A Comparison of the LERS Classification System and Rule Management in PRSM

Jerzy W. Grzymala-Busse[1] and Yiyu Yao[2]

[1] Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS 66045, USA and
Institute of Computer Science Polish Academy of Sciences, 01-237 Warsaw, Poland
jerzy@ku.edu
http://lightning.eecs.ku.edu/index.html
[2] Department of Computer Science, University of Regina, Regina, Saskatchewan, Canada S4S 0A2
yyao@cs.uregina.ca

Abstract. The LERS classification system and rule management in probabilistic rough set models (PRSM) are compared according to the interpretations of rules, quantitative measures of rules, and rule conflict resolution when applying rules to classify new cases. Based on the notions of positive and boundary regions, probabilistic rules are semantically interpreted as the positive and boundary rules, respectively. Rules are associated with different quantitative measures in LERS and PRSM, reflecting different characteristics of rules. Finally, the rule conflict resolution method used in LERS may be applied to PRSM.

1 Introduction

Rule induction is one of the most important applications of rough set theory [5,6,8,9,11,17]. In the standard rough set model, one typically interprets rules induced from the positive region (i.e., the lower approximation) of a concept (class) as *certain* rules and rules induced from the boundary region (i.e., the difference of upper approximation and lower approximation) as *uncertain* or *plausible* rules. One may associate quantitative measures to rules. For example, the precision of a rule, also called accuracy and confidence, is the conditional probability that a rule correctly indicates the concept given the set of all cases matching the rule. From the point view of precision, the interpretation of certain and uncertain rules is reasonable, as the precision of a certain rule is 1 and precision of a plausible rule is between 0 and 1.

The lack of consideration for the degree of overlap of an equivalence class and a concept had motivated many authors to consider probabilistic rough set models (PRSM). Pawlak, Wong, Ziarko [10] proposed to use 0.5 as a threshold to define probabilistic rough set approximations. Yao and Wong [13,14,15] proposed the decision-theoretic rough set model (DTRSM) in which a pair of threshold parameters for defining probabilistic approximations can be determined based on the

C.-C. Chan et al. (Eds.): RSCTC 2008, LNAI 5306, pp. 202–210, 2008.

well established Bayesian decision theory. That is, the probabilistic approximations defined by the parameters would incur minimal risk in deciding the positive, boundary and negative regions. Based on intuitive arguments, Ziarko [16] proposed variable precision rough set model (VPRSM) for probabilistic approximations. Once probabilistic approximations are introduced, one can similarly derive rules [6,11,12].

There is a semantics difficulty with interpreting probabilistic rules induced from the probabilistic positive region, since they are also uncertain (i.e., precision < 1). From the precision point of view, there is no difference between probabilistic rules induced from probabilistic positive and boundary regions, except for their levels of precision. However, this important problem has not received much attention until recently. A solution to the problem is offered by the decision-theoretic rough set model. Given a class, its positive, boundary and negative regions represent three different types of decisions. For example, consider classifying a set of patients according to a particular disease. A patient in the positive region needs “immediately treatment”, a patient in the boundary requires “further investigation”, and a patient in the negative region does not require any treatment. With respect to the first two cases, the notions of positive rules and boundary rules have been introduced [13]. They properly reflect the semantics interpretations of rules induced in PRSM.

Another important issue that need to be considered in PRSM is rule conflict resolution when rules are applied to classify new cases. Many studies focus more on rule induction and pay less attention to rule evaluation where rule conflict resolution must be considered. A solution for rule conflict resolution has been explored in LERS [2,3,4,5], where bucket brigade algorithm [1,7] is adopted and modified. In addition, LERS use different quantitative measure to characterize rules.

Based on the above discussion, we present a comparative study of LERS classification system and rule management in PRSM. This comparison enables us to pool together advantages of the two approaches in an attempt to obtain better rule induction algorithms within rough set theory.

2 Rule Induction

First we are going to present LEM2 (Learning from Examples Module, version 2) methodology of rule induction based on attribute-value pair blocks. LEM2 is one of rule induction modules of the LERS (Learning from Examples based on Rough Sets) data mining system.

