Applied Mathematics & Information Sciencess
An International Journal

http://dx.doi.org/10.12785/amis/080103

Equilibrating the Recognition of the Minority Class in the Imbalance Context

L. Cleofas-Sánchez[1,*], *O. Camacho-Nieto*[2], *J. S. Sánchez-Garreta*[3], *C. Yáñez-Márquez*[1] *and R. M. Valdovinos-Rosas*[4]

[1] Centro de Investigación en Computación, Instituto Politécnico Nacional, Av. Juan de Dios Bátiz s/n, Col. Nueva Industrial Vallejo, 07738 México D.F., México
[2] Centro de Innovación y Desarrollo Tecnológico en Cómputo, Instituto Politécnico Nacional, Av. Juan de Dios Bátiz s/n, Col. Nueva Industrial Vallejo, 07700 México D.F., México
[3] Institute of New Imaging Technologies, Department of Computer Languages and Systems, Universitat Jaume I, Av. Vicent Sos Baynat s/n, 12071 Castellón de la Plana, Spain
[4] Universidad Autónoma del Estado de México, Cerro de Coatepec s/n, Ciudad Universitaria 50110, Toluca, Estado de México

Received: 20 May. 2013, Revised: 3 Sep. 2013, Accepted: 4 Sep. 2013
Published online: 1 Jan. 2014

Abstract: In pattern recognition, it is well known that the classifier performance depends on the classification rule and the complexities presented in the data sets (such as class overlapping, class imbalance, outliers, high-dimensional data sets among others). In this way, the issue of class imbalance is exhibited when one class is less represented with respect to the other classes. If the classifier is trained with imbalanced data sets, the natural tendency is to recognize the samples included in the majority class, ignoring the minority classes. This situation is not desirable because in real problems it is necessary to recognize the minority class more without sacrificing the precision of the majority class. In this work we analyze the behaviour of four classifiers taking into a count a relative balance among the accuracy classes.

Keywords: *CHAT* Associative model, Class Imbalance, Bayesian Network (*BN*), Multilayer Perceptron (*MLP*) and Radial Basis Functions Network (*RBFN*).

1 Introduction

In pattern recognition, the class imbalance is as a big classification problem. In this context, the classifiers commonly assume that the distribution of classes are balanced in the data sets; this situation in real problems is not true (detection of oil spills, medical diagnosis, face recognition, among others [1]). For example, in a medical problem when the amount of healthy cases (900) included in the majority class is higher than the ill cases (100) included in the minority class. Both classes are important, but in this example the classifier may skew its learning to the majority class, and as a consequence the patterns of the minority class will be ignored [1].

The class imbalance problem in some cases is correlated with other problems in the training data, such as class overlapping, size data sets, small disjoint, high dimensionality, and others [2]. Also the classifier behaviour depends on the classification rule. For example, some algorithms generalize knowledge such as algorithms for training trees (C4.5) [3], neural networks, support vector machine among others. This situation is presented mainly when the training data set is imbalanced due to the tendency to assign a certain test sample to the most represented class [43], [5], [1], [6].

This study is focusing on four models which generalize knowledge: three neural networks and one associative memory. Artificial Neural Networks (*ANN*) are mathematical models inspired in the functioning of the human brain, simulating the interconnection existing between the neurons, which allow the information process. The learning process of the Artificial Neural Networks is realized in parallel through the interconnection made between the node layers. For the majority, it is not necessary that the neural network is trained twice, and their knowledge is obtained with the adjustment of the weights [7]. In this sense, some network models are very useful in classification issues, such as the

[*] Corresponding author e-mail: li_rmvr@hotmail.com

Bayesian Network, the Multilayer Perceptron and the Radial basis Functions Networks [8] and [9] and [10].

On the other hand, the associative Memories have the ability of correctly recovering the input patterns; for this, the associative models take into account two phases: learning phase and recovery phase. In the first phase, the associative memories show their learning as a matrix, which represents the associations made among the input patterns (vectors of n components or features) and the output patterns (classes). In the second phase, the input patterns are recovered [11].

Some approaches proposed for handling the imbalance problem are focused on increasing the amount of samples in the minority class (over-sampling), diminishing the amount of samples in the majority class (under-sampling) or biasing the classifier behaviour in the training step in order to identify the minority class better [1]. The first method randomly duplicates minority samples with the aim of making a balance in the classes. The second method randomly eliminates majority class patterns. The third method consists in modifying the cost associated with the erroneous minority class classification [3].

The previous techniques are widely used. However, all are not considered to obtain a relative balance between the performances of each class. That is to say, in some cases, to apply a certain method can invert the imbalance. Therefore, the majority class becomes a minority and the minority becomes the majority [12]. This situation is not desirable, because the imbalance problem was not resolved, only inverted.

Taking in to account the neural approach, some works have been performed in the class imbalance context. In this sense, the researches made by [13] show an improvement in the imbalanced data classification considering the method called Principal Component Analysis (*PCA*) before adding Gaussian noise in the samples used to network learning. In another work, the redundant samples belonging to the majority are eliminated through the method called stochastic sensitivity measure, and in this manner improve the performance of the Radial Basis Function Neural Network class [14].

Few works have been found in relation with the Associative models and the issues implicit in the data sets such as class imbalance, outliers, high-dimensional data sets among others. The first work analyzes the performance of the *HACT* model taking into account the geometric mean and under-sampling methods. This is made on eleven imbalanced data sets [15]. On the other hand, [16] have considered feature selection methods to try the data sets before training the *HACT* model.

In terms of a balanced recognition between the class rates, this work analyzes the behaviour of four models which generalize knowledge. Specifically, the Hybrid Associative Classifier with Translation (*HACT*) and the three well known neural classifiers (such as Bayesian Network, Multilayer Perceptron and Radial Basis Functions Network) are considered. Experiments with thirteen data sets of real-life, show that the better classifiers performance is more noted when a previous preprocessing in the imbalanced data sets is made. These results are obtained considering a balanced recognition. In this sense, the accuracy of the minority class is increased without significantly diminishing the accuracy of the majority class.

