Construction of Tests for Tables with Many-Valued Decisions

Mikhail Moshkov[1] and Beata Zielosko[1,2]

[1] Mathematical and Computer Sciences & Engineering Division
King Abdullah University of Science and Technology
Thuwal 23955-6900, Saudi Arabia
{mikhail.moshkov,beata.zielosko}@kaust.edu.sa
[2] Institute of Computer Science, University of Silesia
39, Będzińska St., 41-200 Sosnowiec, Poland

Abstract. In the paper, authors present a greedy algorithm for construction of exact and partial tests (super-reducts) for decision tables with many-valued decisions. Exact tests can be over-fitted, so instead of them it is often more appropriate to work with partial tests with smaller number of attributes. Based on results for set cover problem authors study bounds on accuracy of greedy algorithm for exact and partial test construction, and complexity of the problem of test cardinality minimization.

Key words: decision table with many-valued decisions, tests, reducts, greedy algorithm

1 Introduction

In real life applications we often meet with decision tables with many-valued decisions, mainly if we work with experimental or statistical data. In such datasets we can have several rows (objects) with equal values of conditional attributes and different values of the decision attribute. Instead of a group of rows, we can consider one row given by values of conditional attributes. We attach to this row a set of decisions: either all decisions for rows from the group, or k the most frequent decisions for objects from the group, etc. As a result we obtain a decision table with many-valued decisions. So, in the case of decision table with many-valued decisions each row is labeled with a nonempty, finite set of decisions, and we should find a decision from the set of decisions attached to this row.

In the rough set theory [4, 5] decision tables are considered often that have equal rows labeled with different decisions. The set of decisions attached to equal rows is called the *generalized decision* for each of these equal rows. The usual way is to find for a given row its generalized decision. However, the problem of finding an arbitrary decision or one of the most frequent decisions from the generalized decision is interesting also.

CONCURRENCY, SPECIFICATION AND PROGRAMMING
M. Szczuka et al. (eds.): Proceedings of the international workshop CS&P 2011
September 28-30, Pułtusk, Poland, pp. 376-384

Tests (super-reducts) and reducts can be considered as a way of knowledge representation. In applications we often deal with decision tables which contain noisy data. In this case, exact reducts can be over-fitted, i.e., they depend essentially on the noise. So, instead of exact reducts with many attributes, it is more appropriate to work with partial reducts with smaller number of attributes.

The problem of construction of test with minimum cardinality is NP-hard. Therefore, we consider approximate polynomial algorithms for test minimization.

In this paper, we consider only binary decision tables with many-valued decisions. However, the obtained results can be extended to the decision tables filled by numbers from the set $\{0, \ldots, k-1\}$, where $k \geq 3$.

This paper consists of six sections. In Sect. 2, main notions are discussed. In Sect. 3, set cover problem, greedy algorithm and bounds on accuracy of greedy algorithm for exact and partial covers are presented. Section 4 contains bounds on accuracy of greedy algorithm for exact and partial tests. In Sect. 5, we discuss an example. Section 6 contains conclusions.

2 Main Notions

In this section, we consider definitions of notions corresponding to decision tables with many-valued decisions.

A *(binary) decision table with many-valued decisions* is a rectangular table T filled with numbers from the set $\{0, 1\}$. Columns of this table are labeled with attributes $f_1, \ldots, f_n$. Rows of the table are pairwise different, and each row is labeled with a nonempty finite set of decisions (natural numbers). Note that each (binary) decision table with one-valued decisions can be interpreted also as a decision table with many-valued decisions. In such table, each row is labeled with a set of decisions which has one element.

We will say that T is a *degenerate* table if either T has no rows, or the intersection of sets of decisions attached to rows of T is nonempty.

A decision which belongs to the maximum number of sets of decisions attached to rows in T is called the *most common decision for* T. If we have more than one such decision we choose the minimum one. If T is empty then 1 is the most common decision for T.

A table obtained from T by removal of some rows is called a *subtable* of T. A subtable T' of T is called *boundary* subtable if T' is not degenerate but each proper subtable of T' is degenerate. We denote by $B(T)$ the number of boundary subtables of the table T. It is clear that T is a degenerate table if and only if $B(T) = 0$. The value $B(T)$ will be interpreted as *uncertainty* of T.

We will say that attribute f_i *divides* boundary subtable Θ if and only if this attribute is not constant on the rows of Θ (for example, for binary decision table at the intersection with the column f_i we can find some rows which contain 1 and some rows which contain 0).

