in order to validate the goodness of our proposal is introduced. Section 5 is the conclusion section.

II. RELATED WORK

This problem is more closely related to cost-sensitive classification problem.[28,29,30] The classical machine learning algorithm may be biased toward the majority class and as a result may predict the minority class examples poorly. This problem is growing in importance and has been identified as one of the 10 main challenges of data mining. We focus our study on imbalanced datasets with binary classes that is there is only one positive and one negative

class considering the former to be one with lower number of examples and the latter the one with the higher number of examples. When multiple classes are present, the binary-class approach may be directly applicable via pair wise coupling techniques [32]. Specifically, in a previous work by [31] the authors carried out an experimental analysis in which it is shown that this methodology allows the achievement of a good behavior for processing in multiclass imbalanced datasets.

As mentioned earlier, the imbalanced dataset problem can be tackled using two main types of solution.

I. Solutions at the Data level [33, 7, 8, 34]: this kind of solution consists of balancing the class distribution by oversampling the minority class (positive instances) or under sampling the majority class (negative instances) or by applying hybrid models which combine the previous techniques.
II. Solutions at the algorithmic level: in this case we need to adapt our method to deal directly with the imbalance between the classes, for example, modifying the cost per class [35] or adjusting the probability estimation in the leaves of decision tree to favor the positive class [36].

In published research works, it has been shown that applying a preprocessing step in order to balance the class distribution is a positive solution to the problem of imbalanced datasets [31, 33]. Furthermore, the main advantage of these techniques is that they are independent of the classifier used. In [37], a system is presented that combines these two general solutions (data and algorithm level) obtaining good results. In this work we evaluate different instance selection methods together with oversampling and hybrid techniques to adjust class distribution in the training data. Specifically we have chosen the methods that have been studied in [33].These methods are classified into three groups:

- Under sampling methods
- "Tomek Links" [39] can be defined as follows: given two examples ei and ej belonging to different classes with d(ei,ej) the distance between ei and ej.A(ei,ej) pair is called Tomek link if there is no example El so that d(ei,ej) < d(ei,ej) or d(ej,el) < d(ei,ej).If two examples form a tomek link then either one of these examples is noise or both examples are borderline. Tomek links can be used as an under sampling method or as a data cleaning method .As an under sampling method, only examples belonging to the majority class are eliminated and as a data cleaning method, examples of both classes are removed.
- "Neighbourhood Cleaning Rule"(NCL) uses the Wilson's edited Nearest Neighbour Rule(ENN)[40] to remove majority class examples.ENN removes any example whose class label differs from the class of at least two of its three nearest neighbours. NCL modifies the ENN in order to increase the data cleaning for a two class problem the algorithm can be described in the following way : for each example ei in the training set,its three nearest neighbors are found. If ei belongs to the majority class and classification given by its three nearest neighbors contradict the original class of ei then ei is removed. If ei belongs to the minority class and its three nearest neighbors misclassify ei then the nearest neighbors that belong to the majority class are removed.

-oversampling methods

"Synthetic minority oversampling technique"(SMOTE) [7] is an oversampling method . Its main idea is to form a new minority class sample by interpolating between several minority class examples that lie together thus the overfitting problem is avoided and the decision boundaries for the minority class spread further into the majority class space. This method is described in detail in next part of this section

-hybrid methods oversampling plus under sampling

- **"SMOTE-Tomek links".** Frequently class members are not well defined as some majority class examples might invade minority class space. The opposite can also be true, since interpolating minority class examples can expand the minority class clusters, introducing artificial minority class examples too deeply into the minority class space . Inducing a classifier in such a circumstances can lead to over fitting. In order to create better defined class clusters, Bastista et al.[33] proposed applying tomek links to the oversampled training set as a data cleaning method. Thus removing only the majority class examples that form tomek links, examples from both classes are removed.
- **"SMOTE-ENN"** The motivation behind this method is similar to SMOTE-Tomek links.ENN tends to remove more examples than theTomek link do, so its is expected to provide a more in depth data cleaning. In constrast to NCL which is an undersampling method, ENN is used to remove examples from both classes. Thus any example that is misclassified by its three nearest neighbour is removed from the training set
- **"Borderline-SMOTE1"**.This method only oversamples or strengthens the borderline minority examples [41].First it finds out the borderline minority examples P, then synthetic examples are generated from them and added to the original training set. This method can be described as follows:
  for every minority example (pi)calculate its mb nearest neighbors from the whole training set if all the m nearest neighbors are majority example pi is considered to be noise and is not operated in

the following step. If m/2<m'< m namely the number of pi's majority nearest neighbors is larger than the number of its minority ones pi is considered to be easily misclassified and put into a set DANGER. If 0<m'<m/2 pi is safe and does not need to participate in the follows step. The examples in DANGER are the borderline data of the minority class P. Finally for each example in DANGER are the borderline data of minority class P. Finally for example in DANGER, we calculate its k-nearest neighbors from P and operate in a similar way to SMOTE.

- "Borderline-SMOTE2" This method is very similar to Borderline-SMOTE1, it not only generates synthetics examples from each example in DANGER and its positive nearest neighbors in P but also does so for its nearest negative neighbor in N (majority class)[41].The difference between it and its nearest negative neighbor is multiplied by a random number between 0 and 0.5 ;thus the newly generated examples are closer to the minority class.
- "Safe-Level SMOTE"; This method assigns each positive instance its safe level before generating synthetic instances[45].Each synthetic instance is positioned closer to the largest safe level before generating synthetic instances so all synthetic instance are generated only in safe regions.

