On Decomposition for Incomplete Data

Rafał Latkowski

Institute of Computer Science, Warsaw University
ul. Banacha 2, 02–097 Warsaw, Poland
rlatkows@mimuw.edu.pl

Abstract. In this paper we present a method of data decomposition to avoid the necessity of reasoning on data with missing attribute values. The original incomplete data is decomposed into data subsets without missing values. Next, methods for classifier induction are applied to such sets. Finally, a conflict resolving method is used to combine partial answers from classifiers to obtain final classification. We provide an empirical evaluation of the decomposition method accuracy and model size with use of various decomposition criteria.

1 Introduction

In recent years a great research effort has been made to develop methods inducing classifiers for data with missing attribute values. Some approaches making possible to handle missing attribute values have been developed within the Rough Sets framework [7, 15]. In those approaches a modification of indiscernibility relation is considered to handle missing attribute values. The other approach presented in *LEM1* and *LEM2* methods [4, 5] is to modify an algorithm that search for all or covering set of decision rules. In this paper we present a method of data decomposition to avoid the necessity of reasoning on data with missing attribute values and without modification of the inductive learning algorithm itself.

The decomposition method was developed to meet certain assumptions. The primary aim was to find a possibility to adapt many existing, well known classification methods that are initially not able to handle missing attribute values to the case of incomplete data. The secondary aim was to cope with the problem of incomplete information systems without making an additional assumption on independent random distribution of missing values and without using data imputation methods [3, 4]. Many real world applications have showed that appearance of missing values is governed by very complicated dependencies and the application of an arbitrary method for data imputation can increase error rate of the classifier.

The decomposition method tries to avoid the necessity of reasoning on data with missing attribute values. The original incomplete data is decomposed into data subsets without missing values. Next, methods for classifier induction are applied to such sets. Finally, a conflict resolving method is used to combine partial answers from classifiers to obtain final classification. We provide an empirical evaluation of the decomposition method in comparison to the standard Rough-Set rule induction method [12, 14] and the decision tree method. The experiments were carried out with the use of RSES-Lib software [1].

2 Preliminaries

In searching for concept approximation we are considering a special type of information systems — decision tables $\mathbb{A} = (U, A \cup \{d\})$, where $d : U \to V_d$ is a decision attribute. In a presence of missing data we may consider the attributes $a_i \in A$ as a functions $a_i : U \to V_i^*$, where $V_i^* = V_i \cup \{*\}$ and $* \notin V_i$. The special symbol "$*$" denotes absence of regular attribute value and if $a_i(x) = *$ we say that a_i is not defined on x. We can interpret $a_i : U \to V_i^*$ as a *partial* function in contrast to $a_i : U \to V_i$ interpreted as a *total* function.

To discover the knowledge hidden in data we can search for patterns of regularities in decision tables. We would like to focus here on searching for regularities that are based on the presence of missing attribute values. A standard tool for describing a data regularities are *templates* [10, 9]. The concept of template require some modification to be applicable to the problem of incomplete information table decomposition.

Definition 1. *Let $\mathbb{A} = (U, A \cup \{d\})$ be a decision table and let $a_i \in V_i$ be a* total descriptor*. An object $u \in U$ satisfies a total descriptor $a_i \in V_i$, if the value of the attribute $a_i \in A$ for this object u is not missing in $\mathbb{A}$, otherwise the object u does not satisfy total descriptor.*

Definition 2. *Let $\mathbb{A} = (U, A \cup \{d\})$ be a decision table. Any conjunction of total descriptors $(a_{k_1} \in V_{k_1}) \wedge \ldots \wedge (a_{k_n} \in V_{k_n})$ is called a* total template*. An object $u \in U$ satisfies total template $(a_{k_1} \in V_{k_1}) \wedge \ldots \wedge (a_{k_n} \in V_{k_n})$ if values of attributes $a_{k_1}, \ldots, a_{k_n} \in A$ for the object u are not missing in $\mathbb{A}$.*

Total templates are used to discover regular areas in data that contain no missing values. Once we have a total template, we can identify it with a subtable of original decision table. Such a subtable consists of the decision attribute, all attributes that are elements of total template and contains all objects that satisfy this template. Obviously, the decision table that corresponds to the total template contains no missing attribute values.

