Using Rough Sets and Maximum Similarity Graphs for Nearest Prototype Classification

Yenny Villuendas-Rey[1,3], Yailé Caballero-Mota[2], and María Matilde García-Lorenzo[3]

[1] Computer Science Department, University of Ciego de Ávila, Carr. a Morón km 9 ½, Cuba
yennyv@informatica.unica.cu
[2] Computer Science Department, University of Camagüey, Circunv. Norte km 3 ½, Cuba
yaile.caballero@reduc.edu.cu
[3] Computer Science Department, University of Las Villas, Carr. a Camajuaní, km 5 ½, Cuba
mmgarcia@uclv.edu.cu

Abstract. The nearest neighbor rule (NN) is one of the most powerful yet simple non parametric classification techniques. However, it is time consuming and it is very sensitive to noisy as well as outlier objects. To solve these deficiencies several prototype selection methods have been proposed by the scientific community. In this paper, we propose a new editing and condensing method. Our method combines the Rough Set theory and the Compact Sets structuralizations to obtain a reduced prototype set. Numerical experiments over repository databases show the high quality performance of our method according to classifier accuracy.

Keywords: nearest neighbor, prototype selection, editing methods.

1 Introduction

The nearest neighbor (NN) classifier is one of the most popular supervised classification methods in Pattern Recognition. However, it suffers from two important drawbacks: (i) High storage and computational requirements: it storages the entire training set, requiring large space. Moreover, to determine the class of a new object, it needs to compare it with every prototype in the training set, and (ii) Sensitivity: the NN classification rule is quite sensitive to noisy and outlier objects.

To overcome these drawbacks researchers have proposed prototype selection methods. These methods aim to obtain a reduced set of representative objects (prototypes) to be used for classification. In the literature, these techniques are known as prototype selection and prototype generation methods. The former obtain a reduced set composed by objects in the original training set, and the later may create artificial prototypes, not present in the original data [1]. Another challenge with the NN classifiers is handling mixed as well as incomplete data. This kind of data affects most prototype generation methods, by lacking of the properties of vector spaces. Some prototype selection methods based on geometric properties of the data space are also inapplicable, because the objects are defined in a Cartesian space instead of a Metric space. Among prototype selection methods, there is a distinction between error based editing and condensing methods. Error based editing methods are focused on remove noisy as well as outlier objects, and aim at smoothing the class boundaries, improving

accuracy. On the contrary, condensing methods aim at reducing redundant objects, and to obtain training set as small as possible. Although they have a different goal, combining both techniques may result into a small training set, with high classifier accuracy. In this paper, we combine the Rough Set Theory and the Compact Set structuralization to achieve this objective, and also to handling mixed and incomplete data sets.

The paper is organized as follows: section two reviews some of the previous work done in prototype selection, section three introduces the proposed method, section four shows the results of the numerical experiments and section five offers some conclusions.

2   Previous Works on Prototype Selection

Since the introduction of Nearest Neighbor classifier, reducing the training set have been an active research area in the Pattern Recognition and Artificial Intelligence field. In this section we only address methods capable of handling mixed as well as incomplete data. The methods to reduce the training set can be divided in editing methods and condensing methods. Condensing methods aim at reducing redundant objects, and to obtain a training set as small as possible. Several condensing methods have been proposed, and among them we can mention the Minimal Consistent Set (MCS) [2], the Generalized Condensed Nearest Neighbor (GCNN) [3], the Prototype Relevance Selection (PRS) [4], the Gray Based Reduction (GBR1, GBR2) [5] and the CSESupport [6]. Recently, several algorithms have been proposed, such as Multiple Instance Learning [7] and Class Boundaries Preserving (CBP) [8].

Opposite to condensing methods, editing methods try to smooth class boundaries, by removing noisy or outlier objects. Usually, editing methods do not reduce a significant amount of objects, but in some cases may improve classifier accuracy. The first method to accomplish this task was the Edited Nearest Neighbor (ENN) by Wilson [9]. It removes every object misclassified by a k-NN classifier, with k value as used defined parameter. Among editing methods are NENN [10], editing based on Rough Sets (EditRS1, EditRS2) [11] and editing based on Maximum Similarity Graphs (MSEditA, MSEditB) [12].