The paper is structured as follow, in section 2 the *HACT* model is exhibited; in section 3 the neural models are described. In this way, the preprocessing methods are shown in section four. Then, the experimental set-up and experimental results are presented in section five and six. Finally, the main concluding remarks are expressed in section seven.

2 Hybrid Associative Classifier(HACT)

The *HAC* model combines two associative memories: Learn Matrix and Linear Associator. The first associative memory requires that the input patterns must be binary vectors. The second associative memory necessitates that the input patterns must be orthonormal vectors. Those aspects are considered as disadvantages of those models. Therefore, the *HAC* model arose to cover those drawbacks. Additionally, the model considers a low computational cost in its process of recognition [11].

The disadvantage of the *HAC* model is presented when some input class patterns have a big magnitude in comparison to the magnitude of other input patterns belonging to another class. In this case, the input patterns with less magnitude will be assigned to the class of those patterns with a bigger magnitude. To correct the limitations, of the *HAC* associative model, the translation of axis was implemented in the *HACT* model. The translation of axis occurs when parallel axis are found.

To carry out the procedure of the *HACT* associative model, the mean vector is obtained from all input patterns. The mean vector works as the centre of a new axis coordinate. In this way, a new data set is generated. The mean vector is obtained through $\bar{x} = \frac{1}{p}\sum_{j=1}^{p} x^{\mu}$, and the translation of axis is made with $x^{\mu'}$=x^{μ} -$\bar{x}$ [11].

The *HACT* associative model obtained its learning taking into account the first phase of the Linear Associator model [17] where the external product is utilized to obtain the associations among input patterns and output patterns. The final matrix represents the learning of the *HACT* model, which is obtained through the sum of all external products:

$$M = \sum_{\mu=1}^{p} (y^{\mu})(x^{\mu})^{t} \qquad (1)$$

The recovery phase of the *HACT* model is made through the second phase of the Learnmatrix associative model: using the matrix obtained in the learning phase of the *HACT* model and the input patterns.

3 Neural Networks

The following subsections three neural networks such as the Bayesian Network, the Multilayer Perceptron and the Radial Basis Functions Network being described. Their main characteristics involve the following aspects: the first network considers the probability theory for its learning, the second network takes into account a single hidden layer in its topology and the third network uses function nodes in its hidden layer.

3.1 Bayesian Network

The probabilistic approach called Bayesian Network (*BN*) was developed by Pearl in 1980. This has been widely used in pattern recognition as a robust classifier. The *NB* operation is realized through a network structure, taking into account the conditional probability (considering an aprior knowledge) in their training and considering the Bayesian theorem in the classification [25] and [19] and [20] and [21]. The *BN* approach can be seen as:

$$BN = (DAG, P) \tag{2}$$

where the *DAG* represents a directed acyclic graph topology and the symbol ”*P*” indicates the conditional probabilities.

Besides, it is of great importance to mention that the approach exhibits the best variable probability, which is distributed throughout the network. Each random variable (events) is represented as an independent network node [22] and [23] and [24]. Additionally, *BN* cannot obtain a best network structure when there is a high dimension in the features space [25].

3.2 Multilayer Perceptron

The networks model has been widely used in pattern recognition for its generalization ability. In this case, the Multilayer Perceptron (*MLP*) was developed as a nonlinear network model organized by layers such as the input layer, the hidden layer and the output layer. The first layer is integrated by input units that represent the attribute examples. The nodes of the second layer allow obtaining several decision boundaries and these are combined to obtain a classification decision. Finally, in the output layer, all output nodes have a zero value except in the node that indicates the class [26] and [27] and [28].

The training of *MLP* network has been widely performed with backpropagation taking into account the gradient descent in the error function; minimizing the error function. On the other hand, literature says that if the network training stays in a local minimum then the posteriori probabilities cannot be obtained [27] and [29]. In addition, the classification examples are obtained through the output network nodes.

3.3 Radial Basis Functions Network

The Radial Basis Functions Network (*RBFN*) is a Feedforward network well known in Pattern Recognition, which emerges from research made by Broomhead, Lowe, Lee among other authors [7]. The *RBF* network topology is formed by an input layer, a hidden layer and an output layer. The hidden layer of the *RBF* is integrated by Kernel functions nodes (each node is associated with different weights) instead of considering single hidden nodes such as the MLP network. Traditionally, the Basis function used in the nodes of the hidden layer has been the Gaussian function [7]. In addition, the *RBF* Network is faster in its learning process than that used by the *MLP* Network [30].

The learning process of the *RBF* Network takes into account a basis function to map the input samples to the hidden layer nodes. Thus, the function can been seen as $\phi \|x - x^n\|$, where the symbol ϕ indicates the non-linear function and the distance (for example the distance Euclidean) is expressed through $\|x - x^n\|$ [30].

The learning of *RBF* network is not finished until the parametres are adjusted in the network. In addition, the error must be reduced until an minimum error value is obtained [32].

4 Preprocessing Methods

Traditionally, the imbalance issue has been tried at algorithm level, at sampling level and using cost-sensitive methods. In the first method, the minority class is handled inside the algorithm. In this case, a modification is made to the algorithm, for this is necessary to know the classifier rule and application domain. Some authors mention that the preprocessing methods are positive solutions to balance the class distribution. When the sampling method is applied it is not important to know the classifier rule inasmuch as the method treats the class imbalance inside data sets [33]. The cost-sensitive technique combines the previous methods, taking into account the cost of misclassification in the learning phase or modifying the algorithm considering the cost on the classification [1].

A preprocessing method included at sampling level is the Smote (Synthetic minority oversampling technique). This approach was proposed by Chawla et al., which is an oversampling method that generates synthetic examples of minority class through a random interpolation [34]. This is performed until a balance among classes is obtained. The procedure to obtain the synthetic examples consists in taking the distance between the current example and one of their *k*-nearest neighbours (it is selected randomly). After that, the differences vector is multiplied by a value between zero and one. Next, the synthetic examples are incorporated [1]. It is important to mention that the Smote method alleviates the overfitting problem generated by random oversampling methods.