2.1 Blocks of Attribute-Value Pairs

We assume that the input data sets are presented in the form of a *decision table*. An example of a decision table is shown in Table 1. Rows of the decision table represent *cases*, while columns are labeled by *variables*. The set of all cases will be denoted by U. In Table 1, $U = \{1, 2, ..., 19\}$. Independent variables are called *attributes* and a dependent variable is called a *decision* and is denoted by d. The

Table 1. A complete decision table

	Attributes		Decision
Case	Width	Gauge	Quality
1	wide	heavy	good
2	wide	heavy	good
3	wide	heavy	good
4	wide	medium	good
5	wide	medium	good
6	wide	medium	bad
7	wide	light	good
8	wide	light	good
9	wide	light	bad
10	wide	light	bad
11	narrow	heavy	good
12	narrow	heavy	good
13	narrow	heavy	good
14	narrow	heavy	bad
15	narrow	medium	good
16	narrow	medium	good
17	narrow	medium	bad
18	narrow	light	bad
19	narrow	light	bad

set of all attributes will be denoted by A. In Table 1, $A = \{Width, Gauge\}$. Any decision table defines a function ρ that maps the direct product of U and A into the set of all values. For example, in Table 1, $\rho(1, Width) = wide$. A decision table with an incompletely specified function ρ will be called *incomplete*.

An important tool to analyze complete decision tables is a block of an attribute-value pair. Let a be an attribute, i.e., $a \in A$ and let v be a value of a for some case. For complete decision tables if $t = (a, v)$ is an attribute-value pair then a *block* of t, denoted $[t]$, is a set of all cases from U that for attribute a have value v. Each attribute-value pair represents one piece of knowledge about a decision table or a property of cases. These pieces of knowledge and the corresponding blocks will serve as a basis of rule induction.

For Table 1, we have,

$$[(Width, wide)] = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\},$$
$$[(Width, narrow)] = \{11, 12, 13, 14, 15, 16, 17, 18, 19\},$$
$$[(Gauge, heavy)] = \{1, 2, 3, 11, 12, 13, 14\},$$
$$[(Gauge, medium)] = \{4, 5, 6, 15, 16, 17\},$$
$$[(Gauhe, light)] = \{7, 8, 9, 10, 18, 19\}.$$

Moreover, the two important blocks related with the decision *Quality*, called *concepts*, are:

$$[(Quality, good)] = \{1, 2, 3, 4, 5, 7, 8, 11, 12, 13, 15, 16\},$$
$$[(Quality, bad)] = \{6, 9, 10, 14, 17, 18, 19\}.$$

These blocks represent knowledge about the entire decision table. Rule induction is essential to find relationship between the blocks defined by attributes and the blocks defined by a decision.

The notion of blocks can be used to explain the basic concepts of the rough set theory [8,9]. Let B be a nonempty subset of A. The indiscernibility relation *IND*(B) is a relation on U defined for $x, y \in U$ as follows:

$$(x, y) \in IND(B) \;\; if\ and\ only\ if \;\; \rho(x, a) = \rho(y, a) \;\; for\ all\ a \in B.$$

The indiscernibility relation $IND(B)$ is an equivalence relation. Equivalence classes of $IND(B)$ are called *elementary sets* of B and are denoted by $[x]_B$. The indiscernibility relation $IND(B)$ may be computed using the idea of blocks of attribute-value pairs. More specifically, the elementary blocks of $IND(B)$ are intersections of the corresponding blocks of attribute-value pairs, i.e., for any case $x \in U$,

$$[x]_B = \bigcap\{[(a, v)] | a \in B, \rho(x, a) = v\}.$$

In other words, the elementary block containing x is intersection all blocks defined by values of x all attributes in B.

For Table 1, the elementary sets of $IND(A)$ are given by:

$$[1]_A = [(Width, wide)] \cap [(Gauge, heavy)] = \{1, 2, 3\} = [2]_A = [3]_A,$$
$$[4]_A = [(Width, wide)] \cap [(Gauge, medium)] = \{4, 5, 6\} = [5]_A = [6]_A,$$
$$[7]_A = [(Width, wide)] \cap [(Gauge, light)] = \{7, 8, 9, 10\} = [8]_A = [9]_A = [10]_A,$$
$$[11]_A = [(Width, narrow)] \cap [(Gauge, heavy)] = \{11, 12, 13, 14\} = [12]_A = [13]_A = [14]_A,$$
$$[15]_A = [(Width, narrow)] \cap [(Gauge, medium)] = \{15, 16, 17\} = [16]_A = [17]_A,$$
$$[18]_A = [(Width, narrow)] \cap [(Gauge, light)] = \{18, 19\} = [19]_A.$$

It follows that the elementary blocks of $IND(A)$ are $\{1, 2, 3\}$, {4, 5, 6}, {7, 8, 9, 10}, {11, 12, 13, 14}, {15, 16, 17} and {18, 19}.

