Gamma classifier based instance selection

[1]Jarvin A. Antón Vargas, [1]Yenny Villuendas-Rey, [2]Itzamá López-Yáñez

[1]Departamento de Ciencias Informáticas, Universidad de Ciego de Ávila, Cuba
[2] Centro de Innovación y Desarrollo Tecnológico en Cómputo, Instituto Politécnico Nacional, México
janton@unica.cu; yenny@cav.uci.cu; itzama@gmail.com

Abstract. Filtering noisy and mislabeled instances is an important problem in Pattern Recognition. This paper introduces a novel similarity function based on the Gamma operator, and use it for instance selection for improving the Gamma classifier. Numerical experiments over repository databases show the high quality performance of the proposal according to classifier accuracy and instance retention ratio.

Keywords: Gamma Classifier, Instance Selection, Supervised Classification.

1 Introduction

The training dataset plays a key role for supervised classification. Training data allows building classifiers able to estimate the label or class of a new unseeing instance. Several researchers have pointed out that if the dataset has high quality instances, the classifier can produce predictions that are more accurate [1]. However, in several real-world applications, it is not possible to obtain a training set without noising and mislabeled instances. To overcome this problem, several instance selection and generation algorithms have been proposed [1, 2].

The Gamma classifier [3, 4] is a recently proposed supervised classifier, ant it has been applied successfully to several prediction tasks, such as air quality [5], pollutant time series [6] and development effort prediction of software projects [7]. Despite of the excellent performance of the Gamma classifier, it is noted that it is affected by noisy or mislabeled instances.

However, most of the instance selection algorithms are designed for the Nearest Neighbor (NN) classifier [8]. Little work has been done for selecting instance to other supervised classifiers, such as ALVOT [9, 10] and Neural Networks [11], and there are no directly applicable to the Gamma classifier.

This paper proposes a similarity function based on the Gamma operator of the Gamma classifier, and use it for similarity comparisons in NN instance selection algorithms. The

thorough experimental study carried out shows the significant performance gains of the proposed approach.

2 Gamma Classifier and Gamma Based Similarity

The Gamma classifier is based on two operators named Alpha and Beta, which are the foundation of the Alpha-Beta associative memories [12]. The Alpha and Beta operators are defined in a tabular form considering the sets $A = \{0, 1\}$ and $B = \{0, 1, 2\}$, as shown in figure 1.

$\alpha : A \times A \rightarrow B$		
x	y	$\alpha(x, y)$
0	0	1
0	1	0
1	0	2
1	1	1

$\beta : B \times A \rightarrow A$		
x	y	$\beta(x, y)$
0	0	0
0	1	0
1	0	0
1	1	1
2	0	1
2	1	1

Fig. 1. Operators Alpha and Beta.

In addition to the Alpha and Beta operator, the Gamma classifier also uses two other operators: the u_β operator and the generalized gamma similarity operator, γ_g. The unary operator u_β receives as an input a binary n-dimensional vector, and returns a number $\mathrm{p} \in \mathbb{Z}^+$ according to the following expression:

$$u_\beta = \sum_{i=1}^{n} \beta\,(x_i, x_i) \tag{1}$$

The generalized gamma similarity operator receives as input two binary vectors x and y and also a non-negative integer θ, and returns a binary digit, as follows:

$$\gamma_g(x, y, \theta) = \begin{cases} 1 & if\ m - u_\beta[\alpha(x, y) mod 2] \leq \theta \\ 0 & otherwise \end{cases} \tag{2}$$

That is, the γ_g operator returns 1 if the input vectors differentiates at most in θ bits, and returns zero otherwise.

The Gamma classifier is designed for numeric patterns, and assumes that each pattern belongs to a single class. However, as the generalized gamma similarity operator receives as input two binary vectors, the Gamma classifier codifies numeric instances using a modified Johnson-Möbius code [3]. In figure 2 we show a simplified schema of the Gamma classifier.

According to the classification strategy of the Gamma classifier, we propose a similarity function to compare pairs of instances, regarding the θ parameter. This allows us to detect noisy or mislabeled instances.