This issue is presented when the examples are duplicated and it does not generate new information in the data sets [35]. The Smote method can be seen as [36]:

```
–$O$ original data set.
–$P$ minority class.
–Begin
    1.For each sample $x$ in $P$
    2.Find the $k$-nearest neighbors to $x$ in $P$
    3.Obtain $y$ by randomizing one from $k$ samples
    4.difference = $x$- $y$
    5.$gap$ = random number between 0 and 1
    6.$n$ = $x$+ $difference*gap$
    7.Add $n$ to $O$
    8.End for
–End
```

The Wilson's Edited Nearest Neighbour Rule is an undersampling method proposed by Wilson in 1972. This method eliminates the atypical examples near the decision boundary [37]. For this, the Wilson method uses the classifier called Nearest Neighbour to obtain the class label of the training examples. In this case, if the current example label does not correspond to the label of their k-nearest neighbours, then the current example is eliminated. It is important to mention that the majority class decreases slightly in the number of the examples when the method searches their nearest neighbours inside of the majority class [38]. The Wilson method is expressed as Wilson:1972:

```
–Input: $M$ = Data set original, $k$ = k-nearest neighbors.
–output: $S$ = CD Edited.
–begin
    1.$S$ = $M$
    2.For each $x_i$ in $M$ do Discard $x_i$ of $S$ if this is
      misclassified using the $k$-nearest neighbors- $NN$
      on $M - x_i$
    3.End For
–End
```

5 Experimental set-up

The goal of this work is to analyze the performance of four classifier (such as *CHAT*, *BN*, *MLP* and *RBF*) in the context of a balanced recognition among classes. It is important to mention that the parametre values of the *BN*, *MLP* and *RBF* networks were obtained automatically with Weka. In addition, the classifiers performance was evaluated with the geometric mean and Area under the *ROC* (Receiver Operating Characteristics).

The experiments were made taking into account a previous preprocessing in the data sets. Thus, the Wilson oversampling eliminates the patterns that are near to the decision boundary. On the other hand, with the Smote method, the examples of the minority class are increased.

5.1 Description of data sets

The data sets were taken from the *KEEL* repository (http ://www.keel.es/dataset.php), specifically from the imbalanced data sets section. All data sets are class-two problems with different characteristics such as the imbalance rate (*IR*), the features dimensionality (or the number of features (*F*)) and the data sets size (or number patterns (*P*)). This can be seen in the following Table 1:

Table 1 Data sets

Data sets	F	P	IR
Wisconsin	9	214	1.86
Haberman	3	306	2.78
Vehicle1	18	846	2.90
Glass0123_456	9	214	3.20
Vehicle3	18	846	3.00
Ecoli1	7	336	3.36
Glass6	9	214	6.38
yeast0256_3789	8	1004	9.14
Glass04_5	9	92	9.22
Shuttle-c0_c4	9	1829	13.87
Glass4	9	214	15.47
Yeast1458_7	8	693	22.10
Yeast-2_8	8	482	23.10

The data sets are sorted by the level of the class imbalance presented. The *IR* is obtained by dividing the number patterns of the minority class (*Min*) between the number of patterns of the majority class (*May*); this can be seen as $IR = Min/May$. In literature a high *IR* is considered when there is a value greater than ten. It is possible to observe in the Table 1 a high imbalance rate on four data sets. In addition, the method k-cross-validation was considered to obtain five partitions of each original data set.

5.2 Performace Measures

In this section the measures for checking the performance of the classifier in the imbalance context are described. In this paper, two performance measurements are used such as the geometric mean and the *ROC* curve (*AUC*) to evaluate the neural networks and the *HACT* model performance.

Traditionally, the overall accuracy (*Acc*) has been used in the balanced data sets context. However, it is not appropriate to use the *Acc* measure in imbalanced data sets, because the classification model would not consider the correct classification from each class separately. In this way, it is possible to obtain a classification model which reports an accuracy of 90% with a very high imbalance rate. The overall accuracy is expressed as the number of patterns classified correctly (all classes) among the total patterns in the test data sets [41]:

$$Acc = \frac{TP+TN}{TP+FN+TN+FP} \tag{3}$$

where *TP* and *TN* indicate the correct classification of the minority and majority classes. The misclassification of both classes is expressed as *FP* (minority class) and *FN* (majority class).

On the other hand, a measure which considers the accuracies by class is geometric mean. This measure takes into account a symmetric distribution over the negative recognition rate (TN_r=TN/TN+FP) and positive recognition rate (TP_r=TP/TP+FN) [41].

$$MG = \sqrt{(TP_r)*(TN_r)} \tag{4}$$

In some cases the geometric mean can obtain a partial solution when some of the rates have a zero value. In this case, the most accuracy is provided by one class. This disadvantage can be resolved with the Area under the *ROC* (Receiver Operating Characteristics) or *AUC*. This measure is used in the context of class imbalance, and takes in count the positive classification rate and negative classification rate separately. The *AUC* can be seen as [42] and [43]:

$$AUC = \frac{TP_r+TN_r}{2} \tag{5}$$

6 Experimental results and discussion

In pattern recognition it is of great importance to recognize the minority class. However, this situation is difficult to achieve with imbalanced data sets. In this case, the classifiers tend to bias their learning to the majority class. The purpose of this paper is to analyze the balanced recognition between the *TP_r* and *TN_r* rates without degrading the accuracy of majority class. It is important to mention that this recognition is performed with imbalance data sets. In addition, the balanced recognition is considered when there is a difference of 20% between the accuracy of classes (majority and minority).

In the first section the experimental results without considering a previous preprocessing in the imbalanced data sets are showed. After that, the results obtained with preprocessing methods such as undersampling and oversampling are exhibited. In addition, all results presented in the tables exhibit the average accuracy of the five partitions obtained from each data set taking into account the cross validation method. Finally, the best results obtained by clasifiers are underlined and the relative recognition between the classes is indicated in bold.

6.1 Experimental results without preprocessing

This section exhibits the experimental results without a preprocessing in the imbalanced data sets. The values of true positive and true negative rates are shown in Table 2. After that, the results obtained with the AUC and MG measures are represented in Table 3.