2.2 Rules in LERS

Based on the elementary blocks of the equivalence relation induced by a subset B of the attribute set A, one can define a pair of lower and upper approximations for each concept $D_i \subseteq U$. That is,

$$\underline{apr}_B(D_i) = \bigcup\{[x]_B \mid [x]_B \subseteq D_i\}$$

$$= \bigcup\{[x]_B \mid P(D_i \mid [x]_B) = 1\};$$
$$\overline{apr}_B(D_i) = \bigcup\{[x]_B \mid [x]_B \cap D_i \neq \emptyset\}$$
$$= \bigcup\{[x]_B \mid P(D_i \mid [x]_B) > 0\},$$

where $P(D_i \mid [x]_D) = |D_i \cap [x]_B|/|[x]_B|$ is the conditional probability and $|\cdot|$ is the cardinality of a set.

Thus, the lower approximations of the concepts from Table 1 are:

$$\underline{apr}_A([(Quality, good)]) = \{1, 2, 3\},$$
$$\underline{apr}_A([(Quality, bad)]) = \{18, 19\},$$

And the upper approximations of the concepts from Table 1 are:

$$\overline{apr}_A([(Quality, good)]) = \{1, 2, ..., 17\},$$
$$\overline{apr}_A([(Quality, bad)]) = \{4, 5, ..., 19\}.$$

The LERS data mining system computes lower and upper approximations for every concept and then induces rules using one of the selected modules. Rules induced from lower and upper approximations are called *certain* and *possible*, respectively [2].

The LEM2 algorithm search for rules by using a family of blocks such that their intersection is either a subset of the concept or has an overlap with the concept [3]. In the LERS format, every rule is associated with three numbers: the total number of attribute-value pairs on the left-hand side of the rule, the total number of cases correctly classified by the rule during training, and the total number of training cases matching the left-hand side of the rule, i.e., the rule domain size.

For Table 1, the LEM2 module of LERS induces the following rule sets:
the certain rule set:

2, 3, 3
(Gauge, heavy) & (Width, wide) -> (Quality, good),
2, 2, 2
(Gauge, light) & (Width, narrow) -> (Quality, bad),

and the following possible rule set:

1, 7, 10
(Width, wide) -> (Quality, good),
1, 6, 7
(Gauge, heavy) -> (Quality, good),
1, 4, 6
(Gauge, medium) -> (Quality, good),
1, 4, 9
(Width, narrow) -> (Quality, bad),
1, 4, 6
(Gauge, light) -> (Quality, bad),
1, 2, 6
(Gauge, medium) -> (Quality, bad).

2.3 Rules in PRSM

The set $\mathrm{POS}_B(D_i) = \underline{apr}_B(D_i)$ is called the positive region of D_i, and the set $\mathrm{BND}_B(D_i) = \overline{apr}_B(D_i) - \underline{apr}_B(D_i)$ is called the boundary region of D_i. According to the two regions, one can form two types of rules called *positive* and *boundary* rules, respectively [13].

If an elementary block is in the positive region of a decision class, one obtains a positive rule; if the elementary block is in the boundary region, one obtains one or several boundary rules. In particular, in the VPRSM format, each rule is associated with two numbers: the conditional probability and marginal probability [6].

For Table 1, we have the following positive rules:

$$\text{(P1)}.\ (Width, wide)\ \&\ (Gauge, heavy) \longrightarrow (Quality, good),\quad 1.00, 0.158,$$
$$\text{(P2)}.\ (Width, narrow)\ \&\ (Gauge, light) \longrightarrow (Quality, bad),\quad 1.00, 0.158,$$

and the boundary rules:

$$\text{(B1)}.\ (Width, wide)\ \&\ (Gauge, medium) \longrightarrow (Quality, good),\quad 0.67, 0.158,$$
$$\text{(B2)}.\ (Width, wide)\ \&\ (Gauge, medium) \longrightarrow (Quality, bad),\quad 0.33, 0.158,$$
$$\text{(B3)}.\ (Width, wide)\ \&\ (Gauge, light) \longrightarrow (Quality, good),\quad 0.50, 0.211,$$
$$\text{(B4)}.\ (Width, wide)\ \&\ (Gauge, light) \longrightarrow (Quality, bad),\quad 0.50, 0.211,$$
$$\text{(B5)}.\ (Width, narrow)\ \&\ (Gauge, heavy) \longrightarrow (Quality, good),\quad 0.25, 0.211,$$
$$\text{(B6)}.\ (Width, narrow)\ \&\ (Gauge, heavy) \longrightarrow (Quality, bad),\quad 0.75, 0.211,$$
$$\text{(B7)}.\ (Width, narrow)\ \&\ (Gauge, medium) \longrightarrow (Quality, good),\quad 0.67, 0.158,$$
$$\text{(B8)}.\ (Width, narrow)\ \&\ (Gauge, medium) \longrightarrow (Quality, bad), 0.33, 0.158,$$