3 Method description

The decomposition method consist of two phases. In the first step the data decomposition is done. In the second step classifiers are induced and combined with a help of a conflict resolving method.

In the data decomposition phase the original decision table with missing attribute values is partitioned to a number of decision subtables with complete

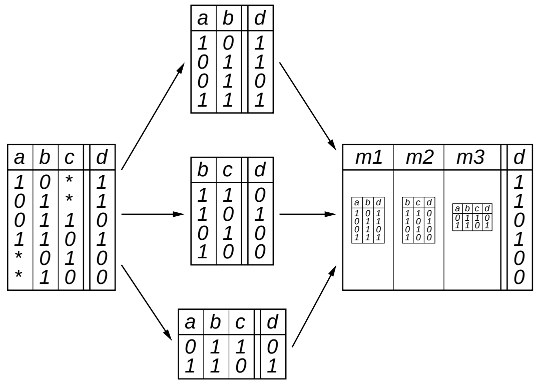

Fig. 1. The incomplete data is decomposed into complete subtables. Then, a conflict resolving method is applied.

object descriptions. Such a data decomposition should be done in accordance to regularities in real-world interest domain. We expect that the decomposition could reveal patterns of missing attribute values with a similar meaning for investigated real-world problem. The complete subtables that are result of the decomposition should correspond to natural subproblems of the whole problem domain.

With the assignment of total templates to complete decision subtables we can consider the data decomposition as a problem of covering data table with templates. The similar problem of data decomposition with the use of templates is described in [9, 10]. A standard approach to cover data table with templates is to iteratively generate the best template for objects that remains uncovered. The algorithm starts from the full set of objects. Than the (sub)optimal template is generated according to chosen criterion. In our experiments we used a dedicated, effective genetic algorithm to generate a sub-optimal template. All objects that satisfy the generated template are removed and the process is continued until the set of uncovered objects becomes empty. The set of templates generated by this algorithm covers all objects from original decision table. We can treat covering set of total templates as the result of decomposition.

Subsets of original decision table must meet some requirements in order to achieve good quality of inductive reasoning as well as to be applicable in case of methods that cannot deal with missing attribute values. We expect the decision subtables to exhaustively cover the input table. They should contain no missing attribute values. It is obvious that the quality of inductive reasoning depends on a particular partition and some partitions are better than others. We should construct the template evaluation criteria for templates defining decision subtables relevant to the approximated concept.

Once we have data decomposed into complete decision subtables we should merge partial classifiers to one global classifier. This is the second step of the decomposition method. Answer of classifiers induced from decision subtables are combined by a conflict resolving method. In presented experiments a rule induction method and a decision tree method based on discernibility measure were used to induce classifiers from the decision subtables. Those methods were chosen to be able to compare missing attribute values handling by ignoring discernibility on them with the decomposition method that does not relay on any other missing attribute values handling. The empirical evaluation provided by Grzymała-Busse in [4] showed that assuming indiscernibility of missing value with any other is a very effective mechanism for missing attribute values handling. To resolve conflicts between partial answers we used a simple voting mechanism and the decision tree metod. The initial experiments showed that application of voting for the decomposition method based on rule induction is not enough to achieve good results. Partially this is a consequence of *positive region* [6, 13] reduction in subtables of original data. The expressiveness of regular classifier makes it possible to combine partial answers induced from inconsistent decision subtables in much more sophisticated way.

Briefly we can summarize the decomposition method as follows:

1. Create a temporary set $\mathbb{T}$ of objects from the original decision table and repeat 2-3 until the temporary set $\mathbb{T}$ is empty;
2. Generate the best total template according to chosen criterion;
3. Remove objects from the temporary set $\mathbb{T}$ that are covered by generated template;
4. Create complete decision subtables that correspond to generated set of templates;
5. Induce classifiers over complete decision subtables;
6. Apply a conflict resolving method to get the final answer.

4 Decomposition criteria

Common approach to measure adequateness of a template for decomposition of a particular data set is to define a function q which describes overall quality of the investigated template. Then, the best template is understood as a template with the best value of such a quality function [9]. To achieve good results we should select quality function q very carefully and in accordance to nature of the optimized problem.