Table 2 Experimental results without preprocessing: using the TP_r and TN_r rates

Data sets	CHAT		BN		MLP		RBF	
	TP_r	TN_r	TP_r	TN_r	TP_r	TN_r	TP_r	TN_r
Wisconsin	**98.32**	**97.07**	**97.92**	**96.84**	**94.58**	**96.64**	**97.90**	**94.82**
Haberman	**59.26**	**66.22**	17.52	93.32	28.20	88.00	15.98	94.24
Vehicle1	**57.98**	**69.31**	**62.16**	**73.44**	65.00	88.40	46.84	87.28
Glass0123_456	**94.00**	**91.38**	**80.18**	**96.34**	**87.74**	**96.32**	**84.36**	**94.46**
Vehicle3	**60.33**	**69.87**	**63.64**	**71.62**	58.94	89.58	41.92	85.34
Ecoli1	**94.83**	**79.88**	**83.16**	**86.86**	**76.68**	**94.98**	**91.02**	**85.68**
Glass6	**96.67**	**82.16**	**86.66**	**95.68**	72.00	97.84	**78.66**	**96.22**
yeast0256_3789	**77.68**	**62.10**	54.36	95.80	49.42	97.34	37.32	98.00
Glass-04_5	**100.00**	**81.62**	**100.00**	**98.82**	**100.00**	**100.00**	**90.00**	**98.82**
Shuttle-c0_c4	**99.20**	**83.18**	**100.00**	**100.00**	**99.20**	**100.00**	**98.40**	**99.82**
Glass4	**90.00**	**75.13**	33.32	96.52	76.68	98.00	**76.68**	**96.50**
Yeast1458_7	**66.67**	**52.63**	0.00	100.00	3.34	99.40	0.00	100.00
Yeast-2_8	**70.00**	**84.65**	55.00	99.78	55.00	99.12	60.00	99.56

From these experiments, firstly the results obtained with the *CHAT* model show a balance between the rates accuracy (*TP_r* and *TN_r*) in all data sets. In this case the class recognition is made without sacrificing the accuracy of the majority class. That situation cannot be observed through all the three neural networks. For example, the Bayesian network reports a balanced recognition among the classes of 61.54% (in eight data sets). And the *MLP* and *RBF* networks exhibited a balance of 38.46% (in five data sets) and of 53.85% (in seven data sets) on the balanced recognition. About the neural networks is possible to observe that the BN network shows a better performance with respect of the another two neural models (MLP and RBF).

Table 3 Experimental results without preprocessing: using the AUC and MG

Data sets	CHAT		BN		MLP		RBF	
	AUC	MG	AUC	MG	AUC	MG	AUC	MG
Wisconsin	**97.70**	**97.70**	**97.38**	**97.38**	**95.61**	**95.60**	**96.36**	**96.35**
Haberman	**62.74**	**62.65**	55.42	40.43	58.106	49.82	55.11	38.81
Vehicle1	**63.65**	**63.39**	**67.80**	**67.57**	76.70	75.80	67.06	63.94
Glass0123_456	**92.69**	**92.68**	**88.26**	**87.89**	**92.03**	**91.93**	**89.41**	**89.27**
Vehicle3	**65.10**	**64.93**	**67.63**	**67.51**	74.26	72.66	63.63	59.81
Ecoli1	**87.36**	**87.04**	**85.01**	**84.99**	**85.83**	**85.34**	**88.35**	**88.31**
Glass6	**89.41**	**89.12**	**91.17**	**91.06**	84.92	83.93	**87.44**	**87.00**
yeast0256_3789	**69.89**	**69.46**	75.08	72.16	73.38	69.36	67.66	60.48
Glass04_5	**90.81**	**90.34**	**99.41**	**99.41**	**100.00**	**100.00**	**94.41**	**94.31**
Shuttle-c0_c4	**91.19**	**90.84**	**100.00**	**100.00**	**99.60**	**99.60**	**99.11**	**99.11**
Glass4	**82.57**	**82.23**	64.92	56.71	87.34	86.69	**86.59**	**86.02**
Yeast1458_7	**59.65**	**59.24**	50.00	0.00	51.37	18.22	50.00	0.00
Yeast2_8	**77.32**	**76.98**	77.39	74.08	77.06	73.83	79.78	77.29

Table 3 shows the values obtained with the *AUC* and *MG* measures. From these result it is possible to observe that the *CHAT* model is the most benefited when there is a balanced recognition in four data sets. In this way, the neural networks can exhibit maximum benefit in two data sets. For example, the *BN* network presents its best performance with Glass6 and Shuttle-c0_vs_c4 data sets. However, the *MLP* and *RBF* networks exhibit their best behaviour in one dataset.

6.2 Experimental results using oversampling and undersampling methods

The experimental results obtained with the preprocessing methods are presented in this section; in specific the undersampling (Wilson) and oversampling (Smote) techniques are used. Firstly, the results obtained with the Wilson method are shown in Tables 4 and 5. After that, the experiments obtained through the Smote method are shown in Tables 6 and 7.

Table 4 Experimental results using undersampling method (TP_r and PF_r)

Data sets	CHAT		BN		MLP		RBF	
	TP_r	TN_r	TP_r	TN_r	TP_r	TN_r	TP_r	TN_r
Wisconsin	**98.32**	**96.85**	**99.16**	**96.84**	**96.24**	**96.62**	**98.32**	**95.06**
Haberman	**61.76**	**71.11**	20.02	91.54	21.14	90.24	25.96	92.9
Vehicle1	**60.30**	**68.52**	**58.46**	**74.72**	48.84	91.72	43.72	84.56
Glass0123_456	**96.00**	**91.99**	76.36	96.94	**84.54**	**95.12**	**84.36**	**96.94**
Vehicle3	**60.81**	**68.45**	51.34	78.24	33.48	93.54	26.00	90.86
Ecoli1	**97.42**	**74.86**	**85.66**	**86.06**	71.52	94.56	**90.92**	**85.68**
Glass6	**96.67**	**81.62**	76.68	99.46	76.68	97.84	62.00	98.92
yeast0256_3789	**78.74**	**59.12**	57.36	96.58	48.42	98.00	35.36	98.22
Glass-04_5	**100.00**	**73.31**	**100.00**	**98.82**	**100.00**	**100.00**	50.00	100.00
Shuttle-c0_c4	**99.20**	**84.29**	**100.00**	**100.00**	**99.20**	**100.00**	**98.40**	**99.94**
Glass4	**90.00**	**74.15**	29.98	95.02	40.02	98.00	20.00	99.00
Yeast1458_7	**66.67**	**47.50**	0.00	100.00	0.00	100.00	0.00	100.00
Yeast2_8	**55.00**	**99.78**	55.00	99.78	55.00	99.78	55.00	99.56