The proposed Gamma based similarity (GBS) uses the generalized gamma operator, but it considers the standard deviation of the feature instead of the θ parameter. Let be X and Y to instances, the Gamma based similarity between them is computed as:

$$GBS(X,Y) = \sum_{i=1}^{p} \gamma_g\,(x_i, y_i, \sigma_i) \tag{3}$$

where p is the amount of features describing the instances, σ_i is the standard deviation of the i-th feature, and x_i and y_i are the binary vectors associated with the i-th feature in instances X and Y, respectively.

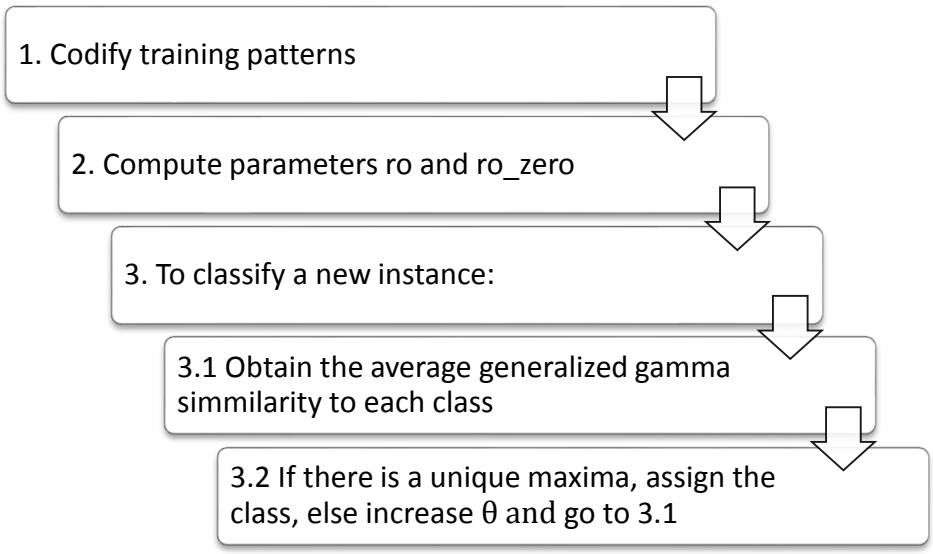

Fig. 2. Simplified schema of the classification process with the Gamma classifier.

Considering this novel similarity, we are able to apply several instance selection algorithms which were designed for the Nearest Neighbor classifier, and test its performance in the filtering of noisy and mislabeled instances for the Gamma classifier.

3 Experimental Results

We select some of the most representative instance selection algorithms and perform the test over seven databases from the Machine Learning repository of the University of California at Irvine [13]. Table 1 shows the characteristics of the selected databases.

We selected error-based editing methods due to their ability of smoothing decision boundaries and to improve classifier accuracy. The selected methods are the Edited Nearest Neighbor (ENN) proposed by Wilson [14], the Gabriel Graph Editing method (GGE) proposed by Toussaint [15] and the MSEditB method, proposed by García-Borroto et al. [16].

The ENN algorithm (Edited Nearest Neighbor) is the first error-based editing method reported [14]. It was proposed by Wilson in 1972 and it consist on the elimination of the objects misclassified by a 3-NN classifier. The ENN works by lots, because it flags the misclassified instances and then simultaneously deltetes them all, which guaranteed order independence. The ENN has been extensively used in experimental comparisons, showing very good performance [1].

Table 1. Databases used in the experiments.

Databases	Objects	Attributes	Classes
balance-scale	625	4	3
breast-w	699	9	2
ecoli	336	7	8
heart-statlog	270	13	2
ionosphere	351	34	2
iris	150	4	3
vehicle	846	18	4

The GGE algorithm is based on the construction of a Gabriel graph. A Gabriel graph is a directed graph such that two instances $x \in U$ and $y \in U$ form an arc if and only if $\forall z \in U\ (d((x + y/2), z) > d(x, y)/2)$, where d is a dissimilarity function. That is, two instances x and y are related in a Gabriel graph if there is no object in the hypersphere centered in the middle point of x and y, and with radius the distance between x and y.

