Towards Selecting Reducts for Building Decision Rules for Rule-Based Classifiers

Manuel S. Lazo-Cortés^{1(✉)}, José Fco. Martínez-Trinidad²,
Jesús A. Carrasco-Ochoa², and Nelva N. Almanza-Ortega¹

¹ TecNM/Instituto Tecnológico de Tlalnepantla, Tlalnepantla de Baz, Mexico

{manuel.lc,nelva.ao}@tlalnepantla.tecnm.mx

² Instituto Nacional de Astrofísica, Óptica y Electrónica,

Sta. Ma. Tonantzintla, Mexico

{fmartine,ariel}@inaoep.mx

Abstract. In rule-based classifiers, calculating all possible rules of a learning sample consumes many resources due to its exponential complexity. Therefore, finding ways to reduce the number and length of the rules without affecting the efficacy of a classifier remains an interesting problem. Reducts from rough set theory have been used to build rule-based classifiers by their conciseness and understanding. However, the accuracy of the classifiers based on these rules depends on the selected rule subset. In this work, we focus on analyzing three different options for using reducts for building decision rules for rule-based classifiers.

1 Introduction

Rule-base classification is a Data Mining technique that consists in, given a set of training instances, identifying certain characteristics of the instances to construct rules that are later used for classifying new instances. Rule-based classifiers are easy to interpret, easy to generate and can correctly classify new instances. The advantages of rule-based classifiers are that they are extremely expressive since they are symbolic and operate with data attributes without any transformation.

The calculation of all possible rules of a training sample is a task that requires many computational resources due to its exponential complexity. So, finding ways to reduce the number of rules without affecting the accuracy of a classifier is still an open research problem, see for example [7].

On the other hand, feature selection is a significant task in supervised classification and other problems of pattern recognition focused on eliminating irrelevant and/or redundant attributes [10]. It consists in selecting subsets of the whole set of attributes to reduce the space dimension according to certain criteria.

The objective of reducing the dimensions is to find a minimum (or almost minimum) set of attributes that retains all the essential information of the training sample for further tasks of classification or description. Reducing the dimension can help to reduce the number of generated rules, as well as their length. These rules would be simpler and easier to interpret. Then, in practical applications, minimum length descriptions are preferred (see for example [1, 5, 6, 16]).

Reducts have been used to build classifiers based on rules [8,9]. A reduct is a minimum subset of attributes that retains the same discernibility capacity as the whole set of attributes when considering objects belonging to different classes [12]. Decision rules derived from reducts are useful in practice because of their conciseness and understandability. Nevertheless, usually the number of reducts is too high and consequently the number of rules is too high too.

In this work, we focus our effort on analyzing the following three questions at using reducts for building rules in rule-based classifiers: should we use all the reducts? Is it enough to use a single reduct? Is it enough using only the shortest reducts?

Here, we present a controlled experimentation as an approach to discuss the above questions. The rest of the document is organized as follows. Section 2 provides some preliminary concepts. In Sect. 3, we present the experiments and discuss the results. Our conclusions are summarized in Sect. 4.

2 Preliminaries

In this section, we present some definitions and notations to make the paper more understandable.

2.1 Reducts

In Rough Set Theory [12], the main data representation is a decision table, which is a special case of an information table. Formally, a decision table is defined as

Definition 1 (*decision table*). A decision table is a pair $\mathcal{S}_d = (\mathcal{U}, A_t)$ where $\mathcal{U}$ denotes the set of objects under study, A_t is the set of attributes used to describe the objects.

It is important to introduce the definition of the indiscernibility relation.

Definition 2 (*indiscernibility relation*). Given a subset of conditional attributes $A \subseteq A_t$, the indiscernibility relation is defined as $IND(A|\mathbf{d}) = \{(u, v) \in \mathcal{U} \times \mathcal{U} : \forall \mathbf{a} \in A, [\mathbf{a}(u) = \mathbf{a}(v)] \vee [\mathbf{d}(u) = \mathbf{d}(v)]\}$, where $\mathbf{a}(u)$ denotes the value of the attribute $\mathbf{a}$ for the object u .