2.1 Evaluation in imbalanced domains

The performance of machine learning algorithm is typically evaluated using predictive accuracy. However this is not appropriate when the data are imbalanced or when costs of different errors vary markedly [6]. Weiss and Hirsh[42] showed that the error rate of the classification of the rules of the minority class is 2 or 3 times greater than the rules that identify the examples of the majority class and that the examples of the minority class are less likely to be predicted than the examples of the majority one. Because of this, instead of using the error rate (or accuracy) in the context of imbalance problems more appropriate metrics are considered. A confusion matrix is a form of contingency table showing the difference between the true and predicted classes for a set of labeled examples that introduces the well-known measures; true positive(TP) true negative (TN) false positive(FP) false negative (FN).Using these we obtain the classical rate of accuracy and error as follows :

$$Error\ rate = \frac{FP+FN}{TP+FP+TN+FN}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$= 1 - Error\ rate$$

It is also possible to derive four performance metrics that directly measure the classification performance of positive and negative classes independently:

- True positive rate Tprate=TP/ (TP+FN) is the percentage of positive cases correctly classified as belonging to the positive class.
- True negative rate TNrate = TN/ (FP +TN) is the percentage of negative cases correctly classified as belonging to the negative class.
- False positive rate FPrate=FP/ (FP+ TN) is the percentage of negative cases misclassified as belonging to the positive class
- False negative rate FNrate=FN/ (TP + FN) is the percentage of positive cases misclassified as belonging to the negative class.

These four performance measures have the advantage of being independent of class costs and prior probabilities. The aim of a classifier is to minimize the false positive and negative rates or similarly to maximize the true negative and positive rates.

One appropriate metric that could be used to measure the performance of classification over imbalanced data sets is the Receiver Operating Characteristics (ROC) graphics [43].In these graphics the tradeoff between the benefits TPrate and costs FPrate can be visualized and it acknowledges the fact that the capacity of any classifier cannot increase the number of true positive without also increasing the false positives. The area under the ROC curve (AUC) [43] corresponds to the probability of correctly identifying which of the two stimuli is noise and which is signal plus noise.AUC provides a single number summary for the performance of learning algorithms.

2.2 SMOTE: synthetic minority oversampling Technique

The SMOTE algorithm [7] oversamples the minority class by taking each minority class sample and= introducing synthetic examples along the line segments joining any / all of k minority class nearest neighbors. With this approach, the positive class is oversampled by taking each minority class sample and introducing synthetic examples along the line segments joining any all of k minority class nearest neighbors. Depending upon the amount of oversampling required, neighbors from the K-nearest neighbors are randomly chosen. This process is illustrated in fig1. Where xi is the selected point, xi1 to xi4 are some selected nearest neighbors and r1 to r4 the synthetic data points created by the randomized interpolation. Synthetic examples are generated in the following way:-take the difference between the feature vector (sample) under consideration and its nearest neighbor. Multiply this difference by a random number between 0 and 1 and add it to the feature vector under consideration. This causes the selection of random points along the line segment between two specific features.

In Enzyme-computation algorithm, the oversampling occurs by introducing synthetic examples along the line segment joining any of the k minority class

nearest neighbour. we have taken the Average of all the data-points and then we compute the maximum distance from the average called dmax.So we take Avg- dmax.Also we compute the minimum distance form the average called dmin and we take Avg-dmin.

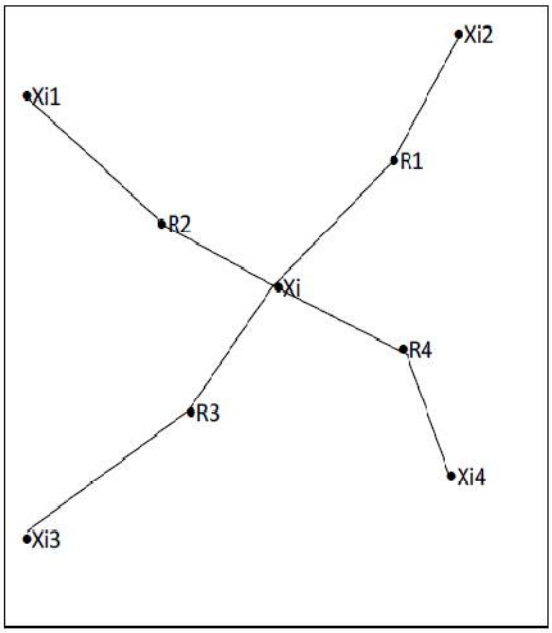


Fig.1. An illustration of how to create the synthetic data points in SMOTE algorithm

We discuss the SMOTE algorithm next .It is shown in Figure 1.The pseudo-code for Algorithm SMOTE appears as follows The Algorithm is detailed as follows

I. "Step 1" selects the number of minority class samples to be used for generating new instances when the degree of oversampling is lower than 100%.

II. "Step 2" is aimed to compute all the Kneighbors for each sample to be replicated.

III. Finally "Step 3"computes the interpolation as explained above. For the sake of clarity, an example of this usage in shown in fig. 2.

III. ENZYME COMPUTATION ALGORITHM

3.1 Enzymatic model for data classification: Artificial Activation system

In this section, we present a new proposal for making the distribution between classes in imbalanced training sets uniform. The hybrid method has two stages:

1. First, we create new instances using the SMOTE algorithm.

2. Second, a cleaning method based on Enzyme-Computation is applied to include the original examples and the synthetic minority examples that belong to the lower

Input: Number of Minority class sample T; nattrs Number of attributes; Sample [][] array of the original minority class samples, Amount of SMOTE N %:Number of nearest neighbors k.

Output: Synthetic [] and array of (N/100)*T synthetic minority class samples

Begin

1. If N < 100
   - 1.1 Randomize the T minority class sample
   - 1.2 Fill Sample[][] with the first(N/100)*T samples
   - 1.3 T=(N/100)*T
   - 1.4 N=100

   End if

2. For i → 1to T
   - 2.1 Compute K nearest neighbors for I and save the indices in an array nnarray
   - 2.2 Populate(N,I,nnarray)

   End for

3. While N ≠ 100 do
   - 3.1 Choose a random number between 1 and k,call it nn
   - 3.2 For attr ← 1 to nattrs
   - 3.3 Compute : *dif=Sample[nnarray[nn][attr]]-* Sample[i][attr]
   - 3.4 Compute : gap= random number between 0 and 1
   - 3.5 Synthetic[newindex][attr]=Sample[i][attr]+gap+dif

   End for

   newindex ++

4. N=N-1

End while

End

Fig. 2. Pseudo-code for SMOTE Algorithm

Consider a sample (6,4) and let (4,3) be its nearest neighbor.