A standard approach to measure template quality is to define a quality function using *width* and *height* of a template [10, 9]. The *template height* is the number of objects that satisfy a template and the *template width* is the number of attributes that are elements of a template. To obtain a quality function q of a template we have to combine width and height to get one value. A usual formula that combines these two factors is

$$q = w \cdot h. \tag{1}$$

We can also add a simple mechanism to control the importance of each factor

$$q = w^{\alpha} \cdot h, \tag{2}$$

Table 1. Comparison of the number of subtables (templates) used in the decomposition method.

Table	-	$w \cdot h$	$w \cdot h \cdot G^1$	$w \cdot h \cdot G^8$	$w \cdot h \cdot H^1$	$w \cdot h \cdot H^8$	$w \cdot h \cdot P^1$	$w \cdot h \cdot P^8$
att	1.0	3.9	4.7	11.9	2.0	2.0	4.2	4.6
bld2	1.0	2.0	2.0	1.0	3.0	2.4	2.7	3.2
cmc2	1.0	2.0	5.0	7.0	1.0	1.0	2.4	3.3
ech	1.0	3.5	4.0	3.8	1.9	1.2	3.2	3.4
edu	1.0	3.0	4.0	8.0	1.1	3.5	2.9	6.4
hab2	1.0	3.9	5.0	5.0	2.0	2.0	2.9	2.8
hco	1.0	5.6	8.4	11.1	3.1	3.9	6.1	10.1
hep	1.0	3.9	3.9	3.9	3.0	2.7	4.0	4.4
hin	1.0	4.1	19.8	27.0	3.2	5.8	4.6	8.5
hur2	1.0	2.0	2.0	1.0	2.0	1.4	1.9	1.5
hyp	1.0	2.0	6.9	8.7	1.0	1.0	2.0	2.0
inf2	1.0	2.0	2.0	2.1	1.4	1.5	2.1	2.8
pid2	1.0	3.0	3.0	2.0	2.8	2.6	3.1	3.2
smo2	1.0	2.0	3.0	3.0	1.0	1.0	1.7	1.6

where $\alpha > 0$. If we apply $\alpha > 1$ the importance of the width, thus importance of the size of available object description, increases and the number of necessary templates to cover original decision table is higher. The empirical results showed, however, that α does not have significant impact on overall classification quality.

The quality function based only on width and height is not always enough to classify objects with the decomposition method better than by the method with native missing attribute values handling. The empirical evaluation demonstrated that in data exist many templates with similar width and height, but with different potential for the data decomposition.

We can estimate the template quality by evaluating homogeneity of the indiscernibility classes induced by this template. Such a measure should corresponds to the quality of the classification by prime implicants [6]. We used too measures for estimating the template quality:

$$G = \sum_{i=1}^{K} \frac{max_{c \in V_d} card(\{y \in [x^i]_{IND} : d(y) = c\})}{card([x^i]_{IND})}, \tag{3}$$

$$H = \frac{\sum_{i=1}^{K} max_{c \in V_d} card(\{y \in [x^i]_{IND} : d(y) = c\})}{K \cdot L}, \tag{4}$$

where K is the number of indiscernibility classes $[x^1]_{IND}, \ldots, [x^K]_{IND}$ and L is the size of subtable. We can easily incorporate these factors into the quality function

$$q = w^\alpha \cdot h \cdot G^\beta, \;\; q = w^\alpha \cdot h \cdot H^\beta, \tag{5}$$

where β controls the influence of factors to the whole quality value.

The second measure is similar to the *wrapper* approach in the feature selection [2]. Instead of estimating the template quality we can use the predictive

Table 2. Comparison of the decomposition method that uses various template evaluation criteria with the rule induction method.