We can find several definitions of reduct (see for example, [11]), nevertheless, according to the aim of this paper, we refer to reducts assuming the classical definition of discerning decision reduct [13] as follows.

Definition 3 (*reduct*). Given a decision table $\mathcal{S}_d$, an attribute set $R \subseteq A_t$ is called a reduct, if R satisfies the following two conditions:

- (i) $IND(R|\mathbf{d}) = IND(A_t|\mathbf{d})$;
- (ii) For any $\mathbf{a} \in R$, $IND((R - \{\mathbf{a}\})|\mathbf{d}) \neq IND(A_t|\mathbf{d})$.

All attribute subsets satisfying condition (i) are called super-reducts.

This definition ensures that a reduct has no lower ability to distinguish objects belonging to different classes than the whole set of attributes, being minimal with regard to inclusion, i.e. a reduct does not contain redundant attributes or, equivalently, a reduct does not include other super-reducts. The original idea of reduct is based on inter-class comparisons.

Example 1. Given the following decision table, where $A_t = \{a_1, a_2, a_3, a_4, a_5\}$ is the set of attributes, $\mathcal{U} = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\}$ is the set of objects and $\mathbf{d}$ is the decision attribute.

$$M = \begin{array}{c} \begin{array}{ccccc} & a_1 & a_2 & a_3 & a_4 & a_5 & \mathbf{d} \\ \begin{array}{c} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{array} & \left(\begin{array}{cccccc} 4 & high & 0 & GR & 0 & 1 \\ 3 & high & 1 & MX & 1 & 1 \\ 3 & low & 0 & MX & 2 & 1 \\ 2 & low & 1 & GR & 0 & 1 \\ 5 & high & 1 & BC & 3 & 2 \\ 7 & veryhigh & 1 & HG & 4 & 2 \\ 7 & medium & 1 & TM & 3 & 3 \\ 8 & medium & 0 & BC & 1 & 3 \end{array} \right) \end{array} \end{array}$$

We have that $\{a_1, a_2\}$, $\{a_2, a_4\}$, $\{a_3, a_4\}$ and $\{a_1, a_3, a_5\}$ are reducts. Notice that for example $\{a_2, a_3\}$ does not fulfill the definition of reduct, since $\mathbf{a}_2(u_2) = \mathbf{a}_2(u_5) = high$ and $\mathbf{a}_3(u_2) = \mathbf{a}_3(u_5) = 1$ being $1 = \mathbf{d}(u_2) \neq \mathbf{d}(u_5) = 2$.

If in M we just keep the columns belonging to a reduct, you can easily see that the classes remain distinguishable, and each column is essential for that purpose.

M_1 , M_2 , M_3 and M_4 are the resulting decision tables when the representation space is reduced respectively to each one of the reducts.

$$M_1 = \begin{array}{c} \begin{array}{ccc} & a_1 & a_2 & \mathbf{d} \\ \begin{array}{c} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{array} & \left(\begin{array}{ccc} 4 & high & 1 \\ 3 & high & 1 \\ 3 & low & 1 \\ 2 & low & 1 \\ 5 & high & 2 \\ 7 & veryhigh & 2 \\ 7 & medium & 3 \\ 8 & medium & 3 \end{array} \right) \end{array} \end{array}$$

$$M_2 = \begin{array}{c} \begin{array}{ccc} & a_2 & a_4 & \mathbf{d} \\ \begin{array}{c} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{array} & \left(\begin{array}{ccc} high & GR & 1 \\ high & MX & 1 \\ low & MX & 1 \\ low & GR & 1 \\ high & BC & 2 \\ veryhigh & HG & 2 \\ medium & TM & 3 \\ medium & BC & 3 \end{array} \right) \end{array} \end{array}$$