The GGE algorithm consist in deleting those instances connected to others of different class labels. It deletes borderline instances, and keep class representatives ones.

The MSEditB algorithm [16] uses a Maximum similarity graph to select the objects to delete. A Maximum similarity graph is a directed graph such that each instance is connected to its most similar instances. Formally, let be S a similarity function, an instance $x \in U$ form

an arc in a Maximum similarity graph with an instance $y \in U$ if and only if $d(x,y) = \max_{z \in U} d(x,z)$.

The MSEditB algorithm deletes an instance if it has a majority of its predecessors and successors instances not of its class.

All algorithms were implemented in C# language, and the experiments were carried out in a laptop with 3.0GB of RAM and Intel Core i5 processor with 2.67HZ. We cannot evaluate the computational time of the algorithms, because the computer was not exclusively dedicated to the execution of the experiments.

To compare the performance of the algorithms, it was used the classifier accuracy. The classifier accuracy is measure as the ratio of correctly classified instances. It was also computed the Instance retention ratio (IRR) for every algorithm, in order to determine the amount of selected instances. Table 2 and 4 show the results according to classifier accuracy and instance retention ratio, respectively.

In table 2, we show the accuracy of the Gamma classifier without selecting instances (Gamma) and the accuracy of the Gamma classifier trained using the instances selected by ENN, GGE and MSEditB, respectively.

Table 2. Accuracy of the gamma classifier before and after the selection of instances.

Databases	Gamma	Instances selected by		
		ENN	GGE	MSEEditB
balance-scale	0.838	0.598	0.810	**0.891**
breast-w	0.907	**0.908**	**0.908**	**0.908**
ecoli	**0.708**	0.560	0.474	0.536
heart-statlog	0.819	**0.833**	0.826	0.830
ionosphere	**0.749**	**0.749**	**0.749**	**0.749**
iris	**0.913**	0.813	0.907	0.867
vehicle	0.573	0.576	**0.582**	0.568

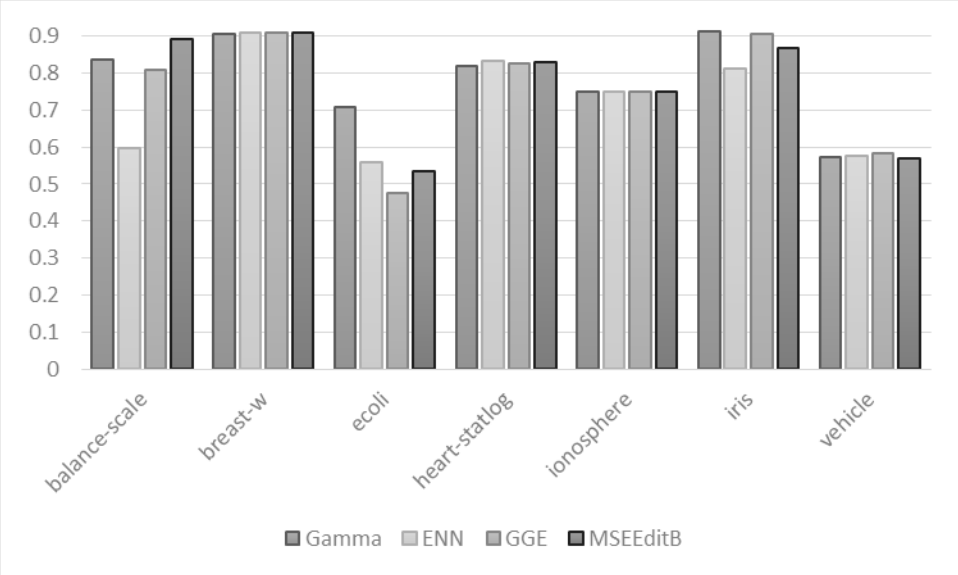

Fig. 3. Accuracy of the Gamma classifier using selected instances.