A *test for the table* T (super-reduct) is a subset of columns $\{f_{i_1}, \ldots, f_{i_m}\}$ such that these attributes divide each boundary subtable. Empty set is a test for T iff T is a degenerate table.

A reduct for the table T is a test for T for which each proper subset is not a test. It is clear that each test has a reduct as a subset. We denote by $R(T)$ the minimum cardinality of a reduct for T.

Let us define the notion of α-test for the table T. Let α be a real number such that $0 \le \alpha < 1$.

An *α-test for the table* T is a subset of attributes $\{f_{i_1}, \ldots, f_{i_m}\}$ such that these attributes divide at least $(1-\alpha)B(T)$ boundary subtables. Empty set is an α-test for T iff T is a degenerate table. An *α-reduct for the table* T is an α-test for T for which each proper subset is not an α-test. We denote by $R_\alpha(T)$ the minimum cardinality of an α-test for the table T. It is clear that each α-test has an α-reduct as a subset. Therefore $R_\alpha(T)$ is the minimum cardinality of an α-reduct. It is clear also that the set of tests for the table T coincides with the set of 0-tests for T. Therefore $R_0(T) = R(T)$.

3 Set Cover Problem

In this section, we present two variants of set cover problem: problem of construction of minimum exact cover (Section 3.1) and problem of construction of minimum partial cover (α-cover, Section 3.2). We consider bounds on accuracy of greedy algorithm for both variants of set cover problem.

3.1 Exact Covers

Let A be a set containing $N > 0$ elements, and $F = \{S_1, \ldots, S_p\}$ be a family of subsets of the set A such that $A = \bigcup_{i=1}^{p} S_i$. A subfamily $\{S_{i_1}, \ldots, S_{i_t}\}$ of the family F will be called a *cover* if $\bigcup_{j=1}^{t} S_{i_j} = A$. The problem of searching for a cover with minimum cardinality t is called the *set cover problem*. It is well known that this problem is an NP-hard problem.

U. Feige [2] proved that if $NP \not\subseteq DTIME(n^{O(\log\log n)})$ then for any ε, $0 < \varepsilon < 1$, there is no polynomial algorithm that constructs a cover whose cardinality is at most $(1-\varepsilon)C_{\min} \ln N$, where $C_{\min}$ is the minimum cardinality of a cover.

Now, we present well known *greedy* algorithm for the set cover problem.

Set $B := A$, and $COVER := \emptyset$.

(*) In the family F we find a set S_i with minimum index i such that

$$|S_i \cap B| = \max\{|S_j \cap B| : S_j \in F\}.$$

Then we set $B := B \setminus S_i$ and $COVER := COVER \cup \{S_i\}$. If $B = \emptyset$ then we finish the work of the algorithm. The set $COVER$ is the result of the algorithm work. If $B \neq \emptyset$ then we return to the label (*).

We denote by C_{greedy} the cardinality of cover constructed by the greedy algorithm. Remind that $C_{\min}$ is the minimum cardinality of a cover.

We will present the following well known result with our own proof.

Theorem 1. $C_{\text{greedy}} \leq C_{\min} \ln N + 1$.

Proof. Denote $m = C_{\min}$. If $m = 1$ then, as it is not difficult to show, $C_{\text{greedy}} = 1$, and the considered inequality holds. Let $m \geq 2$. Let S_i be a subset of maximum cardinality in the family F. It is clear that $|S_i| \geq N/m$ (otherwise, $C_{\min} > m$ which is impossible). So, after the first step we will have at most $N - N/m = N(1-1/m)$ uncovered elements in the set A. After the first step we will have the following set cover problem: the set $A \setminus S_i$ and the family $\{S_1 \setminus S_i, \ldots, S_p \setminus S_i\}$. For this problem, the minimum cardinality of a cover is at most m. So, after the second step, when we choose a set $S_j \setminus S_i$ with maximum cardinality, the number of uncovered elements in the set A will be at most $N(1-1/m)^2$, etc.

Let the greedy algorithm in the process of cover construction make g steps and construct a cover of cardinality g. Then after the step number $g-1$ we have at least one uncovered element in the set A. Therefore $N(1-1/m)^{g-1} \geq 1$ and $N \geq (1 + 1/(m-1))^{g-1}$. If we take the natural logarithm of both sides of this inequality we obtain $\ln N \geq (g-1)\ln(1 + 1/(m-1))$.