Table	-	$w \cdot h$	$w \cdot h \cdot G^1$	$w \cdot h \cdot G^8$	$w \cdot h \cdot H^1$	$w \cdot h \cdot H^8$	$w \cdot h \cdot P^1$	$w \cdot h \cdot P^8$
att	53.00%	55.10%	54.70%	54.00%	50.40%	50.40%	54.10%	57.80%
bld2	57.97%	59.42%	59.42%	59.42%	58.26%	57.97%	61.45%	61.74%
cmc2	43.38%	47.52%	46.03%	44.47%	42.70%	42.70%	44.81%	46.57%
ech	25.19%	61.83%	61.07%	63.36%	62.60%	67.18%	62.60%	65.65%
edu	41.10%	53.60%	52.30%	51.50%	53.70%	53.90%	54.00%	52.10%
hab2	45.10%	63.40%	63.40%	63.40%	76.14%	76.14%	68.95%	70.59%
hco	80.16%	76.09%	75.00%	74.46%	63.86%	64.95%	71.20%	72.83%
hep	79.35%	72.90%	76.77%	74.19%	78.06%	79.35%	74.84%	78.71%
hin	53.70%	65.00%	61.50%	63.00%	58.40%	68.00%	64.90%	62.80%
hur2	52.63%	59.81%	59.81%	59.81%	59.81%	53.59%	58.37%	68.90%
hyp	97.60%	96.43%	96.62%	96.55%	95.23%	95.23%	96.52%	97.00%
inf2	64.71%	45.38%	44.96%	45.38%	42.02%	38.24%	52.52%	58.40%
pid2	63.02%	64.71%	65.10%	60.16%	64.45%	65.49%	64.71%	67.19%
smo2	52.40%	62.21%	58.39%	58.42%	69.53%	69.53%	62.80%	66.69%

accuracy of the data subset. The classifier itself is executed on decision subtable determined by the total template and the number of correct answers is counted.

$$P = \frac{\textit{number of correct answers}}{\textit{number of objects}}. \tag{6}$$

Also this factor can be easily incorporated into the quality function

$$q = w^{\alpha} \cdot h \cdot P^{\beta}, \tag{7}$$

where $\beta > 0$ controls the influence of predictive accuracy factor to the whole quality value.

5 Empirical Evaluation

There were carried out some experiments in order to evaluate the decomposition method with various template evaluation functions. A genetic algorithm was used for generation of the best template with respect to the selected decomposition criterion. Results were obtained from the ten-fold Cross-Validation (CV10) evaluation. The rule induction method and the decision tree method were used as a classifier. The experiments were performed with different decomposition approaches as well as without using decomposition method at all. Data sets from *Recursive-Partitioning.com* [8] were used for evaluation of the decomposition method. Data sets contain missing values in the range from 10% to 100.0% of all values in data.

Table 1 presents the average number of generated templates. The number of templates corresponds to the number of subclassifiers being result of the data decomposition. As we can see there are no strong general correlation between

Table 3. Comparison of the decomposition method that use various template evaluation criteria with the decision tree method.

Table	-	$w \cdot h$	$w \cdot h \cdot G^1$	$w \cdot h \cdot G^8$	$w \cdot h \cdot H^1$	$w \cdot h \cdot H^8$	$w \cdot h \cdot P^1$	$w \cdot h \cdot P^8$
att	56.80%	58.00%	53.50%	54.30%	60.00%	59.40%	60.40%	66.70%
bld2	59.71%	56.23%	56.23%	61.16%	52.75%	50.43%	64.35%	72.17%
cmc2	42.36%	43.65%	45.76%	46.03%	41.75%	41.82%	47.39%	56.08%
ech	63.36%	64.12%	62.60%	65.65%	65.65%	64.12%	66.41%	66.41%
edu	45.60%	52.20%	51.50%	54.80%	54.00%	53.20%	51.60%	55.30%
hab2	61.11%	74.18%	68.30%	68.30%	73.20%	73.20%	72.55%	73.86%
hco	77.45%	80.43%	82.61%	81.25%	67.12%	67.39%	83.42%	86.96%
hep	66.45%	72.26%	70.97%	73.55%	79.35%	73.55%	75.48%	83.23%
hin	67.60%	68.70%	72.40%	72.80%	62.40%	70.10%	69.70%	74.30%
hur2	52.15%	52.15%	52.15%	51.20%	52.15%	46.41%	56.94%	61.72%
hyp	96.68%	95.51%	95.26%	96.24%	95.23%	95.23%	95.98%	96.49%
inf2	55.88%	55.04%	55.04%	55.88%	36.13%	35.29%	63.03%	81.51%
pid2	59.77%	60.55%	59.38%	59.11%	62.11%	63.15%	60.68%	66.02%
smo2	56.08%	61.96%	59.47%	59.47%	69.53%	69.53%	67.71%	69.60%

the number of templates and classifier accuracy. For some data sets the better classification is related to the increase of the number of templates while for the other data sets better accuracy is achieved without any increase of the number of templates.