(6,4) is the sample for which K-nearest neighbor are being identified (4,3) is one of its K-nearest neighbors.

Let f1_1 = 6 f2_1 = 4 , f2_1 – f1_1 = –2

f1_2 =4 f2_2 = 3 , f2_2 – f1_2 = –1

Figure 3.Example of the SMOTE algorithm

3. approximation of their class in the final training set(called resultset in the algorithm)

The algorithm uses the extended approach of Rough set theory [38, 44] based on similarity relations and includes six steps that we detail below:

1. "Step 1" uses SMOTE for oversampling the original dataset and it matches with the first stage.
2. "Step 2" just builds the final set of instances (output of the algorithm) by including initially the original ones of the data-set.
3. "Step 3" constructs the similarity matrix of all instances of the original dataset. The function used to determine the degree of similarity between two instances xi and xj is defined as follows(assigning its value to the similarity Matrix)

$$Similarity\ Matrix(i,j) = \frac{\sum\ wk * \delta k(xik, xjk)}{M}$$

Where n is the number of features, wk the weight of feature k ,xik and xjk are the values for feature k ,respectively,wk is the function of comparison for feature k, M is the number of features considered in the equivalence relation, B is the features set considered in the equivalence relation. The weight of a feature is defined as

$$wk = \begin{cases} 1\ if\ k \in . \\ 0\ otherwi. \end{cases}$$

δk is calculated for discrete attributes in the following way And for continuous attributes Where max Ak and min Ak are the extremes of the domain intervals for feature K.

4. "Step 4" analyzes which new synthetic data belong to the lower approximation, that is their similarity value is lower than a given threshold which means that there are no similar elements in the set and that they are added to the final "result set"

5. Finally "Step 5" checks whether all new generated instances are similar among them and return the final result-set as the output of the SMOTE algorithm.

Figure 5 shows the algorithm associated with the steps that are previously indicated. As indicated previously, resultset is the output set of the algorithm containing the original instances and the final synthetic instances of the training data-set and syntheticInstance is a vector containing the new instances generated by the SMOTE algorithm. This method can be found in the KEEL software tool so that any interested researcher can reproduce the experimental study. We have six procedures in the Enzyme-computation model In Procedure 1 ,we select electron and set weight =1, indicating that it is Enzyme EC_1. Similarly a procedure EC_2 in which we select radicals (which is nothing but combination of electrons(2*e)) is written and if there is a transfer of radicals then it is enzyme EC2 then set weight=2. And a procedure EC_3 in which combination of electrons and radicals into molecule occurs by setting weight =3 takes place ,is written. If weight =3(3*e).We select water molecule. If there is addition or removal of water molecule i.e breaking of C-H bond then we say it is enzyme EC3. And procedure EC_4 in which we set weight = 4 and select molecule if there is breaking of C-C bond (4*e) then we say it is enzyme EC4.And procedure EC_5 in which we set weight=5 and select molecule. If there is change in geometry or shape of the molecule then it is enzyme EC5.We also write a procedur EC_6 which is illustrated later in which a complex weight of 6 is chosen and combination of molecules take place.The Enzyme type is EC_6 by classification ligases. Thus these six classes of enzymes form the basis of classification of all imblanced data into six classesn after cleaning step is applied to the data. This classification algorithm is applied over Glass Dataset from UCI repository and gives 43% result of performance accuracy prior to cleaning step of data applied to it.On Iris Dataset the result was 33% before data cleaning step was applied to it. Then we apply SMOTE algorithm to generate synthetic instances and clean the data and apply Enzyme computation again. Figure 6 to Fig 6.1,6.2,6.3,6.4 shows the result of application of the algorithm to Glass Dataset.

Data set	Performance Accuracy
Glass	43%
Iris	33%

Table 1: Performance Accuracy for two Datasets

Note: An important note regarding Activation Energy and activation threshold is as follows Factors that affect the activation threshold of the reaction. While it may be energetically favorable to go from reactant to product, this only means that reaction will proceed not that it will go quickly. It isactually the activation energy which determines the rate at which the reaction proceeds. Enzymes stabilize transition states for reactions and thus lower the activation energy required. This has the overall effect of speeding up the reaction.A common measure for how much a reaction sped up is called the rate enhancement equal to the ratio of the catalyzed rate to unanalyzed rate. This ratio varies widely ranging from 1 to 1.4 x 107(which is technically no longer an enzyme-merely a protein to oritidine-monophosphate decarbooxylase (an enzyme involved in DNA synthesis). Activation energy is determined by various factors. According to Arrhenius equation K=Ae-Ea/(RT)

Where A is the frequency factor for the reaction, R is the universal gas constant, T is the absolute temperature. The enzyme is thought to reduce the path of the reaction. This shortened path would require less energy for each molecule of substrate converted to product. Given a total amount of available energy, more molecules of substrate would be converted

```
Algorithm:Enzyme-computation

Result

Use SMOTE for creating new instances using
SMOTE Algorithm

Min-Max values of each Enzyme-substrate
combine

/* The basic algorithm for Enzyme Computation
*/

Procedure EC_1

{

Initialize population matrix

And set  α and β

Call actual function evolution

Call procedure EC_1


While (EP <= Activation_threshhold)

{

 Select electron

If transfer of electron then EC1

Set wt =1

}

Rank energy kevel based on fitness

Get the best solution
```

Figure4. Pseudo-code for Enzyme-computation Algorithm

when enzyme is present than when it is absent. Hence the reaction is said to go faster in a given period of time. So we take EP<= Activation _thresholdAs Activation energy is the energy that is required to get reactions started since many reactions do not occur very quickly (or occur at all) if they are thermodynamically possible. We illustrate procedure EC_6 below: In the datasets we assume the first attribute to be Enzyme(E),the second attribute is taken as Substrate(S),the third attribute is electron(e).The input weights are respectively 1,2,3,4,5,and 6 for EC1,EC2, EC3,EC4,EC5 and EC6. Note: For our purpose we have taken the input weights in increasing order of complexity 1…6 as it applies to Enzyme EC1, EC2, EC3, EC4, EC5, EC6 and then classify data according to the rules for each of the six class. But input weight can also be variable also or can be entered at runtime by the user in a more complex system. The activation _threshold is chosen to lie between limits and .We illustrates the procedure EC_6 below:

IV. EXPERIMENTAL STUDY

In this section we first present the experimental framework, including the benchmark datasets, the parameters and the statistical tests used in order to carry out the performance comparison. Then we introduce experimental analysis which is divided into two parts: first we carry out an analysis of the parameters of our model then we develop the comparative analysis with some preprocessing techniques. The original algorithm that was chosen is the standard C4.5 algorithm for the classification of datasets which are imbalanced datasets.