It is known that for any natural p the inequality $\ln(1+1/p) > 1/(p+1)$ holds. Therefore $\ln N > (g-1)/m$ and $g < m \ln N + 1$. Taking into account that $m = C_{\min}$ and $g = C_{\text{greedy}}$ we obtain $C_{\text{greedy}} < C_{\min} \ln N + 1$. □

Using the mentioned result of U. Feige [2] we obtain that if

$$NP \not\subseteq DTIME(n^{O(\log \log n)})$$

then the greedy algorithm is close to the best (from the point of view of accuracy) approximate polynomial algorithms for solving the set cover problem.

3.2 Partial Covers

Let α be a real number such that $0 \leq \alpha < 1$. Let A be a set containing $N > 0$ elements, and $F = \{S_1, \ldots, S_p\}$ be a family of subsets of the set A such that $A = \bigcup_{i=1}^{p} S_i$. A subfamily $\{S_{i_1}, \ldots, S_{i_t}\}$ of the family F will be called an *α-cover for A, F* if $|\bigcup_{j=1}^{t} S_{i_j}| \geq (1-\alpha)|A|$. The problem of searching for an α-cover with minimum cardinality is NP-hard [6].

We consider a greedy algorithm for construction of α-cover. During each step this algorithm chooses a subset from F which covers the maximum number of uncovered elements from A. This algorithm stops when the constructed subfamily is an α-cover for A, F. We denote by $C_{\text{greedy}}(\alpha)$ the cardinality of constructed α-cover, and by $C_{\min}(\alpha)$ we denote the minimum cardinality of α-cover for A, F. The following statement was obtained by J. Cheriyan and R. Ravi in [1]. We present it with our own proof.

Theorem 2. *Let* $0 < \alpha < 1$. *Then* $C_{\text{greedy}}(\alpha) < C_{\min}(0)\ln(1/\alpha) + 1$.

Proof. Denote $m = C_{\min}(0)$. If $m = 1$ then, as it is not difficult to show, $C_{\text{greedy}}(\alpha) = 1$ and the considered inequality holds. Let $m \geq 2$ and S_i be a subset of maximum cardinality in F. It is clear that $|S_i| \geq N/m$. So, after the first step we will have at most $N - N/m = N(1 - 1/m)$ uncovered elements in the set A. After the first step we have the following set cover problem: the set $A \setminus S_i$ and the family $\{S_1 \setminus S_i, \ldots, S_p \setminus S_i\}$. For this problem, the minimum cardinality of a cover is at most m. So, after the second step, when we choose a set $S_j \setminus S_i$ with maximum cardinality, the number of uncovered elements in the set A will be at most $N(1 - 1/m)^2$, etc.

Let the greedy algorithm in the process of α-cover construction make g steps and construct an α-cover of cardinality g. Then after the step number $g - 1$ more then αN elements in A are uncovered. Therefore $N(1 - 1/m)^{g-1} > \alpha N$ and $1/\alpha > (1 + 1/(m-1))^{g-1}$. If we take the natural logarithm of both sides of this inequality we obtain $\ln 1/\alpha > (g-1)\ln(1 + 1/(m-1))$. It is known that for any natural p, the inequality $\ln(1 + 1/p) > 1/(p+1)$ holds. Therefore $\ln(1/\alpha) > (g-1)/m$ and $g < m \ln(1/\alpha) + 1$. Taking into account that $m = C_{\min}(0)$ and $g = C_{\text{greedy}}(\alpha)$, we obtain $C_{\text{greedy}}(\alpha) < C_{\min}(0)\ln(1/\alpha) + 1$. □

4 Tests and Reducts

We consider here the problem of minimization of test cardinality: for a given decision table T with many-valued decisions we need to construct a test (an α-test) with minimum cardinality.

Based on results for the set cover problem we study bounds on accuracy of the greedy algorithms for test construction, and complexity of the problem of minimization of test cardinality.

Each of the considered algorithms contains two parts:

1. Construction of all boundary subtables,
2. Construction of a test or an α-test.

In Sect. 4.1, we present some auxiliary statements connected with construction of boundary subtables for decision table T with many-valued decisions.

In Sects. 4.2 and 4.3, we apply the greedy algorithms for both variants of set cover problem to construct exact (0-tests) and partial tests (α-tests).

