Generation of Exhaustive Set of Rules within Dominance-based Rough Set Approach

Krzysztof Dembczyński, [1] Roman Pindur, [2] Robert Susmaga [3]

Institute of Computing Science
Poznań University of Technology
60-965 Poznań, Poland

Abstract

The rough sets theory has proved to be a useful mathematical tool for the analysis of a vague description of objects. One of extensions of the classic theory is the Dominance-based Set Approach (DRSA) that allows analysing preference-ordered data. The analysis ends with a set of decision rules induced from rough approximations of decision classes. The role of the decision rules is to explain the analysed phenomena, but they may also be applied in classifying new, unseen objects. There are several strategies of decision rule induction. One of them consists in generating the exhaustive set of minimal rules. In this paper we present an algorithm based on Boolean reasoning techniques that follows this strategy with in DRSA.

Key words: Dominance-based Rough Set approach, decision rules, decision rules induction, exhaustive set of rules.

1 Introduction

Classification is one of the most frequently posed decision problems. It concerns an assignment of objects, described by a set of attributes, to pre-defined classes. Very often, in the analysed data, there may appear some inconsistencies or situations, in which two objects having the same description are assigned to different classes. To deal with such inconsistency, the rough set approach (further related in this paper as the Classic Rough Set Approach – CRSA) has been proposed by Pawlak [6,7]. The key idea of rough sets is the approximation of some knowledge by other knowledge. The granules of knowledge identified by indiscernibility relation are used for these approximations.

[1] Email: `Krzysztof.Dembczynski@cs.put.poznan.pl`
[2] Email: `Roman.Pindur@cs.put.poznan.pl`
[3] Email: `Robert.Susmaga@cs.put.poznan.pl`

The original rough set theory approach does not consider, however, attributes with preference-ordered domains, i.e. criteria. Nevertheless, in many real situations the ordering properties of the considered attributes play a crucial role. F orinstance, such features of objects as product quality, market share or debt ratio are typically treated as criteria in economic problems. Classification taking into account the preference-ordered data is called *sorting*. Motivated b y this observation, Greco, Matarazzo and Slowinski [1,2,3,4] yproposed an extension of the rough sets approach, called Dominance-based Rough Set Approach (DRSA).

In DRSA, where condition attributes are criteria and classes are preference-ordered, the knowledge approximated is a collection of upward and downward unions of classes and the granules of knowledge are sets of objects defined using a dominance relation. The rough set analysis arrives at a set of decision rules, which are induced from rough approximations of unions of decision classes. Decision rules are expressions of the form '*if* ..., *then* ...', which are discriminant and minimal at the same time. We can distinguish three types of rules that describe certain, possible and approximate knowledge. The main role of induced rules is to explain regularities and relationships in the analysed data set. Moreover, the set of rules combined with a particular classification/sorting method may be used to classify/sort new, unseen objects.

Generating decision rules is a complex task. Within the rough sets approaches a number of procedures were proposed that implement this process. Some of them use the strategy of computing an exhaustive set of rules, i.e. the set of all minimal rules. Let us notice that such a set of rules may be obtained b y different approaches. Within CRSA the well-known algorithms are: the all-rules option of the LERS system [5], tec hniquesbased on relative cores [6], the *Explore* algorithm, based on the apriori property [13] and approaches based on the notion of discernibility matrix and Boolean reasoning [11,12,18]. It is obvious that any other CRSA rule induction strategy induces a subset of the exhaustive set of decision rules. The problem looks a little bit different within DRSA, where we distinguish two kinds of rules with respect to their construction: robust rules, i.e. rules based on objects, and non-robust rules. A set of all robust rules is different from the set of non-robust rules. *DomApriori* [14] is an extension of *Explore* that may induce an exhaustive set of non-robust rules. Two other procedures, *All-Rules* [15]and *Aristotle* [8], follo w the strategy of inducing all robust rules. The later one is characterised b y specific binary representation of relations occurred in data.