4.1 Experimental setup: data-sets, parameters, and statistical tests.

In this section we briefly describe the datasets used for the experimental study and the statistical tests used alongside the experimental study. The learning algorithm used for the experimental study is Enzymecomputation which can be easily used for imbalanced problems.

```
Procedure EC_6

{

Initialize dataset=population in current generation
with fitness function resulting from actual
evolution

While (EP <= Activation_threshhold)

{

Set wt=6

Select molecule; if joining of 2 molecules then
EC6

}

Rank energy level based on fitness

Get the best solution


While ( Activation_threshhold <=α &&
Activation_threshhold > β)

{

Determine appropriate IF-THEN rule for a given
test sample


}
```

Figure 5. Pseudo-code for procedure 6 of Enzymecomputation algorithm.

4.1.1 Data-sets and parameters

To analyze our proposal, we considered 4 datasets from UCI repository with high imbalance rates. Multiclass datasets were modified to obtain two class non-balanced problems, so that the union of one or more classes of the minority class and the union of one or more classes of the minority class and the union of one or more of the remaining classes was labeled as majority class. The description of these datasets appears in Table 1(column IR indicates the imbalance ratio).

The sets were divided in order to perform a five folds cross-validation, 80% for training and 20% for testing where the 5 test data-sets form the whole set. For each data-set we consider the average results of the five partitions. Partition was carried out in such a way that the quantity of elements in each class remained uniform. For our experiment, we considered the following parameters for the Enzyme-computation algorithm:

-K: number of nearest neighbors that is fixed to 3

-Distance function to obtain the nearest neighbors, the Euclidean distance is used.

-The class distribution will be rebalanced to 50-50%

These parameters values are those recommended by the authors of the SMOTE algorithm [7] and therefore we have used them as a standard for our experimentation.

4.1.2 Statistical tests

In this paper, we use the hypothesis testing techniques to provide statistical support to the analysis of the results [13].Specifically we will use non-parametric tests, due to the fact that initial conditions that guarantee the reliability of the parametric tests may not be satisfied, causing the statistical analysis to lose credibility with these types of tests[10]For multiple comparisons, we use the Iman-Davenport test[18]to detect statistical differences among a group of results . Furthermore we consider the average ranking of the algorithms in order to show graphically how good a method is with respect to its partners. This ranking is obtained by assigning a position to each algorithm depending on performance for each data-set. The algorithm that achieves the best accuracy on a specific data-set will have first ranking (value 1) then the algorithm with the second best accuracy is assigned the rank 2 and so forth. This task is carried out for all data-sets and finally an average ranking is computed as the mean value of all rankings. These tests are suggested in the studies presented in [10,13] where its use in the field of machine learning is highly recommended. More information can be found at website http://sci2s.ugr.es/sicidm/ together with information on how to apply the statistical tests. 4.2 Enzyme-computation: parameter Analysis As previously introduced in Section.3,the similarity value was fixed during the algorithm runtime. The algorithm started with 0.4 value ;if the cleaning method does not find any instance in the lower approximation, the value is increased by 0.05 while the value remains lower than or equal to 0.9.This is a way to ensure the lowest similarity value but high enough to populate the lower approximation, obtaining good quality objects in lower approximation

4.3 Comparative Analysis.

In this section we compare the Enzyme-computation method with another six well-known preprocessor mechanisms based on SMOTE that is SMOTE algorithm itself and four hybrid approaches SMOTETomekLinks, SMOTE-ENN,Borderline-SMOTE1,Borderline-SMOTE2 and Safe-Level-SMOTE.

The results of the experimental study for the test partitions in shown in table 2 wherein the first column we have included the result over the original data-sets, the best method is highlighted in bold for each data-set. We can observe the goodness of the Enzyme-computation approach since it obtains the highest performance value for all the methodologies that are being compared. Additionally good results for SMOTE-TomekLinks and SMOTE-ENN with respect to SMOTE emphasize the significance of the cleaning step in the oversampling for achieving a superior behavior at the classification stage. Finally all the preprocessing approaches outperform the results with the original data-sets as expected.

Table 3 shows the total number of times every compared algorithm obtains the highest AUC value. There are two numbers per cell. The first number represents how many times the algorithm is the absolute winner while the second number denotes how many times it shares the highest AUC with the other algorithms (ties).The last column sums up the results in the two numbers. The first one indicates how many times the highest AUC is achieved by a single algorithm while the second one shows the total amount of data-sets where the highest AUC is shared.

Table 4 shows the ranking of the algorithms on each data-set selected for this study and for the original data-sets. In order to compare the results, we will use a multiple comparison test to find the best preprocessing algorithm. In table 5 we can observe that the best ranking is obtained by our proposal and the two last positions corresponds to Borderline-Smote1 and Borderline-Smote2.

An Iman-Davenport test is carried out (employing FDistribution with 6 and 258 degrees of freedom for Nds=44) in order to find the statistical differences among the algorithms, obtaining a p-value near to zero. In this manner, Table 6 shows the result of the Holm procedure for comparing our proposal with the remaining ones. The algorithms are ordered with respect to z-value obtained. Thus, by using the

normal distribution we can obtain the corresponding p-value associated with each comparison and this can be compared with the associated □/I in the same row of the table to show whether the associated hypothesis of equal behavior is rejected in favor of the best ranking algorithm, as we can observe the test rejects all cases. We can observe that our approach is statistically superior to all compared methods.