$$M_3 = \begin{array}{c} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{array} \begin{array}{ccc} a_3 & a_4 & \mathbf{d} \\ \left(\begin{array}{ccc} 0 & GR & 1 \\ 1 & MX & 1 \\ 0 & MX & 1 \\ 1 & GR & 1 \\ 1 & BC & 2 \\ 1 & HG & 2 \\ 1 & TM & 3 \\ 0 & BC & 3 \end{array} \right) \end{array}$$

$$M_4 = \begin{array}{c} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{array} \begin{array}{ccc} a_1 & a_3 & a_5 & \mathbf{d} \\ \left(\begin{array}{ccc} 4 & 0 & 0 & 1 \\ 3 & 1 & 1 & 1 \\ 3 & 0 & 2 & 1 \\ 2 & 1 & 0 & 1 \\ 5 & 1 & 3 & 2 \\ 7 & 1 & 4 & 2 \\ 7 & 1 & 3 & 3 \\ 8 & 0 & 1 & 3 \end{array} \right) \end{array}$$

2.2 Rule-Based Classifier

The generation of effective rules is essential for the development of any classifier that is easily understandable by the user. Any mechanism used for rule generation must maintain the underlying semantics of the feature set.

Typically, rule-based classifiers are sensitive to the dimensionality of the dataset since a large number of superfluous or redundant rules may appear. This makes it advisable to try to reduce the dimensionality of the data, and/or the length or complexity of the rules so that the resulting set of learned rules becomes manageable and can overcome the classification results obtained by using rules containing too many attributes.

To build the set of decision rules to be used in our rule-based classifiers, we used the tools included in the software RSES ver. 2.2.2 [4], which has been widely used in the literature, see for example [3, 14, 15].

In RSES, once the reducts of a decision table have been computed, each object in the training sample is matched against each reduct. This matching gives as result a rule having in its conditional part, the attributes of the reduct, each one associated with the values of the currently considered object, and in its decision part it has the class of this training object.

At classifying an unseen object through the generated rule set, it may happen that several rules suggest different decision values. In such conflict situations a strategy to reach a final result (decision) is needed. RSES provides a conflict resolution strategy based on voting. In this method, when the antecedent of a rule matches the unseen object, a vote in favor of the decision value of its consequent is cast. Votes are counted and the decision value reaching the majority of the votes is chosen as the class for the unseen object.

This simple method may be extended by assigning weights to rules. In RSES, this method (known as Standard Voting) assigns as weight for a rule the number of training objects matching the antecedent of this rule. Then, each rule votes with its weight and the decision value reaching the highest weight sum is considered as the class for the object.

Example 2. If we consider the table M_1 from Example 1, applying RSES, we obtain the following rules:

$$\begin{aligned} (a_1 = 4) &\Rightarrow (d = 1[1]) \\ (a_1 = 3) &\Rightarrow (d = 1[2]) \end{aligned}$$

$(a2 = low) \Rightarrow (d = 1[2])$
 $(a1 = 5) \Rightarrow (d = 2[1])$
 $(a2 = veryhigh) \Rightarrow (d = 2[1])$
 $(a2 = medium) \Rightarrow (d = 3[2])$

the number in square brackets at the end means the support of the rule, i.e., the number of objects in the training sample having in their respective features the same values of the features in the antecedent of the rule.

Obviously, we can generate a set of rules from any subset of attributes or a set of subsets of attributes. For our study, we use all the reducts, each one of them individually, and all the shortest reducts as a set of attribute subsets..

3 Experiments

In this section, we show our experimentation in controlled conditions, as a first approach to study three different options to use reducts for building decision rules in rule-based classifiers.

For having controlled conditions in our experiments, we used four datasets (see Table 1) taken from the UCI Machine Learning Repository [2]. We select these datasets because they are small and have a small amount of reducts, and not all of them have the same length and the minimum length is reached in more than one reduct. The datasets *Glass* and *Heart(Statlog)* were previously discretized. All datasets were split into two folds, one for generating rules and the other for testing. For this, we used a ratio of 0.5.