The two types of rules lead to two types of different decision. A positive rule suggests a definite and positive decision regarding the class of a case, and a boundary rule suggests a tentative and boundary decision regarding the class of a case. Semantically, these two classes are different [13].

In probabilistic approaches to rough sets, such as decision-theoretic model [13,14] and variable precision model [16], we have the parameterized approximations:

$$\underline{apr}_B(D_i) = \bigcup\{[x]_B \mid P(D_i \mid [x]_B) \geq \alpha\},$$
$$\overline{apr}_B(D_i) = \bigcup\{[x]_B \mid P(D_i \mid [x]_B) > \beta\},$$

with $\alpha > \beta$. They are referred to as the α-level lower approximation and β-level upper approximation. Similarly, the α-level positive region and the (α, β)-level boundary region can be introduced. Again, we have two types of rules corresponding the the two region.

Suppose $\alpha = 0.75$ and $\beta = 0.50$. For Table 1, rule (B6) becomes a 0.75-level positive rule, and only rules (B1) and (B7) remain to be $(0.75, 0.50)$-level boundary rules. On the other hand, for comparison, the previous rule sets, used in the VPRSM methodology, presented in the LERS format, are:

2, 3, 3
(Width, wide) & (Gauge, heavy) -> (Quality, good),
2, 2, 3
(Width, wide) & (Gauge, medium) -> (Quality, good),
2, 1, 3
(Width, wide) & (Gauge, medium) -> (Quality, bad),
2, 2, 4
(Width, wide) & (Gauge, light) -> (Quality, good),
2, 2, 4
(Width, wide) & (Gauge, light) -> (Quality, bad),
2, 3, 4
(Width, narrow) & (Gauge, heavy) -> (Quality, good),
2, 1, 4
(Width, narrow) & (Gauge, heavy) -> (Quality, bad),
2, 2, 3
(Width, narrow) & (Gauge, medium) -> (Quality, good),
2, 1, 3
(Width, narrow) & (Gauge, medium) -> (Quality, bad) and
2, 2, 2
(Width, narrow) & (Gauge, light) -> (Quality, bad).

With the additional information: $|U| = 19$, rules with the LERS format may be easily converted into VPRSM format, the converse is not true. The conditional probability is a ratio of the second LERS number to the third LERS number, the marginal probability is the ratio of the third LERS number to the cardinality of the universe. By the way, the cardinality of the universe is the same for all rules so it does not need to be recorded for a specific rule.

3 Rule Conflict Resolution

The classification system of LERS is a modification of the *bucket brigade algorithm* [1,7]. The decision to which concept a case belongs is made on the basis of three factors: specificity_factor, strength_factor, and support. They are defined as follows: *specificity_factor* is either the *specificity*, i.e., the total number of attribute-value pairs on the left-hand side of the rule or may be selected by the user to be equal to one. *Strength_factor* is either the *strength*, i.e., total number of cases correctly classified by the rule during training or *rough measure*, i.e., the ratio of the strength to the total number of training cases matching the left-hand side of the rule. For completely specified data sets the rough measure is identical with the conditional probability of the concept given the rule domain. The third factor, *support*, is defined as the sum of scores of all matching rules from the concept, where the score of the rule is the product of its strength_factor and specificity_factor. The concept C for which the support, i.e., the following expression

$$\sum_{matching\ rules\ R\ describing\ C} Strength\_factor(R) * Specificity\_factor(R)$$

is the largest is the winner and the case is classified as being a member of C. Note that the user may exclude support, i.e., the case might be classified only on the basis of its scores associated with rules.

In the classification system of LERS, if complete matching is impossible, all partially matching rules are identified. These are rules with at least one attribute-value pair matching the corresponding attribute-value pair of a case. For any partially matching rule R, the additional factor, called *matching_factor* is computed. Matching_factor (R) is defined as the ratio of the number of matched attribute-value pairs of R with a case to the total number of attribute-value pairs of R. Again, the user may choose the matching_factor to be equal to one. In partial matching, the concept C for which the following expression is the largest

$$\sum_{\substack{partially\ matching \\ rules\ R\ describing\ C}} Matching\_factor(R) * Strength\_factor(R) * Specificity\_factor(R)$$

is the winner and the case is classified as being a member of C.