Tables 2 and 3 present the results of the decomposition method. In the first column there are the results of classification without the decomposition method method. In the following columns the results of the decomposition method are presented with various template quality function described in the header of each column. The table 2 presents the results of the rule induction based method with the decision tree used for conflict resolving. In the table 3 there are the results of the decision tree based method with the simple voting mechanism used for conflict resolving. The decision tree utilized for conflict resolving and partial classifier induction uses conflict measure for evaluation of the tests in tree nodes.

The decomposition method performs better than the both methods, especially when the predictive quality is included in the template quality function. We should consider that evaluation of the predictive quality is very time-consuming, even in spite of partial result caching and other optimizations. The indiscernibility based functions G and H are much more easy to compute, however, the results not always overcome these obtained without the decomposition method.

Table 4 presents the average number of induced rules[1] and table 5 presents the average number of leafs in induced decision tree. The number of induced rules and leafs can be interpreted as a size of the model describing the hypothesis. We expect the decomposition done in according to data regularities to decrease

[1] For inconsistent decision tables the number of induced rules is 0 and the decision is made only by the conflict resolving method.

Table 4. Comparison of the number of induced rules induced with various template evaluators and without the decomposition method at all.

Table	-	$w \cdot h$	$w \cdot h \cdot G^1$	$w \cdot h \cdot G^8$	$w \cdot h \cdot H^1$	$w \cdot h \cdot H^8$	$w \cdot h \cdot P^1$	$w \cdot h \cdot P^8$
att	3154.1	5922.2	5192.2	13615.2	0.0	0.0	6426.8	3654.0
bld2	2443.5	2780.9	2780.9	618.9	2658.8	609.8	3396.7	2438.8
cmc2	6134.0	5839.8	10173.6	16077.1	0.0	0.0	3596.2	1621.2
ech	543.7	786.0	889.4	577.8	341.1	76.9	634.8	508.6
edu	5552.5	4282.3	6412.2	10615.0	0.8	271.6	3476.6	4039.7
hab2	259.8	491.8	722.7	722.7	29.8	29.8	237.8	149.2
hco	12840.1	4515.4	2987.0	3792.8	1261.5	1882.8	4583.1	6282.2
hep	4656.9	2432.5	2332.3	1495.4	1476.2	1289.0	2178.6	1642.9
hin	736.0	316.6	2898.3	4852.0	7.9	14.2	461.3	514.2
hur2	1148.0	1298.1	1298.1	504.2	1298.1	409.6	1134.3	451.3
hyp	14900.6	1692.4	4552.7	6046.2	2.5	1.6	1810.3	1473.2
inf2	5629.0	6725.2	6725.2	7181.8	3.6	2.4	4217.3	2012.8
pid2	5951.7	10521.4	10333.6	4987.5	7197.3	5893.1	10533.9	5921.3
smo2	10728.4	11881.1	12479.7	12479.7	0.0	0.0	3179.0	98.1

complexity of model induction — thus its size. We should take into consideration that in the decomposition method one object is usually covered by more than one template. This could potentially yield to the great increase of the rule number with use of the decomposition method. As we can observe the number of rules induced with the decomposition method is frequently smaller than the number of rules induced without the decomposition at all.

6 Conclusions

The decomposition method turned out to be an efficient tool for adapting existing methods to deal with missing attribute values in decision tables. It can be applied to various algorithms for classifier induction to enrich them with capabilities of incomplete information systems processing. The time-consuming predictive quality evaluation can be replaced with easier to compute measures of the template quality. The application of the rule-based inductive learning demonstrated that decomposition can simplify the model describing induced hypothesis. The further research will focus on application of rule-based inductive learning with uniform conflict resolving method at the subtables and the whole system level. We believe that decomposition done in accordance to the natural structure of analyzed data can result in classifier close to the common sense reasoning.

Acknowledgments

I wish to thank professor Andrzej Skowron for a great support while writing this paper. This work was partially supported by the Polish State Committee for Scientific Research grant No. 8T11C02519.

Table 5. Comparison of the number of leafs in induced decision trees with various template evaluators and without the decomposition method at all.