For each dataset, the whole set of reducts for the training fold was computed by using RSES [18]. Table 1 shows the characteristics of the reducts: the third column contains the amount of reducts, the next three columns contain the maximum, minimum and average length respectively; and the last column contain the number of shortest reducts.

Table 1. Characteristics of reducts for the four datasets used in our experiments

Dataset	Attributes	Reducts	Max length	Min length	Avg length	Shortest
Glass	9	4	6	5	5.5	2
Heart-(Statlog)	13	62	8	5	6.8	2
Pima Indians Diabetes	8	12	6	4	4.5	8
Zoo	16	34	5	4	4.9	2

Figure 1 shows a screenshot of RSES for one of the projects executed in the study of the selected databases.

Fig. 1. Screenshot of the project Zoo in RSES

After computing the reducts, using again RSES, the before mentioned sets of rules were generated, first using all the reducts, then using each reduct separately, in addition the rule set considering only the shortest reducts was generated. To use an external selection criterion, the CAMARDF [19] algorithm was used, which generates a minimum length reduct, and the set of rules obtained from that shortest reduct was separately considered.

Each testing fold was classified using the rule-based algorithm standard voting, taking into account the four different cases:

1. the rules generated by all the reducts
2. the rules generated by each reduct individually
3. the rules generated by all the shortest reducts and
4. the rules generated by one shortest reduct obtained by the CAMARDF algorithm.

Table 2 shows the obtained results. The columns headed as *all*, *shortest* and *CAMARDF* contain the accuracies obtained for variants 1, 3 and 4, respectively. The remaining columns contain the maximum and minimum accuracy values obtained when applying variant 2.

As we can see from Table 2, for the second and third databases, the best result is achieved when considering the rules obtained from all reducts. In the case of Pima-diabetes, this result is also obtained if the rules generated by the

Table 2. Accuracies of the different rule-based classifiers for the four datasets

Dataset	All	Min	Max	Shortest	CAMARDF
Glass	0.60	0.46	0.55	0.63	0.53
Heart-(Statlog)	0.89	0.58	0.85	0.66	0.58
Pima Indians Diabetes	0.69	0.54	0.62	0.69	0.54
Zoo	0.94	0.84	0.96	0.88	0.88

shortest reducts are considered. For the Glass dataset, the best accuracy was achieved for the classifier built with the shortest reducts.

However, in the case of the Zoo dataset, neither the rule-based classifier built with all the reducts, nor the classifier built with the shortest ones, obtained the highest result. For this dataset, the best result was obtained twice, for a classifier based on an individual reduct, but in none of the cases it was obtained by using one of the shortest reducts. Taking into account that for this dataset there are 34 reducts, it means that if we randomly choose a reduct to generate the rules for a classifier, the probability of constructing one of maximum accuracy classifiers is approximately 0.06. If we decide to choose one of minimum length reducts, then the probability is 0.

Apparently, Sil and Das [17] report a different result for the Zoo dataset using the 10-fold cross-validation technique, although the authors' use of the term minimum length reducts is confusing. As they say, they achieve the minimum length reducts by eliminating the redundant attributes of each of the reducts, which is unclear.

Finally, if we decide to select just one shortest reduct to build the classifier using the CAMARDF algorithm, it never would achieve the best result, for any of the four datasets.

4 Conclusions

The problem of how to choose the best rules when building a rule-based classifier remains a problem to solve. Although some authors underestimate the task of computing all reducts of a dataset (perhaps covered by its requirement of excessive resources, given its exponential complexity), our preliminary experiments allow us to conclude that in certain cases, computing a single reduct, or only considering those with a minimum length can provide insufficient results compared to those obtained from all reducts.

These results do not lead us to suggest that all reducts should be computed in every problem. More than anything else, our purpose is to establish that it is an unclosed matter and to emphasize that it is justified to continue investigating effective strategies for the selection of rules, especially if for their construction we rely on reducts, given their properties with respect to the ability to discern between objects of different classes.