Table 4 shows the results by class, that is to say, the accuracy of each class is represented separately. In this way, the *CHAT* model keeps balanced recognition in the thirteen data sets when the undersampling technique called Wilson is used. However, this situation is not presented with results obtained with the neural networks; the *BN*, *MLP* and *RBF* networks show a balanced recognition of 38.46% (in five data sets) and 30.77% (in four data sets). It is important to mention that the blanced recognition is desirable because the classifier ensures an adequate recognition in both classes (minority and majority).

Table 5 shows the *AUC* and *MG* values considering a previous preprocessing with Wilson, this is obtained in terms of accuracy by class (balanced). From these experiments it is possible to observe that the *HACT* model shows its best performance on six (using *AUC*) and seven (Using *MG*) data sets in comparison with the other classifiers. This situation cannot be observed with the results obtained without a previous preprocessing. In this way, it is possible to say that the better results are obtained when using the Wilson method.

Tables 6 and 7 show the experimental results taking into account a previous preprocessing in the data sets. For this, the oversampling method was used, specifically the technique called Smote. This method increased the samples of the minority class until they obtained a balance between two classes.

Table 5 Experimental results using undersampling method (AUC and MG)

Data sets	CHAT		BN		MLP		RBF	
	AUC	MG	AUC	MG	AUC	MG	AUC	MG
Wisconsin	**97.59**	**97.58**	**98.00**	**97.99**	**96.43**	**96.43**	**96.69**	**96.68**
Haberman	**66.44**	**66.27**	55.78	42.81	55.69	43.68	59.43	49.11
Vehicle1	**64.41**	**64.28**	**66.59**	**66.09**	70.28	66.93	64.14	60.80
Glass0123_456	**93.99**	**93.97**	86.65	86.04	**89.83**	**89.67**	**90.65**	**90.43**
Vehicle3	**64.63**	**64.52**	64.79	63.38	63.51	55.96	58.43	48.60
Ecoli1	**86.14**	**85.40**	**85.86**	**85.86**	83.04	82.24	**88.30**	**88.26**
Glass6	**89.14**	**88.83**	88.07	87.33	87.26	86.62	80.46	78.31
yeast0256_3789	**68.93**	**68.22**	76.97	74.43	73.21	68.89	66.79	58.93
Glass04_5	**86.65**	**85.62**	**99.41**	**99.41**	**100.00**	**100.00**	75.00	70.71
Shuttle-c0_c4	**91.75**	**91.44**	**100.00**	**100.00**	**99.60**	**99.60**	**99.17**	**99.17**
Glass4	**82.07**	**81.69**	62.50	53.37	69.01	62.63	59.50	44.50
Yeast1458_7	**57.08**	**56.27**	50.00	0.00	50.00	0.00	50.00	0.00
Yeast2_8	**77.39**	**74.08**	77.39	74.08	77.39	74.08	77.28	74.00

Table 6 Experimental results of TP_r and PF_r (OverSampling)

Data sets	CHAT		BN		MLP		RBF	
	TP_r	TN_r	TP_r	TN_r	TP_r	TN_r	TP_r	TN_r
Wisconsin	**98.32**	**97.07**	**97.92**	**96.84**	94.58	4.83	**97.90**	**94.82**
Haberman	**55.59**	**70.22**	**56.94**	**70.66**	38.10	82.68	34.58	85.32
Vehicle1	**57.07**	**69.63**	**63.08**	**74.40**	**66.26**	**84.88**	**66.76**	**71.40**
Glass0123_456	**80.36**	**93.22**	**86.18**	**95.10**	**88.18**	**95.72**	**96.00**	**94.46**
Vehicle3	**59.38**	**70.82**	**65.48**	**70.64**	**69.76**	**85.46**	**76.78**	**67.98**
Ecoli1	**89.58**	**85.29**	**84.42**	**86.08**	**88.34**	**90.30**	**93.68**	**83.36**
Glass6	**86.67**	**90.27**	**89.98**	**96.76**	**81.98**	**96.22**	**82.00**	**95.14**
yeast0256_3789	**69.53**	**84.86**	50.26	93.82	64.64	87.74	61.52	92.16
Glass04_5	60.00	92.57	**100.00**	**100.00**	**100.00**	**100.00**	**80.00**	**100.00**
Shuttle-c0_c4	69.17	99.59	**100.00**	**100.00**	**99.60**	**100.00**	53.94	99.88
Glass4	**90.00**	**83.59**	**83.34**	**97.50**	**90.00**	**94.02**	**83.34**	**97.00**
Yeast1458_7	**60.00**	**69.53**	3.34	96.82	40.00	66.40	**56.66**	**57.28**
Yeast2_8	55.00	99.78	35.00	99.14	60.00	93.94	60.00	92.22

The balanced recognition between *TP_r* and *TF_r* rates was not performed fully with the *CHAT* model and the three neural networks. In this way, the maximum balanced recognition was reached with the *CHAT* model and *BN* network on ten data sets (76.92% in all cases). However, the experiments presented in Table 6 show that the *MPL* and *RBF* networks obtain a balanced recognition over eight (61.54% of all cases) and nine (69.23% of all cases) data sets. Despite this, the best balanced recognition between the classes is obtained using the Smote method. This situation cannot be observed with the results obtained without a previous preprocessing or considering the Wilson method.