As shown, the instance selection algorithms were able to improve the Gamma classifier accuracy in four databases, and to obtain the same accuracy with fewer instances in one database. Still, for the ecoli and iris datasets, no improvement were obtained.

However, to determine the existence or not of significant differences in algorithm´s performance it was used the Wilcoxon test [17]. It was set as null hypothesis no difference in performance between the gamma classifier without instance selection (Gamma) and the gamma classifier with instance selection algorithms, and as alternative hypothesis that latter had better performance. It was set a significant value of 0.05, for a 95% of confidence. Table 3 summarizes the results of the Wilcoxon test, according to classifier accuracy.

Table 3. Wilcoxon test comparing classifier accuracy.

Original Gamma vs	**ENN**	**GGE**	**MSEditB**
wins-looses-ties	3-3-1	3-3-1	3-3-1
probability	0.345	0.600	0.735

The Wilcoxon test obtains probability values greater than the significance level, and thus, we do not reject the null hypothesis. These results confirm the proposed approach is able to preserve classifier accuracy, using a small amount of instances.

Table 4. Instance retention ratio obtained by the selection of instances.

Databases	ENN	GGE	MSEEditB
balance-scale	0.799	**0.723**	0.837
breast-w	0.973	0.968	**0.963**
ecoli	0.812	0.884	**0.792**
heart-statlog	0.889	**0.712**	0.807
ionosphere	0.915	**0.842**	0.854
iris	0.726	**0.689**	0.852
vehicle	0.659	0.751	**0.541**

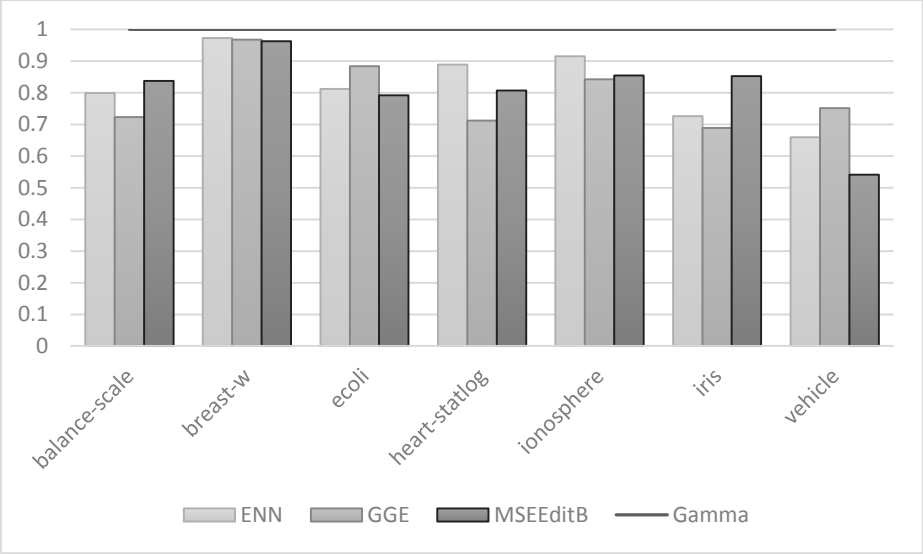

Fig. 4. Instance retention ratio obtained by the algorithms.

As shown in table 4, all instance selection methods are able to delete among the 40% and 4% of the data, without decreasing the classifier accuracy. These results confirm the proposed approach is able to obtain an adequate training set for the Gamma classifier, without losing representative objects.

Table 5. Wilcoxon test comparing instance retention ratio.

Original Gamma vs	**ENN**	**GGE**	**MSEditB**
wins-looses-ties	0-7-0	0-7-0	0-7-0
probability	**0.018**	**0.018**	**0.018**

According to instance retention ratio, the Wilcoxon test rejects the null hypothesis in all cases. That is, the number of selected objects using ENN, GGE and MSEditB with the proposed gamma based similarity function, was significantly lower than the original amount of instances in the training set.

The experimental results carried out show that selecting instances by using a similarity function based on the Gamma operator maintains classifier accuracy, and also reduces the cardinality of the training sets, diminishing the computational cost of the Gamma classifier.