Although editing and condensing methods achieve complementary objectives, the smoothing power of editing methods, and the high reduction rates of condensing algorithms have not been sufficiently exploited by combining both approaches into a single algorithm. The analysis of previous prototype selection methods reveals that improving NN in both accuracy and computational requirements by reducing the training set is still an open problem. Previous methods cannot accurately deal with both objectives, which is the basic motivation of the method introduced in this paper.

3   Editing Based on Rough Sets and Compact Sets

3.1   Rough Sets and Maximum Similarity Graphs

Rough Set Theory (RST) has been an excellent mathematical tool for data analysis and it has offered an interesting theoretic base for the solution of many problems

within knowledge discovery. Rough Sets Theory was proposed by Pawlak in 1982 [13]. In RST, a training set can be represented as a Decision System. First, an Information System is a pair S= (U, A), where U is a non-empty finite set of objects called the Universe and A is a non-empty finite set of attributes, and a Decision System is any information system of the form $DS = A \cup \{c\}$, where $c \notin A$ is the decision attribute. Classical definitions of lower and upper approximations were originally introduced with reference to an indiscernible relation which assumed to be an equivalence relation. Let $B \subseteq A$ and $X \subseteq U$. B defines an equivalence relation and X is a concept. X can be approximated using only the information contained in B by constructing the B-lower and B-upper approximations of X, denoted by B^*X and B_*X respectively. The objects in B_*X are sure members of X, while the objects in B^*X are possible members of X. Rough set model has several advantages to data analysis, especially to edit training sets. It is based on the original data only and does not need any external information; no assumptions about data are necessary, and it is suitable for analyzing both quantitative and qualitative features [11].

Maximum Similarity Graph (MSG) are a useful tool for structuralizing data in the Pattern Recognition field [14]. A Maximum Similarity Graph is a directed graph, such as it connects each object with its most similar neighbor. Formally, let be $G = (X, \theta)$ a MSG for a set of objects X, with arcs θ. Two objects $x_i, x_j \in X$ form an arc $(x_i, x_j) \in \theta$ if $\max_{x \in X}\{sim(x_i, x)\} = sim(x_i, x_j)$, where $sim(x_i, x_j)$ is a similarity function, usually $sim(x_i, x_j) = 1 - \Delta(x_i, x_j)$ and $\Delta(x_i, x_j)$ is a dissimilarity function. In case of ties, the maximum similarity graph establishes a connection between the object and each of its nearest neighbors (figure 1).
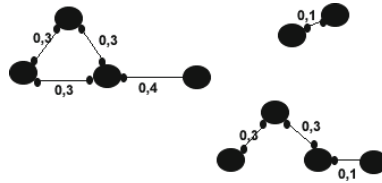

Fig. 1. Maximum similarity graph, using 1- Δ(.,.) as similarity function

Among the advantages of Maximum Similarity Graphs we can mention that they do not need parameters for their construction, except of the similarity function to compare two objects. In addition, the objects form arcs only to their most similar objects in the training matrix, which is valuable information particularly in high Bayes risk zones. The arcs between objects also contribute to predict the certainty of the correct classification of an object [12].

In a Maximum Similarity Graph, each connected component is called a Compact Set. In figure 2 we show the Compact Sets of figure 1. Formally, a subset $N \neq \emptyset$ of X is a compact group if and only if [14]:

$$a) \forall x_j \in X \left[x_i \in N \wedge \left(\begin{array}{c} \max\limits_{\substack{x_i \in X \\ x_i \neq O_j}}\{sim(x_i, x_j)\} = sim(x_i, x_j) \\ \vee \max\limits_{\substack{x_i \in X \\ x_i \neq x_j}}\{sim(x_j, x_i)\} = sim(x_j, x_i) \end{array} \right) \right] \Rightarrow x_j \in N$$

$$b)\forall x_i, x_j \in N, \exists x_{i_1}, \cdots, x_{i_q} \in N \begin{bmatrix} x_i = x_{i_1} \wedge x_j = x_{i_q} \wedge \forall p \{1, \cdots, q-1\} \\ \begin{bmatrix} \max_{\substack{x_t \in X \\ x_t \neq o_{i_p}}} \{\text{sim}(x_{i_p}, x_t)\} = \text{sim}(x_{i_p}, x_{i_{p+1}}) \\ \vee \max_{\substack{x_t \in X \\ x_t \neq x_{i_p}}} \{\text{sim}(x_{i_{p+1}}, x_t)\} = \text{sim}(x_{i_{p+1}}, x_{i_p}) \end{bmatrix} \end{bmatrix}$$

c) Every isolated object is a compact set, degenerated.
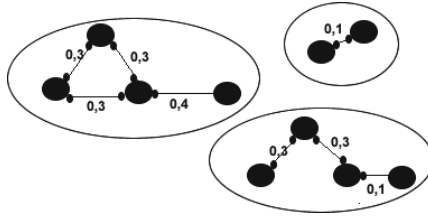