References

1. Arora, S., Anand, P.: Binary butterfly optimization approaches for feature selection. *Expert Syst. Appl.* **116**, 147–160 (2019)
2. Bache, K., Lichman, M.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine (2013). <http://archive.ics.uci.edu/ml>
3. Barman, T., Rajesh, G., Archana, R.: Rough set based segmentation and classification model for ECG. In: Conference on Advances in Signal Processing (CASP), pp. 18–23. IEEE (2016)
4. Bazan, J.G., Szczuka, M.: The rough set exploration system. In: Peters, J.F., Skowron, A. (eds.) *Transactions on Rough Sets III*. LNCS, vol. 3400, pp. 37–56. Springer, Heidelberg (2005). https://doi.org/10.1007/11427834_2
5. El-Islem-Karabadi, N., Khelf, I., Seridi, H., Aridhi, S., Remond, D., Dhifli, W.: A data sampling and attribute selection strategy for improving decision tree construction. *Expert Syst. Appl.* **129**, 84–96 (2019)
6. Hansen, M., Yu, B.: Model selection and the principle of minimum description length. *J. Am. Stat. Assoc.* **96**, 746–774 (2001)
7. Herrera-Semenets, V., Pérez-García, O.A., Hernández-León, R., van den Berg, J., Doerr, C.: A data reduction strategy and its application on scan and backscatter detection using rule-based classifiers. *Expert Syst. Appl.* **95**, 272–279 (2018)
8. Lazo-Cortés, M.S., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A.: Class-specific reducts vs. classic reducts in a rule-based classifier: a case study. In: Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A., Olvera-López, J.A., Sarkar, S. (eds.) *MCPR 2018*. LNCS, vol. 10880, pp. 23–30. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-92198-3_3
9. Lazo-Cortés, M.S., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A.: On the use of constructs for rule-based classification: a case study. In: Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F., Olvera-López, J.A., Salas, J. (eds.) *MCPR 2019*. LNCS, vol. 11524, pp. 327–335. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21077-9_30
10. Liu, H., Motoda, H.: *Computational Methods of Feature Selection*. Chapman & Hall/CRC, Boca Raton (2007)
11. Miao, D.Q., Zhao, Y., Yao, Y.Y., Li, H.X., Xu, F.F.: Reducts in consistent and inconsistent decision tables of the Pawlak rough set model. *Inf. Sci.* **179**(24), 4140–4150 (2009)
12. Pawlak, Z.: Rough sets. *Int. J. Comput. Inf. Sci.* **11**, 341–356 (1982)
13. Pawlak, Z.: *Rough sets, Theoretical Aspects of Reasoning About Data*, pp. 315–330. Kluwer Academic Publishers, Dordrecht (1992)
14. Rana, H., Lal, M.: A rough set theory approach for rule generation and validation using RSES. *Int. J. Rough Sets Data Anal.* **3**(1), 55–70 (2016)
15. Rana, H., Lal, M.: A comparative study based on rough set and classification via clustering approaches to handle incomplete data to predict learning styles. *Int. J. Decis. Support Syst. Technol.* **9**(2), 1–20 (2017)
16. Si, H., Zhou, J., Chen, Z., Wan, J., Xiong, N., Zhang, W., Vasilakos, A.: Association rules mining among interests and applications for users on social networks. *IEEE Access* **7**, 116014–116026 (2019)
17. Sil, J., Das, A.K.: Variable length reduct vs. minimum length reduct-a comparative study. *Procedia Technol.* **4**, 58–68 (2012)

18. Skowron, A., Bazan, J., Szczuka, M., Wroblewski, J.: Rough Set Exploration System (version 2.2.2). <http://logic.mimuw.edu.pl/~rses/>
19. Zhou, J., Miao, D., Feng, Q., Sun, L.: Research on complete algorithms for minimal attribute reduction. In: Wen, P., Li, Y., Polkowski, L., Yao, Y., Tsumoto, S., Wang, G. (eds.) RSKT 2009. LNCS (LNAI), vol. 5589, pp. 152–159. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-02962-2_19