Despite the fact that induction of an exhaustive set of rules has been considered many times, it is hard to determine the motivation of using this strategy. This approach produces the most comprehensive knowledge base on the analysed data set but it certainly requires a considerable amount of computing time and operational memory, as the complexity of the process is exponential. Such discussions hav e taken place several times in rough sets literature(see, for instance: [5,13]). In this paper we skip this discussion and simply focus on presenting another technique for obtaining such representation of analysed data within DRSA. The algorithm adopts the basic methods of Boolean reasoning in looking for object-related reducts (in the context of DRSA). Additionally, we introduce the notion of the dominating/dominated local reduct and define the dominance matrix and the dominance function.

The structure of the paper is as follows. In section 2, a brief reminder of DRSA is presented. Section 3 contains description of the proposed algorithm and section 4 shows a short example of its application. Section 5 concludes the paper.

2 Dominance-based Rough Set Approach

2.1 Data representation

Data are often presented as a table, where columns are labelled by *criteria*, rows by *objects*, and entries of the table are *criterion values*. Formally a *decision table* is the 4-tuple $S = \langle U, Q, V, f \rangle$, where U is a finite set of objects, Q is a finite set of criteria, $V = \bigcup_{q \in Q} V_q$, where V_q is the domain of the criterion q, and $f : U \times Q \to V$ is an *information function* such that $f(x, q) \in V_q$ for every $(x, q) \in U \times Q$. The set Q is divided into *condition criteria* (set $C \neq \emptyset$) and the *decision criterion* d.

It is assumed that the domain of a criterion $q \in Q$ is completely preordered by an *outranking relation* $\succeq_q$; $x \succeq_q y$ means that x is at least so good as (*outranks*) y with respect to the criterion q [10]. In the following, without any loss of generality, we consider condition criteria having numerical domains, i.e. $V_q \subseteq \Re$ ($\Re$ denotes the set of real numbers) and being of type gain i.e.: $x \succeq_q y \Rightarrow f(x, q) \geqslant f(y, q)$, where $q \in C$, $x, y \in U$. The former constraint permits simple use of such operators as: $\geqslant$ or $\leqslant$. In general, the domain of condition criterion may be also discrete, but the preference order between its values has to be provided.

The decision criterion d, the domain of which is $V_d = \{v_d^t, t \in T\}$, $T = \{1, ..., n\}$, induces a partition $Cl(d) = \{Cl_t, t \in T\}$ of U into a finite number of classes $Cl_t = \{x \in U : f(x, d) = v_d^t\}$. Each object $x \in U$ is assigned to one and only one class $Cl_t \in Cl(d)$. The classes from $Cl(d)$ are preference-ordered according to an increasing order of class indices, i.e. for all r, $s \in T$, such that r¿s, the objects from Cl_r are strictly preferred to the objects from Cl_s. For this reason, we can consider the *upward* and *downward unions* of classes, which are defined, respectively, as: $Cl_t^{\geqslant} = \bigcup_{s \geqslant t} Cl_s$, $Cl_t^{\leqslant} = \bigcup_{s \leqslant t} Cl_s$, $t \in T$. The statement $x \in Cl_t^{\geqslant}$ means "x belongs to *at least* class Cl_t", while $x \in Cl_t^{\leqslant}$ means "x belongs to *at most* class Cl_t".

2.2 Dominance relation and approximation of class unions

The dominance relation that identifies granules of knowledge is defined as follows. For a given decision table S, x *dominates* y with respect to $P \subseteq C$, denoted by $x \; D_P \; y$, if $x \succeq_q y, \forall q \in P$. For each $P \subseteq C$, the dominance relation D_P is reflexive and transitive, i.e. it is a partial pre-order.