CONCLUDING REMARKS

In this paper we have presented a new proposal for editing training sets for highly imbalanced data-sets. The proposal belongs to the set of techniques known as hybrid oversampling and under sampling. The novelty of this proposal is that the quality of the new synthetic instances is evaluated using Enzyme computation method .This evaluation allows us to include only those artificial instances that are within the lower approximation of the minority class. From the results of our experimental analysis we have observed the good average results obtained by the Enzyme-computation technique for preprocessing within the framework of imbalanced data-sets.

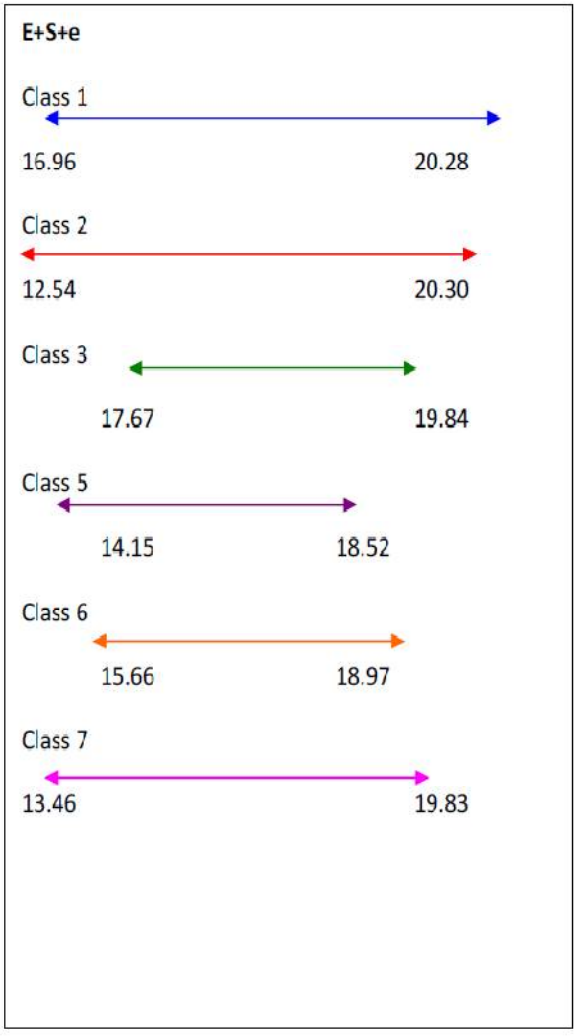


Figure 6 : Glass Dataset

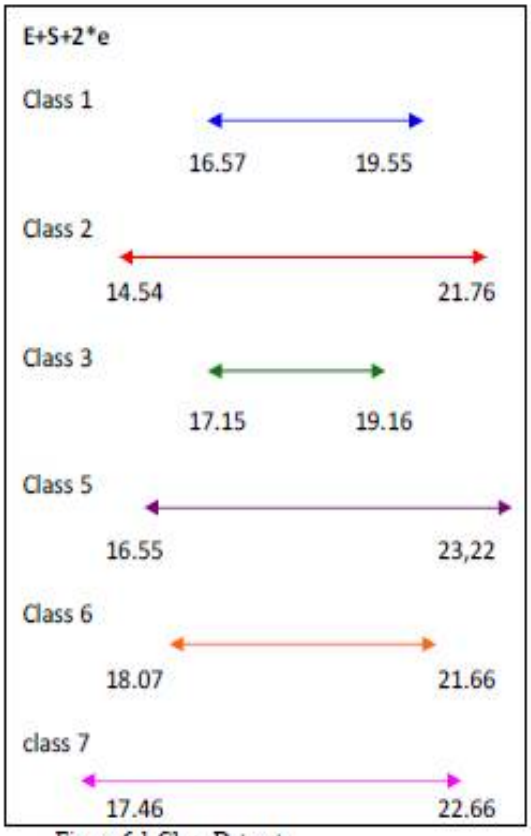
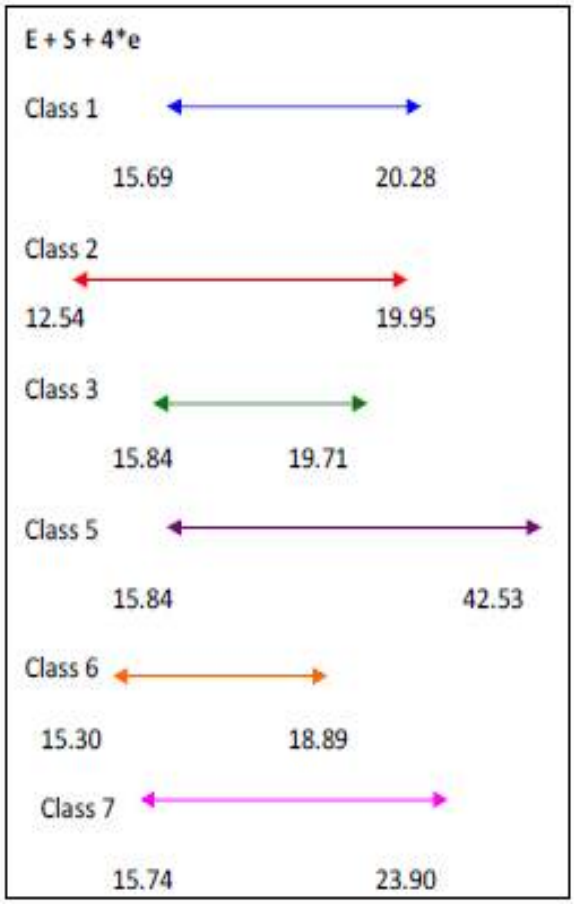