4.1 Auxiliary Statements

We denote by $Tab(t)$, where t is a natural number, the set of decision tables with many-valued decisions such that each row in the table has at most t decisions (is labeled with a set of decisions which cardinality is at most t).

Lemma 1. *Let T' be a boundary subtable with m rows. Then each row of T' is labeled with a set of decisions which cardinality is at least $m - 1$.*

Proof. Let rows of T' be labeled with sets of decisions $D_1, \ldots, D_m$ respectively. Then $D_1 \cap \ldots \cap D_m = \emptyset$ and for any $i \in \{1, \ldots, m\}$, the set $D_1 \cap \ldots \cap D_{i-1} \cap D_{i+1} \cap \ldots \cap D_m$ contains a number d_i. Assume that $i \neq j$ and $d_i = d_j$. Then $D_1 \cap \ldots \cap D_m \neq \emptyset$ which is impossible. Therefore $d_1, \ldots, d_m$ are pairwise different numbers. It is clear that for $i = 1, \ldots, m$, the set $\{d_1, \ldots, d_m\} \setminus \{d_i\}$ is a subset of the set D_i. □

Corollary 1. *Each boundary subtable of a table $T \in Tab(t)$ has at most $t+1$ rows.*

Therefore, for tables from $Tab(t)$, there exists a polynomial algorithm for the computation of parameter $B(T)$. For example, for any decision table T with one-valued decisions (really, for any table from $Tab(1)$) the equality $B(T) = P(T)$ holds, where $P(T)$ is the number of unordered pairs of rows of T with different decisions.

4.2 Exact Tests

We can apply the greedy algorithm for set cover problem to construct tests for decision tables with many-valued decisions. Let T contain n columns labeled with attributes $f_1, \ldots, f_n$. We construct a set cover problem $A(T)$, $F(T)$ corresponding to the table T, where $A(T)$ is the set of all boundary subtables of T, $F(T) = \{S_1, \ldots, S_n\}$, and, for $i = 1, \ldots, n$, S_i is the set of boundary subtables from $A(T)$ divided by f_i. One can show that $\{f_{i_1}, \ldots, f_{i_m}\}$ is a test for T if and only if $\{S_{i_1}, \ldots, S_{i_m}\}$ is a cover for $A(T)$, $F(T)$. Let us apply the greedy algorithm for set cover problem to $A(T)$, $F(T)$. As a result, we obtain a cover corresponding to a test for T. This test is a result of the considered algorithm work. We denote by $R_{\mathrm{greedy}}(T)$ the cardinality of the constructed test.

Theorem 3. *Let T be a decision table with many-valued decisions. Then*

$$R_{\mathrm{greedy}}(T) \le R(T) \ln B(T) + 1 .$$

This result follows immediately from the description of the considered algorithm and from Theorem 1.

For any natural t, for tables from the class $Tab(t)$ the considered algorithm has polynomial time complexity.

The next two statements follow immediately from similar ones obtained in [3] for decision tables with one-valued decisions.

Proposition 1. *The problem of minimization of test cardinality for decision tables with many-valued decisions is NP-hard.*

Theorem 4. *If $NP \not\subseteq DTIME(n^{O(\log \log n)})$ then for any ε, $0 < \varepsilon < 1$, there is no polynomial algorithm which for a given decision table T with many-valued decisions constructs a test for T which cardinality is at most*

$$(1 - \varepsilon) R(T) \ln B(T) .$$

The comparison of Theorems 4 and 3 shows that under the assumption $NP \notin DTIME(n^{O(\log \log n)})$ the greedy algorithm is close to the best (from the point of view of accuracy) approximate polynomial algorithms for minimization of test cardinality.

4.3 Partial Tests

We use the greedy algorithm for construction of α-covers to construct α-tests. Let T be a table with many-valued decisions containing n columns labeled with attributes $f_1, \dots, f_n$. Let α be a real number such that $0 < \alpha < 1$, $B(T)$ is the number of boundary subtables of the table T.

We consider a set cover problem $A(T)$, $F(T) = \{S_1, \dots, S_n\}$ where $A(T)$ is the set of all boundary subtables of the table T and $|A(T)| = B(T)$. For $i = 1, \dots, n$, the set S_i is the set of boundary subtables from $A(T)$ divided by f_i. One can show that $\{f_{i_1}, \dots, f_{i_m}\}$ is an α-test for T if and only if $\{S_{i_1}, \dots, S_{i_m}\}$ is an α-cover for $A(T)$, $F(T)$. Evidently, for the considered set cover problem $C_{\min}(0) = R(T)$, where $R(T)$ is the minimum cardinality of 0-test for T.