Fig. 2. Compact sets of the objects. Each ellipse corresponds to a compact set.

Compact Sets are formed by highly similar objects, and have been used successfully to edit nearest neighbor classifiers [12].

3.2 Hybridizing Rough Sets and Compact Sets

In order to consider the advantages of both Rough Sets and Compact Sets, we use the definition of Neighborhood Rough Sets introduced by Hu et al. [15]. Given an arbitrary object $x_i \in X$ and a set of attributes $B \subseteq A$, the neighborhood $\partial_B(x_i)$ of x_i in feature space B is defined as the set of objects which dissimilarity values with respect x_i, taking into consideration only the attributes in B, is lower than a threshold ∂.   Formally, $\partial_B(x_i) = \{x_j | x_j \in X, \Delta^B(x_i, x_j) \leq \partial\}$,   where Δ is   a dissimilarity function. As mention by Hu et al., a neighborhood granule degrades to an equivalent class if the threshold $\partial = 0$. In this case, the objects in the same neighborhood granule are equivalent to each other. Consequently, the neighborhood rough sets are a generalization of Pawlak´s rough sets. In our case, we want to obtain neighborhood granules without fixing the threshold ∂, so we redefine the neighborhood $\partial_B(x_i)$ of x_i in feature space B, as a Minimum Neighborhood Rough Set, taking as the ∂ value the minimum dissimilarity between x_i and every other object in the data.

$$\partial_B(x_i) = \left\{x_j \,\middle|\, x_j \in X, \Delta^B(x_i, x_j) = \min_{x_j \in X}(\Delta^B(x_i, x_j))\right\} \tag{1}$$

Given a Decision System $DS = A \cup \{C\}$, the lower approximation of the decision is defined as the union of the lower approximation of each decision class. The lower approximation of the decision is also called the positive region of the decision. The minimum neighborhood $\partial_B(x_i)$ of x_i in feature space B can be rewritten as:

$$\partial_B(x_i) = \{x_j | (x_i, x_j) \in \theta\} \tag{2}$$

where   are the arcs in a Maximum Similarity Graph, using as similarity function $1 - \Delta^B$. In a Minimum Neighborhood Rough Set, the positive region of a decision class, $POS_B(c)$ is composed by all objects connected in a Maximum Similarity Graph with objects of the same class. The positive region will be formed by objects which are sure members of one of the decision classes, and can be correctly classified by a NN classifier, while boundary region is the set of samples which may be misclassified by a NN classifier. Opposite, the objects in the boundary region are connected to objects from more than one decision class.

$$POS_B(c) = \left\{ x_i \middle| x_i \in X, \forall_{(x_i, x_j) \in \theta}, C(x_i) = C(x_j) = c \right\} \tag{3}$$

where $C(x_i)$ denotes the decision class of object x_i.

Our method (Figure 3) first computes the positive region of each decision class. Then, it computes compact sets of the positive region of each decision class, grouping the objects in subclasses according to its similarity (dissimilarity). Then, we condense each compact set to a single object, the representative prototype p. We select as prototype the object that maximizes the similarity with respect to other objects in the compact set CS, that is, the object for which the similarity value is the maximum:

$$p = \underset{o \in CS}{\operatorname{argmax}} \left\{ \sum_{i \in CS} sim(o, i) \right\} \tag{4}$$

Our approach gracefully handles mixed and incomplete data, because it is based on Maximum Similarity Graph computation and Neighborhood Rough Sets, which do not assume any metric or geometric property of data.