In the context of a balance recognition between the accuracy of classes, the *BN* and *MLP* neural networks demonstrate a better classification performance on four and five data sets when a preprocessing (Smote) in the data sets is performed. This situation cannot be observed with the experiments obtained without a previous preprocessing or taking into account the Wilson method.

In addition, the experimental results obtained with a previous preprocessing (Wilson and Smote) show a better

Table 7 Experimental results using the AUC and MG (OverSampling)

Data sets	CHAT		BN		MLP		RBF	
	AUC	MG	AUC	MG	AUC	MG	AUC	MG
Wisconsin	**97.70**	**97.70**	**97.38**	**97.38**	49.71	21.38	**96.36**	**96.35**
Haberman	**62.91**	**62.48**	**63.80**	**63.43**	60.39	56.13	59.95	54.32
Vehicle1	**63.35**	**63.04**	**68.74**	**68.51**	**75.57**	**74.99**	**69.08**	**69.04**
Glass0123_456	**86.79**	**86.55**	**90.64**	**90.53**	**91.95**	**91.87**	**95.23**	**95.23**
Vehicle3	**65.10**	**64.85**	**68.06**	**68.01**	**77.61**	**77.21**	**72.38**	**72.25**
Ecoli1	**87.44**	**87.41**	**85.25**	**85.25**	**89.32**	**89.31**	**88.52**	**88.37**
Glass6	**88.47**	**88.45**	**93.37**	**93.31**	**89.10**	**88.82**	**88.57**	**88.33**
yeast0256_3789	**77.19**	**76.81**	72.04	68.67	76.19	75.31	76.84	75.30
Glass04_5	76.29	74.53	**100.00**	**100.00**	**100.00**	**100.00**	**90.00**	**89.44**
Shuttle-c0_c4	84.38	83.00	**100.00**	**100.00**	**99.20**	**100.00**	76.91	99.60
Glass4	**86.79**	**86.73**	**90.42**	**90.14**	**92.01**	**91.99**	**90.17**	**89.91**
Yeast1458_7	**64.77**	**64.59**	50.08	17.98	53.20	51.54	**56.97**	**56.97**
Yeast2_8	77.39	74.08	67.07	58.91	76.97	75.08	76.11	74.39

benefit in the classifiers performance when there is a balance between the accuracy of each class.

Figure 1 shows the original data set size, as well as sizes of the data sets after performing a previous preprocessing. The axis x corresponds to the data sets, while that the axis y indicates the data sets size. In this Figure some aspects stand out. Firstly, the samples number is increased with the Smote method until it obtains a balanced in the data set. However, with the Wilson method the samples in the data sets are decreased. In this sense, with Wilson method is observed on six data sets (46.15 %) a reduction in the data sets size with respect to Smote; the elimination of the samples is exhibited between a percentage range of 49.62% until 58.23 %. This situation is of great interest when it is observed that the best results in terms of accuracy by class (balanced) are obtained with the Wilson (HACT) and Smote (BN and MLP) methods.

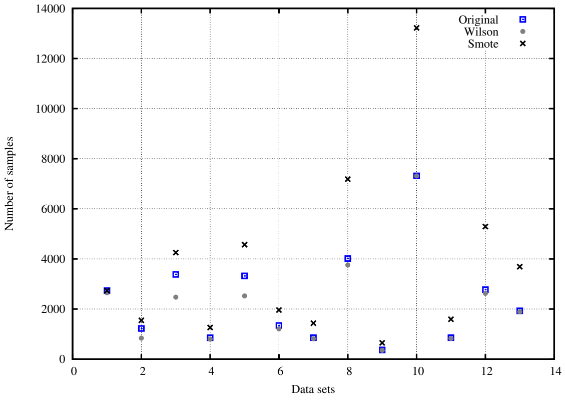

Fig. 1 Data set size ussing preprocessing Methods

The results show the convenience of use the undersampling method, because with small data sets is possible to obtain a low or high classification performance. A case interesting can be observed with the data set called shuttle-c0_c4; using 13220 samples with the Smote method in comparison with the Wilson method (7311 samples).

7 Conclusions

In this work was analyzed the behaviour of *HACT*, *BN*, *MLP* and *RBF* models in the context of a balanced recognition between the classes. The experiments were obtained with preprocessing and without preprocessing methods considering thirteen real-world data sets.

In terms of a balanced recognition, the four classifiers show a situation of great interest when is not considered a previous preprocessing in the imbalanced data sets. The classifiers recognize the minority class without sacrifice the accuracy of majority class in at least a data set. In this sense, the better classification performance in the context of a relative recognition is more emphasized with the *HACT* model in comparison with results obtained with the other classifiers when it is not considered a previous preprocessing.

In the domain of a balanced recognition, the results obtained with preprocessing methods demonstrated a better behavior in three classifiers such as *CHAT*, *BN* and *MLP*. With this is possible to conclude that the neuronal model needs a balanced recognition to obtain a good classification performance. In addition, it is convenient to use the Wilson method inasmuch as with fewer samples can obtain a good performance by class. In this sense, when the Wilson method is used, the HACT performance improves in comparison with results obtained with the Smote method and without a previous preprocessing. On the other hand, it was possible to observe that the BN and MLP networks performance in the context balanced recognition improve when Smote method is considered. This situation cannot be observed with the experiments obtained with the Wilson method and considering a previous preprocessing.

The open lines pointing out to study another classifiers and to deep in the imbalance study into associative memories context.

Acknowledgment

This work has been support by the Spanish Ministry of Education and Science under grant TIN2009-14205 and the Generalitat Valenciana under grant PROMETEO/2010/028 and Conacyt and PIFI and Secretaría de Investigación y Posgrado del Instituto Politécnico Nacional.

References

[1] M. Galar and A. Fernández and E. Barrenechea and H. Bustince, A Review on Ensembles for the Class

Imbalance Problems: Bagging, Boosting, and Hybrid-Based Approaches, IEE Transactions on Systems, Man, and Cybernetics-Part C:Applications and Reviews, **42**, 463-484 (2012).