In general the LERS classification system uses four binary parameters: specificity_factor (either equal to specificity or switched to integer one), strength_factor (either the total number of well-classified training cases or the rough measure), support (either product of scores for each matching rule or each rule participates on its own), and finally matching_factor (either as defined or equal to integer one). Thus the user of the LERS classification system may apply one of 16 different strategies [5]. In the VPRSM methodology, classification is based on conditional probability, one of 16 LERS strategies (in [5] this strategy, based only on the conditional probability, is the strategy # 15). Note that the choice of the classification strategy is crucial and that the best strategy is based on specificity = 1, strength, support, and matching_factor [5].

4 Conclusions

The LERS system induces rules based on attribute-value pairs. Since LERS keep three important quantities of rules, namely, the total number of attributes on the left-hand side of the rule, the total number of cases correctly classified by the rule, and the total number of cases matching the left-hand side of the rule, LERS can be easily applied to discover probabilistic rules. Based on two decision-theoretic rough set model, two types of rules, known as positive rules and boundary rules, can be introduced. LERS system can easily learn the two types of rules. In addition, the rule conflict resolution strategy of LERS can be applied to rule applications and evaluation in probabilistic rough set models.

References

1. Booker, L.B., Goldberg, D.E., Holland, J.F.: Classifier systems and genetic algorithms. In: Carbonell, J.G. (ed.) Machine Learning. Paradigms and Methods, pp. 235–282. The MIT Press, Menlo Park (1990)
2. Grzymala-Busse, J.W.: Knowledge acquisition under uncertainty—A rough set approach. Journal of Intelligent & Robotic Systems 1, 3–16 (1988)

3. Grzymala-Busse, J.W.: A new version of the rule induction system LERS. Fundamenta Informaticae 31, 27–39 (1997)
4. Grzymala-Busse, J.W.: MLEM2: A new algorithm for rule induction from imperfect data. In: Proceedings of the 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2002), pp. 243–250 (2002)
5. Grzymala-Busse, J.W., Wang, C.P.B.: Classification methods in rule induction. In: Proceedings of the Fifth Intelligent Information Systems Workshop, pp. 120–126 (1996)
6. Grzymala-Busse, J.W., Ziarko, W.: Data mining based on rough sets. In: Wang, J. (ed.) Data Mining: Opportunities and Challenges, pp. 142–173. Idea Group Publ. (2003)
7. Holland, J.H., Holyoak, K.J., Nisbett, R.E.: Induction. In: Processes of Inference, Learning, and Discovery. The MIT Press, Menlo Park (1986)
8. Pawlak, Z.: Rough Sets. International Journal of Computer and Information Sciences 11, 341–356 (1982)
9. Pawlak, Z.: Rough Sets. Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)
10. Pawlak, Z., Wong, S.K.M., Ziarko, W.: Rough sets: probabilistic versus deterministic approach. International Journal of Man-Machine Studies 29, 81–95 (1988)
11. Stefanowski, J.: Algorithms of Decision Rule Induction in Data Mining. Poznan University of Technology Press, Poznan (2001)
12. Tsumoto, S., Tanaka, H.: PRIMEROSE: Probabilistic Rule Induction Method based on Rough Sets and Resampling Methods. Computational Intelligence 11, 389–405 (1995)
13. Yao, Y.Y.: Decision-theoretic rough set models. In: Yao, J., Lingras, P., Wu, W.-Z., Szczuka, M.S., Cercone, N.J., Ślęzak, D. (eds.) RSKT 2007. LNCS (LNAI), vol. 4481, pp. 1–12. Springer, Heidelberg (2007)
14. Yao, Y.Y., Wong, S.K.M.: A decision theoretic framework for approximating concepts. International Journal of Man-machine Studies 37, 793–809 (1992)
15. Yao, Y.Y., Wong, S.K.M., Lingras, P.: A decision-theoretic rough set model. In: Proceedings of the 5th Symposium on Methodologies for Intelligent Systems, pp. 17–24 (1990)
16. Ziarko, W.: Variable precision rough sets model. Journal of Computer and Systems Sciences 46, 39–59 (1993)
17. Ziarko, W.: Optimal decision making with data-acquired decision tables. In: Proceedings of the Intelligent Information Systems Symposium, pp. 75–85 (2000)