Prototype Editing based on Rough sets and Compact sets (PERC)
Inputs: Training set T, Attribute set B, Dissimilarity Δ. Outputs: Prototype set P

1. $P = \emptyset$
2. Obtain a Maximum Similarity Graph, $G = (T, \theta)$ of the objects in T
3. For each decision class c:
    3.1. Compute $POS_B(c) = \{x_i | x_i \in T, \forall \theta(x_i, x_j), C(x_i) = C(x_j) = c\}$
    3.2. Compute compact sets CS_c of $POS_B(c)$
        For each $cs \in CS_c$
        3.2.1. Select a representative prototype p using equation 4, $P = P \cup \{p\}$
4. Return P

Fig. 3. The PERC algorithm for prototype selection

Summarizing, our algorithm to edit and condense the training matrix obtains the positive region of each decision class, then computes compacts set of each positive region and returns as the edited set the set of selected representative prototypes, one of each compact set. Our approach takes advantages of the discriminative power of Rough Sets, as well as the structuralization abilities of Compact Sets. The PERC

algorithm smoothes the class boundaries by taking into consideration only the positive region of the decision, and obtains a highly condensed set, by taking representatives of the compact set structuralization of the positive region of each class. Our proposal also handles mixed and incomplete data sets.

4   Experimental Analysis

In order to compare the performance of the propose algorithm, we carried out some numerical experiments in a wide range of databases with both numerical and categorical attributes from the Machine Learning repository of the University of California at Irvine [16]. The description of used databases is given in table 1. We perform 10-fold cross validation and average the results in both classifier accuracy and object reduction rates. We use in our experiments as dissimilarity functions the HEOM and HVDM proposed by Wilson and Martínez [17].

We compare our method against 12 classical and state of the art editing and condensing prototype selection methods. We use the Wilcoxon test to establish the statistical significance of the differences in performance of the studied methods. We set as null hypothesis no difference in performance between each pair of methods, and set a significant value of 0.05, for a 95% of confidence. In table 2, we summarize the results of the Wilcoxon test, using both HEOM and HVDM dissimilarities, according to classifier error and object retention rates.

Table 1. Description of the databases used in numerical experiments

Databases	Attributes (Categorical -Numerical)	Obj.	Missing values	Databases	Attributes (Categorical -Numerical)	Obj.	Missing values
autos	10-16	205	x	heart-h	7-6	294	x
balance	0-4	625		hepatitis	13-6	155	x
breast-c	9-0	286	x	iris	0-4	150	
breast-w	0-9	299		labor	6-8	57	
colic	15-7	368	x	lymph	15-3	148	
credit-a	9-6	690	x	sonar	0-60	208	
dermat.	1-33	366	x	vote	16-0	435	x
glass	0-10	214		wine	13-0	178	
heart-c	7-6	303	x	zoo	16-1	101	

Each cell of the table 2 is formed by a pair (x,y), which x represents the significance value of the Wilcoxon test comparing our method with respect to other algorithm using the HEOM dissimilarity and y, using the HVDM dissimilarity. In bold are represented the values when our method outperforms the other method. The PRS method deletes all objects with the HVDM dissimilarity, being excluded of the statistical comparison (a - sign).

Table 2. Results of the Wicoxon test comparing our method aginst others

Condensing methods	Error	Object reduction	Editing methods	Error	Object reduction
MCS	0.809,0.12	**0.007,0.003**	ENN	0.126,0.289	**0,0**
GCNN	0.171,0.122	**0,0**	NENN	0.687,0.077	**0,0**
GBR1	0.084,0.455	**0.001,0.003**	EditRS1	0.147,0.230	**0,0**
GBR2	**0.049,0.001**	**0.024,0.005**	EditRS2	0.147,0.230	**0,0**
PRS	**0.033,0.001**	**0,-**	MSEditA	0.036*,0.305	**0,0**
CSESupport	0.629,0.099	**0.009,0.004**	MSEditB	0.059,0.050	**0,0**

* our method had worse performance

The results obtained of the experimental comparison show that our proposal obtains similar results with both dissimilarity functions. It obtains the best results according to object reduction, being significantly better than every other method. According to classifier error, the PERC method achieves very good results. It outperforms the GBR1 and GBR2 methods using both dissimilarities, and the PRS method with the HEOM dissimilarity, and ties with all other methods. Our prototype selection schema losses with respect to the MSEditA using the HEOM dissimilarity, but ties this method with the HVDM function. It is important to point out that our method maintains the classifier accuracy using only a very reduced prototype set. The above results show that combining Rough Sets and Compact Sets leads to an edited set with high accuracy and also with much less objects than the original training set.