Given $P \subseteq C$ and U, the granules of knowledge induced by the dominance relation D_P are: the set of objects dominating x, $D_P^+(x) = \{y \in U : yD_Px\}$, and the set of objects dominated by x, $D_P^-(x) = \{y \in U : xD_Py\}$, which are called *P-dominating* set and *P-dominated* set with respect to $x \in U$, respectively. The granules are used for rough approximations. The sets to be approximated are upward and downward unions of classes. Given a decision table S, the P-lower and P-upper approximations of $Cl_t^{\geqslant}$, $t \in T$, with respect to $P \subseteq C$, are defined, respectively, as: $\underline{P}(Cl_t^{\geqslant}) = \{x \in U : D_P^+(x) \subseteq Cl_t^{\geqslant}\}$, $\overline{P}(Cl_t^{\geqslant}) = \{x \in U : D_P^-(x) \cap Cl_t^{\geqslant} \neq \emptyset\}$. Analogously, P-lower and P-upper approximation of $Cl_t^{\leqslant}$, $t \in T$, with respect to $P \subseteq C$, are defined, respectively, as: $\underline{P}(Cl_t^{\leqslant}) = \{x \in U : D_P^-(x) \subseteq Cl_t^{\leqslant}\}$,, and $\overline{P}(Cl_t^{\leqslant}) = \{x \in U : D_P^+(x) \cap Cl_t^{\leqslant} \neq \emptyset\}$. Finally, the P-boundaries of $Cl_t^{\geqslant}$ and $Cl_t^{\leqslant}$ are defined as: $Bn(Cl_t^{\geqslant}) = \overline{P}(Cl_t^{\geqslant}) - \underline{P}(Cl_t^{\geqslant})$, $Bn(Cl_t^{\leqslant}) = \overline{P}(Cl_t^{\leqslant}) - \underline{P}(Cl_t^{\leqslant})$.

2.3 Definition of decision rules

The decision rules are expressions of the form '*if* [conditions], *then* [consequent]' that represent a form of dependency between condition criteria and the decision criterion. Procedures for generating decision rules from a decision table use an inductive learning principle. In order to induce decision rules with the consequent K, objects concordant with K are called *positive examples* (Pos) while all the others – *negative examples* (Neg).

We can distinguish three types of rules: certain, possible and approximate. Certain rules are generated from lower approximations of unions of classes; possible rules are generated from upper approximations of unions of classes and approximate rules are generated from boundary regions. In the following, for the reason of simplicity, we consider only certain rules. Analogous reasoning holds, however, also for possible rules. Approximate rules, on the other hand, are more complex and not all of the described notions could be easily generalized for this kind of rules.

The positive examples for certain rules are those from each lower approximation, i.e. $\underline{P}(Cl_t^{\geqslant})$ and $\underline{P}(Cl_t^{\leqslant})$, of each considered union of classes, $Cl_t^{\geqslant}$ and $Cl_t^{\leqslant}$. The corresponding negative examples are taken from $U - \underline{P}(Cl_t^{\geqslant})$, i.e. $\overline{P}(Cl_t^{\leqslant})$, and $U - \underline{P}(Cl_t^{\leqslant})$, i.e. $\overline{P}(Cl_t^{\geqslant})$, respectively.

Considering upward and downward unions we can distinguish two types of rules:

- D$\geqslant$-*decision rules* with the following syntax:
  if $f(x,q_1) \geqslant r_{q_1}$ and $f(x,q_2) \geqslant r_{q_2}$ and… and $f(x,q_p) \geqslant r_{q_p}$, *then* $x \in Cl_t^{\geqslant}$,
- D$\leqslant$-*decision rules* with the following syntax:
  if $f(x,q_1) \leqslant r_{q_1}$ and $f(x,q_2) \leqslant r_{q_2}$ and… and $f(x,q_p) \leqslant r_{q_p}$, *then* $x \in Cl_t^{\leqslant}$,
  where $P=\{q_1,...,q_p\} \subseteq C$, $(r_{q_1},...,r_{q_p}) \in Vq_1 \times Vq_2 \times ... \times V_{q_p}$ and $t \in T$.

So, in its general form the *condition part* (or *antecedent*) of the decision rule is a conjunction of elementary conditions. The total number of elementary conditions in the rule is called the *length* of the rule.