[2] G. E. A. P. A. Batista and A. C. P. L. F. Carvalho and M. C. Monard, Applying One-Sided Selection to Unbalanced Datasets, Lecture Notes in Artificial Intelligence, **1793**, 315-325 (2000).

[3] N. Japkowicz and Shaju Stephen, The class imbalance problem: A systematic study, Intelligent Data Analysis, **6**, 429-449 (2002).

[4] R. C. Prati and G. E. A. P. A. Batista and M. C. Monard, Class Imbalances versus Class Overlapping:An Analysis of a Learning System Behavior, Lecture Notes in Computer Science, **2972**, 312-321 (2004).

[5] Tin Kam Ho and M. Basu, Complexity measures of supervised classification problems, IEEE Transactions on Pattern Analysis and Machine Intelligence, **24**, 289-300 (2002).

[6] G. E. A. P. A. Batista and R. C. Prati and M. C. Monard, Balancing Strategies and Class Overlapping, Proceedings of the 6th international conference on Advances in Intelligent Data Analysis, Springer-Verlag, 24-35 (2005).

[7] Mohamad H. Hasson, Fundamentals of Artificial Neural Networks, The MTI Massachusetts, (1995).

[8] Fabricio A. Breve and Moacir P. Ponti-Junior and Nelson D. A. Mascarenhas, Multilayer Perceptron Classifier Combination for Identification of Materials on Noisy-Soil Science Multispectral Images, Proceedings of the XX Brazilian Symposium on Computer Graphics and Image Processing, IEEE Computer Society, 239-244 (2007).

[9] Guohai Liu and Xiahong Xiao and Congli Mei and Yuhan Ding, A review of learning algorithm for radius basis function neural network, Control and Decision Conference (CCDC), 1112-1117 (2012).

[10] Nir Friedman and Michal Linial and Iftach Nachman, Using Bayesian networks to analyze expression data, Journal of Computational Biology, **7**, 601-620 (2000).

[11] R. Santiago Montero, Clasificador Hbrido de Patrones basado en la Lernmatrix de Steinbuch y el Linear Associator de Anderson-Kohonen, Tesis de Maestra en Ciencias de la Computacin CIC-IPN, (2003).

[12] Xiulan Hao and Chenghong Zhang and Hexiang Xu and Xiaopeng Tao and Shuyun Wang and Yunfa Hu, An Improved Condensing Algorithm, ACIS-ICIS, 316-321 (2008).

[13] I. B. V. da Silva and P. J. L. Adeodato, PCA and Gaussian noise in MLP neural network training improve generalization in problems with small and unbalanced data sets, The International Joint Conference on Neural Networks (IJCNN), 2664-2669 (2011).

[14] Junjie Hu, Active learning for imbalance problem using L-GEM of RBFNN, International Conference on Machine Learning and Cybernetics (ICMLC), 490-495 (2012).

[15] L. Cleofas Sánchez and M. Guzmán Escobedo and R.M Valdovinos Rosas and C. Yáez Márquez and O. Camacho Nieto, Using Hybrid associative classifier with traslation (HACT) for studying imbalance data sets, Revista Ingeniería e Investigación, 53-57 (2012).

[16] M. Aldape-Pérez and C. Yaez-Márquez and L.O. López Leyva, Feature Selection Using a Hybrid Associative Classifier with Masking Techniques, MICAI0́6. Fifth Mexican International on Artificial Intelligence, 151-160 (2006).

[17] M. AldapePérez, Implementación de los Modelos ALFA-BETA con Lógica Reconfigurable, Tesis de Maestra en Ciencias en Ingeniera de Cmputo con especialidad en sistemas digitales CICIPN, (2007).

[18] Bo Chen and Qin Liao and Zhonghua Tang, A Clustering Based Bayesian Network Classifier, Proceedings of the Fourth International Conference on Fuzzy Systems and Knowledge Discovery, 444-448 (2007).

[19] G. Quer and H. Meenakshisundaram and B. Tamma and B. S. Manoj and R. Rao and M. Zorzi, Research of Student Model Based on Bayesian Network, Global Telecommunications Conference, 1-6 (2010).

[20] Jingxin Guo and Qiaoyan Wen, A service oriented Bayesian network architecture, Broadband Network and Multimedia Technology, 452-455 (2011).

[21] Xin Wang and Peng Guo, A novel binary adaptive differential evolution algorithm for Bayesian Network learning, ICNC, 608-612 (2012).

[22] Nir. Friedman and Dan. Geiger and Moises Goldszmidt, Bayesian Network Classifiers, Machine Learning, 131-163 (1997).

[23] Qing Yang and Xiuping Wang and Zhufeng Huang and Shijue Zheng, Research of Student Model Based on Bayesian Network, Information Technologies and Applications in Education. ISITAE, 514-519 (2007).

[24] Hwang Ju-Won and Lee Young-Seol and Cho Sung-Bae, Structure evolution of dynamic Bayesian network for traffic accident detection, Evolutionary Computation (CEC), 1655-1671 (2011).

[25] Bo Chen and Qin Liao and Zhonghua Tang, A Clustering Based Bayesian Network Classifier, Proceedings of the Fourth International Conference on Fuzzy Systems and Knowledge Discovery, 444-448 (2007).

[26] Alan F. Murray, Applications of Neural Networks, Kluwer Academic Publishers, (1995).

[27] I. Bernard, Multilayer perceptron and uppercase handwritten characters recognition, Proceedings of the Second International Conference on Document Analysis and Recognition, 935-938 (1993).

[28] Robert A. Dunne, A Statistical Approach to Neural Networks for Pattern Recognition, WILEY-INTERSCIENCE, (2007).

[29] C. H. Chen and P. S. P Wang, Handbook of Pattern Recognition and Computer Vision, World Scientific, (2005).

[30] Christopher M. Bishop, Neural Networks for Pattern Recognition, Oxford, (1995).

[31] Zhang Xiao Ming and Ning Guang Liang, An Improved RBF Network On-Line Learning Algorithm, Proceedings of the Second International Symposium on Information Science and Engineering, 547-552 (2009).