Table	-	$w \cdot h$	$w \cdot h \cdot G^1$	$w \cdot h \cdot G^8$	$w \cdot h \cdot H^1$	$w \cdot h \cdot H^8$	$w \cdot h \cdot P^1$	$w \cdot h \cdot P^8$
att	535.5	788.5	1025.0	2712.3	3.0	3.4	859.6	692.5
bld2	21.2	50.1	50.1	34.3	51.4	40.9	66.6	112.8
cmc2	425.9	503.0	1813.6	2616.5	2.0	2.0	607.3	640.7
ech	4.8	19.9	22.1	23.6	4.5	2.6	21.4	27.5
edu	208.3	406.6	641.3	1446.9	2.7	9.9	393.4	353.5
hab2	42.6	109.3	176.5	176.5	30.9	30.9	74.7	70.7
hco	17.3	86.0	201.9	277.7	6.9	8.6	77.1	155.1
hep	6.8	45.9	50.2	56.1	11.6	5.9	53.5	70.3
hin	536.7	259.3	1931.7	2997.1	11.7	23.7	350.1	488.5
hur2	4.0	7.2	7.2	4.4	7.2	34.9	7.3	16.0
hyp	8.3	38.2	601.7	643.3	2.0	2.0	36.3	44.9
inf2	81.6	142.9	142.9	148.5	3.4	2.5	146.6	168.4
pid2	10.2	24.4	28.9	20.1	31.8	31.8	32.6	109.0
smo2	646.0	1240.5	1978.9	1978.9	2.0	2.0	530.6	151.3

References

1. J. G. Bazan and M. S. Szczuka. RSES and RSESlib — a collection of tools for rough set computations. In Ziarko and Yao [16], pages 106–113.
2. M. Dash and H. Liu. Feature selection for classification. *Intelligent Data Analysis*, 1(3), 1997.
3. Y. Fujikawa and T. B. Ho. Scalable algorithms for dealing with missing values. 2001.
4. J. W. Grzymała-Busse and M. Hu. A comparison of several approaches to missing attribute values in data mining. In Ziarko and Yao [16], pages 180–187.
5. J. W. Grzymała-Busse and A. Y. Wang. Modified algorithms LEM1 and LEM2 for rule induction from data with missing attribute values. In *Proceedings of 5th Workshop on Rough Sets and Soft Computing (RSSC'97) at the 3rd Joint Conference on Information Sciences*, pages 69–72, Research Triangle Park (NC, USA), 1997.
6. J. Komorowski, Z. Pawlak, L. Polkowski, and A. Skowron. Rough sets: A tutorial. In S. K. Pal and A. Skowron, editors, *Rough Fuzzy Hybridization. A New Trend in Decision Making*, pages 3–98, Singapore, 1999. Springer.
7. M. Kryszkiewicz. Properties of incomplete information systems in the framework of rough sets. In Polkowski and Skowron [11], pages 422–450.
8. T. Lim. *Missing covariate values and classification trees.* http://www.recursive-partitioning.com/mv.shtml, Recursive-Partitioning.com, 2000.
9. S. H. Nguyen. *Regularity Analysis and its Application in Data Mining.* PhD thesis, Warsaw University, Faculty of Mathematics, Computer Science and Mechanics, 1999.
10. S. H. Nguyen, A. Skowron, and P. Synak. Discovery of data patterns with applications to decomposition and classification problems. In L. Polkowski and A. Skowron, editors, *Rough Sets in Knowledge Discovery 2: Applications, Case Studies and Software Systems*, pages 55–97. Physica-Verlag, 1998.

11. L. Polkowski and A. Skowron, editors. *Rough Sets in Knowledge Discovery 1: Methodology and Applications*. Physica-Verlag, 1998.
12. A. Skowron. Synthesis of adaptive decision systems from experimental data. In A. Aamodt and J. H. Komorowski, editors, *SCAI 1995*, pages 220–238. IOS Press, 1995.
13. A. Skowron and C. Rauszer. The discernibility matrices and functions in information systems. In R. Słowiński, editor, *Intelligent Decision Support. Handbook of Applications and Advances in Rough Sets Theory*, pages 331–362, Dordrecht, 1992. Kluwer.
14. J. Stefanowski. On rough set based approaches to induction of decision rules. In Polkowski and Skowron [11], pages 500–529.
15. J. Stefanowski and A. Tsoukiàs. Incomplete information tables and rough classification. *International Journal of Computational Intelligence*, 17(3):545–566, August 2001.
16. W. Ziarko and Y. Y. Yao, editors. *Proceedings of 2nd International Conference on Rough Sets and Current Trends in Computing, RSCTC-2000*, volume 2005 of *LNAI*. Springer, 2000.