Consider a D$\geqslant$-decision rule *if* $f(x,q_1)\geqslant r_{q_1}$ and $f(x,q_2)\geqslant r_{q_2}$ and... and $f(x,q_p)\geqslant r_{q_p}$, *then* $x \in Cl_t^{\geqslant}$. If there exists an object $y \in \underline{P}(Cl_t^{\geqslant})$ such that $f(x,q_1)= r_{q_1}$ and $f(x,q_2)= r_{q_2}$ and... and $f(x,q_p)= r_{q_p}$ then y is called the *basis* of the rule. Each D$\geqslant$-decision rule having a basis is called *robust* because it is 'founded' on an object. Analogous definition of robust decision rules exists for the possible rules (but not for the approximate rules).

Moreov er, each decision rule should be minimal. Since a decision rule is an implication, by a *minimal* decision rule we understand such an implication that there is no other implication with an antecedent of at least the same weakness (in other words, rule using a subset of elementary conditions or/and weaker elementary conditions) and a consequent of at least the same strength (in other words, rule assigning objects to the same union or sub-union of classes).

3 All-Dominance-based-Rules Induction Algorithm

Proposed *All-Dominance-based-Rules Induction Algorithm* (ADRIA) induces a set of all minimal, robust rules. The key idea is based on incorporating the Boolean reasoning in to DRSA, which is used employ ed to search for object-related reducts, i.e. local reducts. This method may be used to generate certain and possible rules. In the following we present an implementation that allo wscomputing certain rules and assumes that the rules are induced using the subset of condition criteria $P \subseteq C$.

Within DRSA, local reducts are defined as follows. The *dominating lo cal reduct* $r^+(x) \subseteq P$ (or a dominating reduct relative to decision $Cl_t^{\geqslant}$ and object $x \in \underline{P}(Cl_t^{\geqslant})$, $t \in \{2,...,n\}$; where x is called a *base object*) is a subset of criteria such that:

$\forall y \in \overline{P}(Cl_{t-1}^{\leqslant}) \quad (\exists r \in r^+(x) : \neg y \succeq_r x)$, and $r^+(x)$ is minimal with respect to inclusion.

The *dominated lo cal reduct* $r^-(x) \subseteq P$ (dominated reduct relative to decision $Cl_s^{\leqslant}$ and object $x \in \underline{P}(Cl_s^{\leqslant})$, $s \in \{1,...,n-1\}$; where x is called a *base object*) is a subset of criteria such that:

$\forall y \in \overline{P}(Cl_{s+1}^{\geqslant}) \quad (\exists r \in r^-(x) : \neg x \succeq_r y)$, and $r^-(x)$ is minimal with respect to inclusion.

In other words, a dominating/dominated local reduct based on an object x is a minimal subset of criteria that allows to distinguish in the sense of dominance relation the object $x \in Pos$ from all objects belonging to *Neg*.

It is easy to see that on the base of a local reduct it is possible to build a minimal, robust rule. Consider a rule induced from the positive examples *Pos* with the consequent K. The condition part of this rule is constructed b y associating each criterion of a local reduct (based on x) with a value of x on this criterion, and the decision part is compatible with K. We say that the local reduct leaves a *trace* on the object x.

Within CRSA the concepts of the discernibility matrix and the discernibility function [10] were often used in the process of generating all local reducts. Similar concepts may be found in DRSA. Dominance matrix is defined as follows:

$DM(P) = \{\delta^{\succeq}(x,y) : x, y \in U\}$, where $\delta^{\succeq}(x,y) = \{q \in P : x \succeq_q y\}$ and $P \subseteq C$.

In other words, the set $\delta^{\succeq}(x,y)$ contains all criteria on which x outranks object y, while $\delta^{\succeq}(y,x)$ is the set of criteria on which x is outranked by y. Moreover, if $\delta^{\succeq}(x,y) = P$, then x P-dominates y, and if $\delta^{\succeq}(y,x) = P$, then x is P-dominated by y. Objects x and y are indifferent with respect to P if $\delta^{\succeq}(x,y) = \delta^{\succeq}(y,x) = P$. The matrix DM is not symmetric.