4 Conclusions

Instance selection is an important preprocessing step for learning with most supervised classifiers. In this paper, a novel similarity measure is introduced, based on the Gamma operator of the Gamma classifier. We used the proposed similarity to select relevant instances for this classifier. Experimental results carried out over several repository data show that using the proposed similarity function for instance selection preserves classifier accuracy, and decreases the computational cost of the Gamma classifier.

References

1. S. García, J. Derrac, J. R. Cano, and F. Herrera, "Prototype Selection for Nearest Neighbor Classification: Taxonomy and Empirical Study," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, pp. 417-435, 2012.
2. I. Triguero, J. Derrac, S. Garcia, and F. Herrera, "A taxonomy and experimental study on prototype generation for nearest neighbor classification," Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 42, pp. 86-100, 2012.
3. I. López Yáñez. Clasificador automático de alto desempeño (MS dissertation, Instituto Politécnico Nacional-Centro de Investigación en Computación), 2007.
4. I. López-Yáñez, L. Sheremetov and C. Yáñez-Márquez, "A novel associative model for time series data mining," Pattern recognition Letters, vol 41, pp. 23-33, 2014.
5. C. Yánez-Márquez, I. López-Yánez and G.D. Morales, "Analysis and prediction of air quality data with the gamma classifier," Progress in Pattern Recognition, Image Analysis and Applications, pp. 651-658, 2008.
6. I. Lopez-Yanez, A.J. Argüelles-Cruz, O. Camacho-Nieto and C. Yanez-Marquez, "Pollutants time-series prediction using the Gamma classifier," International Journal of Computational Intelligence Systems, vol 4, no 4, pp. 680-711, 2012.
7. C. López-Martın, I. López-Yánez and C. Yánez-Márquez, "Application of Gamma Classifier to Development Effort Prediction of Software Projects," Appl. Math, vol 6 no 3, pp. 411-418, 2012.
8. T. M. Cover and P. E. Hart, "Nearest Neighbor pattern classification," IEEE Transactions on Information Theory, vol. 13, pp. 21-27, 1967.

9. M. A. Medina-Pérez, M. García-Borroto, Y. Villuendas-Rey and J. Ruiz-Shulcloper, “Selecting objects for ALVOT,” Progress in Pattern Recognition, Image Analysis and Applications, pp. 606-613, 2006.
10. M.A. Medina-Pérez, M. García-Borroto and J. Ruiz-Shulcloper, “Object selection based on subclass error correcting for ALVOT,” Progress in Pattern Recognition, Image Analysis and Applications, pp. 496-505, 2007.
11. H. Ishibuchi, T. Nakashima and M. Nii, “Learning of neural networks with GA-based instance selection,” IFSA World Congress and 20th NAFIPS International Conference, vol. 4, pp. 2102-2107, 2001.
12. C. Yáñez-Márquez and J. Luis Díaz de L., “Memorias Asociativas basadas en relaciones de orden y operaciones binarias,” Computación y Sistemas vol 6 no 4, pp. 300 - 311, ISSN 1405-5546, Centro de Investigación en Computación – Instituto Politécnico Nacional, México 2003.
13. A. Asuncion and D. Newman, UCI machine learning repository, 2007.
14. D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," IEEE Transactions on Systems, Man and Cybernetics, vol. SMC-2, pp. 408-421, 1972.
15. G. T. Toussaint, "Proximity Graphs for Nearest Neighbor Decision Rules: Recent Progress," in 34 Symposium on Computing and Statistics INTERFACE-2002, Montreal, Canada, 2002, pp. 1-20.
16. M. García-Borroto, Y. Villuendas-Rey, J. A. Carrasco-Ochoa, and J. F. Martinez Trinidad, "Using Maximum Similarity Graphs to edit nearest neighbor classifiers," Lecture Notes on Computer Science, vol. 5856, pp. 489-496, 2009.
17. J. Demsar, "Statistical comparison of classifiers over multiple datasets," The Journal of Machine Learning Research, vol. 7, pp. 1-30, 2006.