5   Conclusions

Prototype selection for improving the classifiers accuracy is a very important task in supervised classification problems with the NN classifier. In this paper a novel method is introduced, resulting from hybridizing Rough Sets with Maximum Similarity Graphs. Our method smooth decision boundaries, by using positive decision regions, and condenses the training data by selecting representative objects from compact sets structuralizations. The method is also able deal with databases containing objects described by features no exclusively numeric or categorical. Experimental results carried out over several repository data show the high performance of the proposed method.

References

1. Triguero, I., Derrac, J., García, S., Herrera, F.: A taxonomy and Experimental Study on prototype generation for Nearest Neighbor classification. IEEE Transactions on Systems, Man, and Cybernetics (2012), doi:10.1109/TSMCC.2010.2103939
2. Dasarathy, B.D.: Minimal consistent set (MCS) identification for optimal nearest neighbor decision systems design. IEEE Transactions on Systems, Man and Cybernetics 24, 511–517 (1994)

3. Chou, C.H., Kuo, B.A., Cheng, F.: The Generalized Condensed Nearest Neighbor rule as a data reduction technique. In: 18th International Conference on Pattern Recognition, pp. 556–559 (2006)

4. Olvera-López, J.A., Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F.: Prototype Selection Via Prototype Relevance. In: Ruiz-Shulcloper, J., Kropatsch, W.G. (eds.) CIARP 2008. LNCS, vol. 5197, pp. 153–160. Springer, Heidelberg (2008)

5. Huang, C.-C.: A novel gray-based reduced NN classification method. Pattern Recognition 39, 1979–1986 (2006)

6. García-Borroto, M., Villuendas-Rey, Y., Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F.: Finding Small Consistent Subset for the Nearest Neighbor Classifier Based on Support Graphs. In: Bayro-Corrochano, E., Eklundh, J.-O. (eds.) CIARP 2009. LNCS, vol. 5856, pp. 465–472. Springer, Heidelberg (2009)

7. Zafra, A., Gibaja, E.L., Ventura, S.: Multiple instance learning with multiple objective genetic programming for web mining. Applied Soft Computing 11, 93–102 (2011)

8. Nikolaidis, K., Goulemas, J.Y., Wu, Q.H.: A class boundary preserving algorithm for data condensation. Pattern Recognition 44, 704–715 (2011)

9. Wilson, D.L.: Asymptotic properties of nearest neighbor rules using edited data. IEEE Transactions on Systems, Man and Cybernetics SMC-2, 408–421 (1972)

10. Hattori, K., Takanashi, M.: A new edited k-nearest neighbor rule in the pattern classification problem. Pattern Recognition 33, 521–528 (2000)

11. Caballero, Y., Bello, R., Salgado, Y., García, M.M.: A method to edit training set based on rough sets. International Journal of Computational Intelligence Research 3, 219–229 (2007)

12. García-Borroto, M., Villuendas-Rey, Y., Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F.: Using Maximum Similarity Graphs to Edit Nearest Neighbor Classifiers. In: Bayro-Corrochano, E., Eklundh, J.-O. (eds.) CIARP 2009. LNCS, vol. 5856, pp. 489–496. Springer, Heidelberg (2009)

13. Pawlak, Z.: Rough Sets. International Journal of Information & Computer Sciences 11, 341–356 (1982)

14. Ruiz-Shulcloper, J., Abidi, M.A.: Logical combinatorial pattern recognition: A Review. In: Pandalai, S.G. (ed.) Recent Research Developments in Pattern Recognition. Transword Research Networks, USA, pp. 133–176 (2002)

15. Hu, Q., Yu, D., Liu, J., Wu, C.: Neighborhood rough sets based heterogeneous feature selection. Information Sciences 178, 3577–3594 (2008)

16. Merz, C.J., Murphy, P.M.: UCI Repository of Machine Learning Databases. University of California at Irvine, Department of Information and Computer Science, Irvine (1998)

17. Wilson, R.D., Martinez, T.R.: Improved Heterogeneous Distance Functions. Journal of Artificial Intelligence Research 6, 1–34 (1997)