Let us apply the greedy algorithm to the considered set cover problem. This algorithm constructs an α-cover which corresponds to an α-test for the decision table T. From Theorem 2 it follows that the cardinality of this test is at most

$$R(T)\ln(1/\alpha) + 1.$$

We denote by $R_{\text{greedy}}(\alpha, T)$ the cardinality of the test constructed by the following polynomial algorithm: for a given α, $0 < \alpha < 1$, decision table T, we construct the set cover problem $A(T)$, $F(T)$ and then apply to this problem the greedy algorithm for construction of α-cover. We transform the obtained α-cover into an α-test. According to what has been said above we have the following statement.

Theorem 5. *Let T be a nondegenerate decision table with many-valued decisions and α be a real number such that $0 < \alpha < 1$. Then*

$$R_{\text{greedy}}(\alpha, T) \le R(T)\ln(1/\alpha) + 1.$$

Let us show that the problem of minimization of α-test cardinality is NP-hard.

Let us consider a set cover problem A, F where $A = \{a_1, \dots, a_N\}$ and $F = \{S_1, \dots, S_m\}$. We define a decision table $T(A, F)$. This table has m columns corresponding to the sets $S_1, \dots, S_m$ respectively, and $N + 1$ rows. For $j = 1, \dots, N$, the j-th row corresponds to the element a_j. The last $(N+1)$-th row is filled by 0. For $j = 1, \dots, N$ and $i = 1, \dots, m$, at the intersection of j-th row and i-th column 1 stays if and only if $a_j \in S_i$. The set of decisions corresponding to the last row is equal to $\{2\}$. All other rows are labeled with the set of decisions $\{1\}$.

One can show that a subfamily $\{S_{i_1}, \dots, S_{i_t}\}$ is an α-cover for A, F, $0 \le \alpha < 1$, if and only if the set of attributes $\{f_{i_1}, \dots, f_{i_m}\}$ is an α-test for $T(A, F)$.

So, we have a polynomial time reduction of the problem of minimization of α-cover cardinality to the problem of minimization of α-test cardinality for decision tables with many-valued decisions. Since the first problem is NP-hard [6], we have

Proposition 2. *For any α, $0 \le \alpha < 1$, the problem of minimization of α-test cardinality for decision tables with many-valued decisions is NP-hard.*

5 Example

Let we have a finite set $S = \{(a_1, b_1), \ldots, (a_n, b_n)\}$ of points in the plane and a mapping μ which corresponds to each point (a_p, b_p) a nonempty subset $\mu(a_p, b_p)$ of the set $\{green, yellow, red\}$. Colors are interpreted as decisions, and for each point from S we need to find a decision (color) from the set of decisions attached to this point. We denote this problem by (S, μ).

For solving the problem (S, μ), we use attributes corresponding to straight lines which are given by equations of the kind $x = \beta$ or $y = \gamma$. These attributes are defined on the set S and take values from the set $\{0, 1\}$. Consider the line given by equation $x = \beta$. Then the value of the corresponding attribute is equal to 0 on a point $(a, b) \in S$ if and only if $a < \beta$. Consider the line given by equation $y = \gamma$. Then the value of the corresponding attribute is equal to 0 if and only if $b < \gamma$.

We now choose a finite set of straight lines which allow us to construct a test with the minimum cardinality for the problem (S, μ). It is possible that $a_i = a_j$ or $b_i = b_j$ for different i and j. Let $a_{i_1}, \ldots, a_{i_m}$ be all pairwise different numbers from the set $\{a_1, \ldots, a_n\}$ which are ordered such that $a_{i_1} < \ldots < a_{i_m}$. Let $b_{j_1}, \ldots, b_{j_t}$ be all pairwise different numbers from the set $\{b_1, \ldots, b_n\}$ which are ordered such that $b_{j_1} < \ldots < b_{j_t}$.

One can show that there exists a test with minimum cardinality which use only attributes corresponding to the straight lines defined by equations $x = a_{i_1} - 1$, $x = (a_{i_1} + a_{i_2})/2, \ldots$, $x = (a_{i_{m-1}} + a_{i_m})/2$, $x = a_{i_m} + 1$, $y = b_{j_1} - 1$, $y = (b_{j_1} + b_{j_2})/2, \ldots$, $y = (b_{j_{t-1}} + b_{j_t})/2$, $y = b_{j_t} + 1$.