Figure 6.1 Glass Dataset



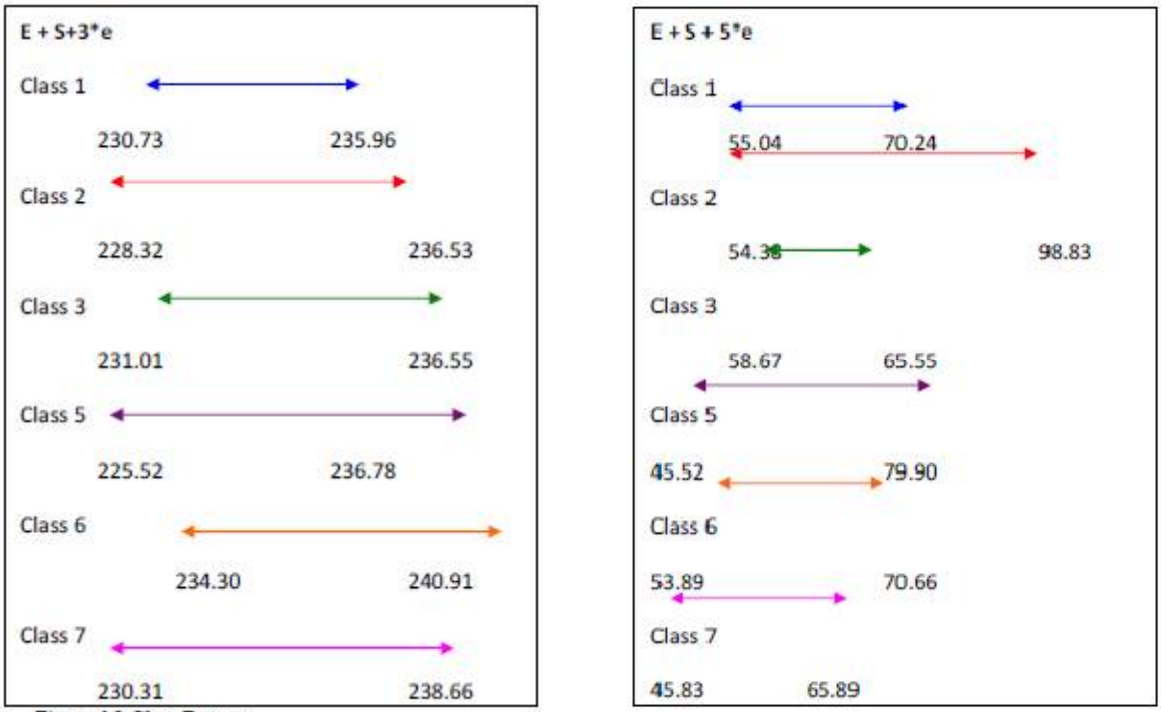


Figure 6.2 Glass Dataset

Data-set	*#Ex*	*#Attributes*	*%Class(min., maj.)*	*IR*
ecoli0137vs26	224	7	(2.49,97.51)	43.84
shuttle0vs4	1463	9	(6.72,93.28)	13.94
yeastB1vs7	367	7	(6.53,93.47)	14.3
shuttle2vs4	196	9	(4.65,95.35)	20.5
gass016vs2	153	9	(8.85,91.15)	10.29
glass016vs5	147	9	(4.89,95.11)	19.44
pageblock13vs4	472	10	(5.93,94.07)	15.85
yeast05679vs4	422	8	(9.66,90.34)	9.35
yeast1289vs7	757	8	(3.16,96.84)	30.5
yeast1458vs7	554	8	(4.33,95.67)	22.10
yeast2vs4	411	8	(9.92,90.08)	9.8
Ecoli4	278	7	(6.74,93.26)	13.84
Yeast4	1187	8	(3.43,96.57)	28.41
Vowel0	790	13	(9.01,90.99)	10.10
Yeast2vs8	381	8	(4.15,95.85)	23.10
Glass4	171	9	(6.07,93.93)	15.47
Glass5	171	9	(4.20,95.80)	22.81
Glass2	171	9	(7.94,92.06)	11.59
Yeast5	1187	8	(2.96,97.04)	32.78
Yeast6	1187	8	(2.49,97.51)	39.16
abalone19	3338	8	(0.77,99.23)	128.87
abalone918	584	8	(5.65,94.25)	29.76

Table 2 Description of the data-sets used in the experiments

Data-set	Original	Smote	S-TL	S-ENN	Border1	Border2	Safelevel
ecoli0137vs26	0.7481	0.8136	0.8136	0.8209	**0.8445**	0.8445	0.8118
shuttle0vs4	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	**0.9988**
yeast1R1vs7	0.6775	0.7003	**0.7371**	0.7277	0.6472	0.6437	0.6621
shuttle2vs4	**1.0000**	0.9917	1.0000	1.0000	1.0000	1.0000	1.0000
glass016vs2	0.5938	0.6062	**0.6388**	0.6390	0.5738	0.5212	0.6338
glass016vs5	**0.8943**	0.8129	0.8629	0.8743	0.8386	0.8300	0.8429
pageblock13vs4	**0.9978**	0.9955	0.9910	0.9888	0.9978	0.9944	0.9831
yeast05679vs4	0.6156	0.6832	0.6332	**0.7037**	0.6058	0.5473	0.5603
yeast1289vs7	0.6156	0.6832	0.6332	**0.7037**	0.6058	0.5473	0.5603
yeast1458vs7	0.5000	0.5367	0.5563	0.5201	0.4955	0.4910	**0.5891**
yeast2vs4	0.8537	0.8588	**0.9042**	0.9153	0.8635	0.8576	0.8647
Ecoli4	0.8437	0.8310	0.8544	**0.9044**	0.8358	0.8155	0.8386
Yeast4	0.6135	0.7004	0.7307	0.7257	0.7124	0.6882	**0.7945**
Yuwel0	0.9706	0.9494	0.9444	0.9455	0.9278	**0.9766**	0.9556
Yeast2vs8	0.5250	0.8066	0.8045	**0.8197**	0.6827	0.6968	0.8112
Glass4	0.7542	0.8500	**0.9150**	0.8650	0.7900	0.8325	0.9020
Glass5	**0.8976**	0.8829	0.8805	0.7756	0.8854	0.8402	0.8939
Glass2	0.7194	0.5424	0.6269	**0.7457**	0.7092	0.5701	0.6979
Yeast5	0.8833	0.9233	0.9427	0.9406	0.9118	0.9219	**0.9542**
Yeast6	0.7115	**0.8280**	0.8270	0.8270	0.7928	0.7485	0.8153
abalone19	0.5000	0.5202	0.5162	0.5166	0.5202	0.5202	0.5363
abalone918	0.5983	0.6215	0.6675	0.7193	0.7216	0.6819	0.8112