[32] Zhang Xiao Ming and Ning Guang Liang, An Improved RBF Network On-Line Learning Algorithm, Proceedings of the Second International Symposium on Information Science and Engineering, 547-552 (2009).

[33] A. Fernández and S. García and F. Herrera, Addressing the classification with imbalanced data: open problems and new challenges on class distribution, HAIS’11, 1-10 (2011).

[34] Iaki Albisua and Olatz Arbelaitz and Ibai Gurrutxaga and Javier Muguerza and Jesús M. Pérez, C4.5 Consolidation

Process: An Alternative to Intelligent Oversampling Methods in Class Imbalance Problems, CAEPIA'11, 74-83 (2011).
[35] Sheng Tang and Si-Ping Chen, The generation mechanism of synthetic minority class examples, Information Technology and Applications in Biomedicine, 444-447 (2008).
[36] Piyasak Jeatrakul and Kok Wai Wong and Chun Che Fung, Classification of imbalanced data by combining the complementary neural network and SMOTE algorithm, Proceedings of the 17th international conference on Neural information processing: models and applications, 152-159 (2010).
[37] Xinjian Guo and Yilong Yin and Cailing Dong and Goping Yang and Guangtong Zhou, On the Class Imbalance Problem, Fourth International Conference on Natural Computation, 192-201 (2008).
[38] Ricardo Barandela and Rosa M. Valdovinos and J. Salvador Sánchez and Francesc J. Ferri, The Imbalanced Training Sample Problem: Under or over Sampling?, Lecture Notes in Computer Science, **3138**, 806-814 (2004).
[39] D.L. Wilson, Asymptotic properties of nearest neighbor rules using edited data, IEEE Transactions on Systems (Man and Cybernetics), **2**, 408-421 (1972).
[40] Rukshan Batuwita and Vasile Palade, A New Performance Measure for Class Imbalance Learning. Application to Bioinformatics Problems, Proceedings of the International Conference on Machine Learning and Applications, IEEE Computer Society, 545-550 (2009).
[41] Rukshan Batuwita and Vasile Palade, A New Performance Measure for Class Imbalance Learning. Application to Bioinformatics Problems, Proceedings of the International Conference on Machine Learning and Applications, IEEE Computer Society, 545-550 (2009).
[42] Yixin Cai and Mo-Yuen Chow and Wenbin Lu and Lexin Li, Evaluation of distribution fault diagnosis algorithms using ROC curves, Power and Energy Society General Meeting, 1-6 (2010).
[43] Ronaldo C. Prati and Gustavo E. A. P. A. Batista and Mara Carolina Monard, Class Imbalances versus Class Overlapping:An Analysis of a Learning System Behavior, Lecture Notes in Computer Science, **2972**, 312-321 (2004).

L. Cleofas is currently a PhD student at the Instituto Politécnico Nacional, Centro de Investigación en Computación (México). She received a BSc in Computer Science from the Universidad Autónoma del Estado de México (México), in 2004 and a MSc in Computer Science from the Instituto Tecnológico de Toluca in 2009. Hers main research interests are pattern recognition; in specific classification supervised, preprocessing method, genetic algorithms among others.

O. Camacho is currently the director from Centro de Innovación y Desarrollo Tecnológico en cómputo. He received a BSC in Communications and Electronics Engineering from the Instituto Politécnico Nacional, ESIME-ZAC in 1989 and a MSc in Computer Engineering from the Instituto Politécnico Nacional, Centro de Investigación Tecnológica en Computación del IPN in 1995 and PhD in Computer Architecture from Universidad Politécnica de Cataluña, Barcelona-España in 1999. He is the author or coauthor of some scientic publications. He is member of the Sistema Nacional de Investigadores (México). His main researches interests are pattern recognition, Associative Memories among others.

J. S. Sánchez is a Full Professor in the Department of Computer Languages and Systems at Universitat Jaume I (Castelló de la Plana, Spain) since 1992. He received a BSc in Computer Science from the Universidad Politécnica de Valencia in 1990 and a Ph.D. in Computer Science Engineering from Universitat Jaume I in 1998. He has authored more than 150 scientic publications, and is co-editor of two books and guest editor of several special issues in international journals. He is a Senior Member of IEEE and IAPR. He serves as an Associate Editor for the Pattern Analysis and Applications Journal. His current research interests lie in the areas of pattern recognition and machine learning, including non-parametric classification, feature and prototype selection, ensembles of classifiers, and clustering.

C. Yáñez is an Associate Professor from Instituto Politécnico Nacional, Centro de Investigación en Computación (México). He received a BSc in Physics and Mathematics from the Instituto Politécnico Nacional, school Physics and Mathematics in 1989 and a MSc in Computer Engineering (Digital Systems) from the Instituto Politécnico Nacional, Centro de Investigación Tecnológica en Computación from the IPN in 1995 and PhD in Computer Science from Instituto Politécnico Nacional, Centro de Investigación en Computación (México) in 2002. He is the author or coauthor of some scientic publications. He is member of the Sistema Nacional de Investigadores (SNI II, México). His main researches interests are pattern recognition, Associative Memories, Mathematical Morphology and Soft Computing.

R. M. Valdovinos is a Professor from the Universidad Autónoma del Estado de México (Engineering school), PhD in Advanced Informatic Systems from the Universitat Jaume I (in 2006, Spain) and from the Instituto Técnologico de Toluca (in 2010, México). She was member of the Sistema Nacional de Investigadores (in 2008-2010 México) and has the desirable profile of PROMEP. Member of the Evaluators Accredited of the CONACYT, of the International Association of Pattern Recognition, of the Mexican Association of Science and Technology for Development, of the Mexican Association of Artificial Intelligence and of the Mexican Network of Research and Development in Computation. She is regulator member of the Network of Artificial Intelligence (UAEM-ITT-UJI). Her main researches interests are Multiple Classifier Systems, Genetic Algorithms, Artificial Neural Networks and applications of pattern recognition and data mining for resolving real-problems.