We now describe a decision table $T(S, \mu)$ with $m+t+2$ columns and n rows. Columns of this table are labeled with attributes $f_1, \ldots, f_{m+t+2}$, corresponding to the considered $m + t + 2$ lines. Attributes $f_1, \ldots, f_{m+1}$ correspond to lines defined by equations $x = a_{i_1} - 1, x = (a_{i_1} + a_{i_2})/2, \ldots, x = (a_{i_{m-1}} + a_{i_m})/2, x = a_{i_m} + 1$ respectively. Attributes $f_{m+2}, \ldots, f_{m+t+2}$ correspond to lines defined by equations $y = b_{j_1} - 1$, $y = (b_{j_1} + b_{j_2})/2, \ldots, y = (b_{j_{t-1}} + b_{j_t})/2, y = b_{j_t} + 1$ respectively. Rows of the table $T(S, \mu)$ correspond to points $(a_1, b_1), \ldots, (a_n, b_n)$. At the intersection of the column f_l and row (a_p, b_p) the value $f_l(a_p, b_p)$ stays. For $p = 1, \ldots, n$, the row (a_p, b_p) is labeled with the set of decisions $\mu(a_p, b_p)$.

From Corollary 1 it follows that each boundary subtable of the table $T(S, \mu)$ has at most four rows. Thus, the following statement holds:

Proposition 3. $B(T(S, \mu)) \leq |S|^4$.

In fact, each boundary subtable of the table $T(S, \mu)$ has at most three rows. One can show that there are only two types of boundary subtables of the table $T(S, \mu)$:

- With two rows labeled with disjoint sets of decisions, for example, $\{yellow\}$ and $\{green, red\}$, or $\{yellow\}$ and $\{green\}$.
- With three rows labeled with sets of decisions $\{green, yellow\}$, $\{green, red\}$, and $\{yellow, red\}$.

Using this fact we can easily compute the value $B(T(S,\mu))$:

$$B(T(S,\mu)) = N(g)N(y) + N(g)N(r) + N(y)N(r) + N(g)N(y,r) \\ +N(y)N(g,r) + N(r)N(g,y) + N(g,y)N(g,r)N(y,r),$$

where $N(g)$ is the number of rows in $T(S,\mu)$ labeled with the set of decisions $\{green\}$, $N(g,r)$ is the number of rows in $T(S,\mu)$ labeled with the set of decisions $\{green, red\}$, etc.

Using this fact we can also easily find all boundary subtables of the table $T(S,\mu)$.

6 Conclusions

We studied algorithms for exact and partial test construction for decision tables with many-valued decisions. First part of each algorithm is connected with the construction of boundary subtables, the second one is a greedy algorithm for test construction.

Based on results for the set cover problem we obtained bounds on accuracy of the greedy algorithms for exact and partial tests, and complexity of the problems of test cardinality minimization.

We studied binary decision tables with many-valued decisions but the considered approach can be used also for decision tables with more than two values of attributes.

Acknowledgements

The authors are grateful to the anonymous reviewer for helpful comments.

References

1. Cheriyan, J., Ravi, R.: Lecture notes on approximation algorithms for network problems. `http://www.math.uwaterloo.ca/jcheriya/lecnotes.html` (1998)
2. Feige, U.: A threshold of $\ln n$ for approximating set cover (Preliminary version). In: 28th Annual ACM Symposium on the Theory of Computing, pp. 314–318. ACM Press, New York (1996)
3. Moshkov, M., Piliszczuk, M., Zielosko, B.: Partial Covers, Reducts and Decision Rules in Rough Sets: Theory and Applications. Studies in Computational Intelligence, vol. 145. Springer, Heidelberg (2008)
4. Pawlak, Z.: Rough Sets – Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)
5. Skowron, A., Rauszer, C.: The discernibility matrices and functions in information systems. In: Slowinski, R. (ed.) Intelligent Decision Support. Handbook of Applications and Advances of the Rough Set Theory, pp. 331–362. Kluwer Academic Publishers, Dordrecht (1992)
6. Ślęzak, D.: Approximate entropy reducts. Fundamenta Informaticae 53, 365–390 (2002)