Table 3 : Comparison of the AUC results for preprocessing algorithms for test

	Original	SMOTE	S-TL	S-ENN	Border1	Border2	SafeLevel	Enzyme-Computation	Test
Test	1/3	3/1	4/1	4/2	0/4	5/2	7/1	15/3	39/5

Table 5 Winner Algorithm

Algorithm	Ranking
Enzyme-Computation	2.61364
S-ENN	3.92045
S-TL	3.96591
SMOTE	4.23864
Safelevel	4.34091
Borderline-SMOTE2	5.18182
Borderline-SMOTE1	5.36364

Table 6: Ranking Obtained through Friedman'test

i	Algorithm	Z=(R0 –Ri)/SE	P	Holm/Hochberg/Hommel	Hypothesis
6	Borderline-SMOTE1	5.2658490926	1.395E-7	0.008333	Reject
5	Borderline-SMOTE2	4.9176937807	8.756E-7	0.01	Reject
4	Safelevel	3.3074754631	9.414E-4	0.0125	Reject
3	SMOTE	3.1116381002	0.001860	0.016667	Reject
2	S-TL	2.5894051323	0.009614	0.025	Reject
1	S-ENN	2.5023663043	0.012336	0.05	Reject

Table 7 : Holm's table for □=0.05,Enzyme-computation is the control method

DATA –SETS	IST	2ND	3RD	4TH	5TH	6TH	7TH
ecoli0137vs26	BORDER2^	BORDER1^	S-ENN	S-TL	SMOTE	SAFELEVEL	ORIGINAL
shuttle0vs4	BORDER2	BORDER1	S-ENN	S-TL	SMOTE	ORIGINAL	SAFELEVEL
yeastB1vs7	S-TL	S-ENN	SMOTE	SAFELEVEL	BORDER1	BORDER2	ORIGINAL
shuttle2vs4	SAFELEVEL^	BORDER2^	BORDER1^	S-ENN^	S-TL^	ORIGINAL^	SMOTE
gass016vs2	S-ENN	S-TL	SAFELEVEL	SMOTE	ORIGINAL	BORDER1	BORDER2
glass016vs5	ORIGINAL	S-ENN	S-TL	SAFELEVEL	BORDER1	BORDER2	SMOTE
pageblock13vs4	ORIGINAL^	BORDER1^	SMOTE	BORDER2	S-TL	S-ENN	SAFELEVEL
yeast05679vs4	SAFELEVEL	S-TL	SMOTE	S-ENN	BORDER1	BORDER2	ORIGINAL
yeast1289vs7	S-ENN	SMOTE	S-TL	ORIGINAL	BORDER1	SAFELEVEL	BORDER2
yeast1458vs7	SAFELEVEL	S-TL	SMOTE	S-ENN	ORIGINAL	BORDER1	BORDER2
yeast2vs4	S-ENN	S-TL	SAFELEVEL	BORDER1	SMOTE	BORDER2	ORIGINAL
Ecoli4	S-ENN	S-TL	ORIGINAL	SAFELEVEL	BORDER1	SMOTE	BORDER2
yeast4	SAFELEVEL	S-TL	S-ENN	BORDER1	SMOTE	BORDER2	ORIGINAL
Vowel0	BORDER2	ORIGINAL	SAFELEVEL	SMOTE	S-ENN	S-TL	BORDER1
yeast2vs8	S-ENN	SAFELEVEL	SMOTE	S-TL	BORDER2	BORDER1	ORIGINAL
Glass4	S-TL	SAFELEVEL	S-ENN	SMOTE	BORDER2	BORDER1	ORIGINAL
Glass5	ORIGINAL	SAFELEVEL	BORDER1	SMOTE	S-TL	BORDER2	S-ENN
Glass2	S-ENN	ORIGINAL	BORDER1	SAFELEVEL	S-TL	BORDER2	SMOTE
Yeast5	SAFELEVEL	S-TL	S-ENN	SMOTE	BORDER2	BORDER1	ORIGINAL
Yeast6	S-TL	SMOTE	S-ENN	SAFELEVEL	BORDER1	BORDER2	ORIGINAL
abalone19	SAFELEVEL	BORDER2	BORDER1	SMOTE	S-ENN	S-TL	ORIGINAL
abalone918	SAFELEVEL	BORDER1	S-ENN	BORDER2	S-TL	SMOTE	ORIGINAL

TABLE 4.Perormance ranking for test

REFERENCES

[1] V.N.Vapnik,"The nature of statistical Learning theory", Springer, second ed.,1999
[2] C.J.C. Burges,"A tutorial on support vector machine for pattern recognition,"Data Min.Knowl.Disc 2,1998
[3] Hulse J.,Khoshgoftar,T.,Napolitano(2007),"A experimental perspectives on learning from imbalanced data",In proceedings of the 24th International Conference on Machine learning,pp.935-942
[4] Haibo He,Edwardo A. Garcia(2009) ," Learning from imbalanced Data"
[5] C.V.KrishnaVeni,T.Sobha Rani (2011)"On the classification of imbalanced Datasets",IJCST Vol2,Sp
[6] 1,December 2011
[7] Nitesh V. Chawla,Nathalie Japkowicz,Aleksander Kolcz(2004),"Editorial :Special Issue on learning from imbalanced Datasets" SigKdd Explorations,Vol 6,Issue 1-6
[8] Chawla NV, Bowyer KW,Hall LO,Kegelmeyer WP (2002),"SMOTE: Synthetic minority over sampling technique",J Artif Intell Res 16: 321-357
[9] Nitesh V. Chawla • David A. Cieslak, Lawrence O. Hall • Ajay Joshi (2008)" Automatically countering
[10] imbalance and its empirical relationship to cost", www3.nd.edu/~dial/publications/chawla2008
[11] Chen Y-S,Cheng C-H(2010)," Forecasting PGR of the financial industry using a rough sets classifier based on attribute-granularity", Knowledge and Information Systems archive,Volume 25 Issue 1, October 2010 ,Pages 57-79

[12] Demsar J(2006),"Statistical Comparisons of Classifiersover Multiple Data Sets", mlr.csail.mit.edu/…volume7/demsar06a/demsar06a. df
[13] Alberto Fernández, García S, María José del Jesus, Francisco Herrera(2008),"A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets", dl.acm.org/citation.cfm?id=1390886
[14] Ron Kohavi,"A study of cross-validation and bootstrap for accuracy estimation and model selection", IJCAI'95 Proceedings of the 14th international joint conference on Artificial intelligence ,- Volume 2
[15] Garcia S,Fernandez A,Luengo J,Herrera F,"A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability"Soft Comput 13(10): 959-977
[16] Greco S,"Rough sets theory for multicriteria decision analysis",Eur J Oper Res 129:1-47
[17] Huang J,Ling CX," Using AUC and Accuracy in Evaluating Learning Algorithms",IEEE Trans Know Data Eng 17(3): 299-310
[18] Maŕıa Jose´ Del Jesus, Francisco Herrera , "Improving the performance of fuzzy rule based classification systems for highly imbalanced data-sets using an evolutionary adaptive inference system. Bio-
[19] InspiredSystems",http://dx.doi.org/10.1016/j.nonrwa 20 05.04.006
[20] Huan YM, Hung CM,Jiau HC,"Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem", http://dx.doi.org/10.1016/j.nonrwa.2005.04.006(200)
[21] Iman R,Davenport J(1980),"Approximations of the critical region of the Friedman statistic",Commun Stat Part A Theory Methods 9: 571-595
[22] Suresh S,Sundararajan N,Saratchandran P (2008) Risksensitive loss functions for sparse multi-category classification problems,Inf Sci 178(12):2621-2638 20. Mazurowski M,Habas P,Zurada J,Lo J,Baker J, Tourassi G (2008) Training neural network classifiers for medical decision-making: the effects of imbalanced datasets on classification performance.Neural Netw 21(2-3): 427-436
[23] Yang Q,Wu X (2006) 10 challenging problems in data mining research.Int J Inf Technol Decis Mak 5 (4):597- 604
[24] Alun Thomas , Rob Cannings , Nicholas A. M. Monk and Chris Cannings,On the structure of protein-protein interaction networks, arxiv.org/pdf/q-bio/0309012
[25] He H,Garcia EA (2009) Learning from imbalanced data.IEEE Trans Knowl Data Eng 21(9) :1263-1284
[26] Orriols-Puig A,Bernado-Mansilla E (2009) Evolutionary rule-based systems for imbalanced datasets.Soft Comput 13(3) :213-225
[27] Sun Y,Wong AK,Kamel MS (2009) Classification of imbalanced data: a review.Int J Pattern Recognit Artif Intell 23(4):687-719
[28] Garcia S,Fernandez A,Luengo J,Herrera F (2010) Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining :experimental analysis of power.Inf sci 180:2044- 2064
[29] Garcia S,Herrera F(2008) An extension on statistical comparisons of classifiers over multiple datasets for all pairwise comparisons. J Mach Learn Res 9: 2677-2694
[30] Ling C, Sheng V(2006) Test strategies for costsensitive decision trees. IEEE Trans Knowl Data
[31] Engg 18(8):1055-1057
[32] Sun Y,Kamel MS,Wong AK,Wang Y (2007) Cost-sensitive boosting for classification of imbalanced data.Pattern Recognit 40:3358-3378 18
[33] Zhou Z-H,Liu X-Y (2006) Training cost-sensitive neural networks with methods addressing the class imbalance problem .IEEE Trans Knowl Data Eng 18(1):63-77
[34] Fernandez A,del Jesus MJ, Herrera F(2010) Multi-class imbalanced data-sets with linguistic fuzzy rule based classification systems based on pairwise learning.13th International Conference on Information Processing and Management of Uncertainity in knowledge-Based Systems(IPMU2010) LNAI 6178 pp 89-98 159(18):2378-2398
[35] Furnkranz J (2002) Round Robin classification J Mach Learn Res 2: 721-747
[36] Batista GEAPA,Prati RC,Monard MC(2004) A study of the behavior of several methods for balancing machine learning training data. SIGKDD Explor 6(1): 20-29
[37] Garcia S,Herrera F(2009) Evolutionary undersampling for classification with imbalanced data-sets:proposals and taxonomy .Evol Comput 17(3): 275-306
[38] Gryzmala-Busse JW,Stefanowski J,Wilk S(2005) A comparison of two approaches to dat mining form imbalanced data.J Intell Manuf 16(6) :565- 573
[39] Weiss GM,Provost F(2003) Learning when training data are costly:the effect of class distribution on tree induction J. Artif Intell Res 19: 315-354
[40] Wang BX,Japkowiccz N (2010) Boosting Support Vector machines for imbalanced datasets.Know Inf Syst 25(1):1-20
[41] Tsumoto S(2003) Automated extraction of hierarchical decision rules from clinical databases using rough set model. Expert Syst Appl 24:189- 197
[42] Tomek I (1976) Two modifications of CNN .IEEE Trans Syst Man Commun 6: 769-772
[43] Wilson DL (1972) Asymptotic properties of nearest neighbor rules using edited data .IEE E Trans Syst Man Commun 2(3):408-421
[44] Han H, Wang WY, Mao BH (2005) Borderline- SMOTE : a new over-sampling method in imbalanced datasets learning.International conference on intelligent computing (ICIC05) LNCS 3644.Springer,pp 878-887
[45] Weiss GM, Hirsh H (2000) A quantative study of small disjuncts In: Proceedings of the 17th national conference on artificial intelligence pp 665-670
[46] Bradley AP (1997) The use of Area under the ROC Curve in the evalautaion of machine learning algorithms.Pattern Recognit 30(7):1145- 1159
[47] Pawlak Z (1982) Rough Sets. Int J Comput Inf Sci 11: 145-172
[48] Bunkhumpornpat C,Sinapiromsaran K,Lursinsap C(2009)' safe level SMOTE:safe-level-synthetic minority over-sampling technique for handling class imbalanced problem'Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD09) LNCS 3644,S pringer ,pp 475-482.

★★★