Dominance function $Df(P)$ is a Boolean function defined as follows. Let q^* be a Boolean variable corresponding to $q \in P$, and let $\bigcup \neg\delta^{\succeq}(x,y)$ denote a Boolean sum of all negated Boolean variables associated with the set of criteria $\delta^{\succeq}(x,y)$.

P-Dominance function for dominating local reducts $Df^{x\geqslant}(P)$ based on object $x \in \underline{P}(Cl_t^{\geqslant})$, $t \in \{2, ..., n\}$ is defined as:

$Df_t^{x\geqslant}(P) = \prod \{\bigcup \neg\delta^{\succeq}(y,x) : y \in \overline{P}(Cl_{t-1}^{\leqslant})\}$, $P \subseteq C$.

P-Dominance function for dominated local reducts $Df^{x\leqslant}(P)$ based on object $x \in Cl_s^{\leqslant}$, $s \in \{1, ..., n-1\}$ is defined as:

$Df_s^{x\leqslant}(P) = \prod \{\bigcup \neg\delta^{\succeq}(x,y) : y \in \overline{P}(Cl_{s+1}^{\geqslant})\}$, $P \subseteq C$.

Transformation of this functions from its conjunctive form to the non-redundant disjunctive form corresponds to problem of searching of all local reducts relative to x. The set of all prime implicants of $Df^{x\geqslant}(P)$ (or $Df^{x\leqslant}(P)$) determines the set of all dominating (or dominated) local reducts relative to x with respect to $P \subseteq C$. The transformation can be aptly realized through the *FRGA* (Fast Reduct Generating Algorithm) originally presented by Susmaga (see: [16,18]).

The general scheme of the presented algorithm is presented in Figure 1. For simplicity reasons, we consider the induction of D$\geqslant$decision rules. The set of positive examples *Pos* consists of objects belonging to $\underline{P}(Cl_t^{\geqslant})$ and the set of negative examples *Neg* contains all other objects.

The main part of the algorithm consists of generating all dominating local reducts with the *FRGA* algorithm and testing the minimality of the rule. In step 2.1 all local reducts of the object x_i with respect to negative examples are computed with *FRGA*. Next, on the base of the local reduct, a rule is created (procedure *trace* in step 2.3). Step 2.4 consists of testing if the rule is minimal. The non-minimal rules are omitted from the final result.

Testing the minimality of a rule is a complex and computationally difficult process. The simplest method consists in comparing a newly generated rule to every rule that was generated previously. This naïve method is not computationally effective. Below we present an algorithm, in which it is enough

Input: set of positive examples Pos, set of negative examples $Ne\ g$
Output: set of all minimal robust rules R that distinguish objects belonging to Pos from objects belonging to $Ne\ g$

Inducing of exhaustive set of minimal robust rules

Step 1: $R = \emptyset$
Step 2: for i = 1... $|Pos|$
Step 2.1: $R(x_i) = FRGA(x_i, Neg)$;
Step 2.2: for eac hj = 1.. $|R(x_i)|$
Step 2.3: $r_l = Trace(r_j, x_i)$; $r_j \in R(x_i)$
Step 2.4: $Min(r_l, x_i, R)$;
Step 2.5: if r_l is minimal, then $R = R \cup r_l$;
End: The final result is the set of all minimal robust rules R.

Fig. 1. Algorithm for inducing of exhaustive set of minimal robust rules

to examine some properties of the rule using the dominance matrix to verify minimality of the rule.

Let us remind that a decision rule is minimal if there exists no other rule with an antecedent of at least the same weakness (generality) and a consequent of at least the same strength (specifity). According to the above, in order to ensure minimality of a decision rule, the following three conditions must be tested:

a) no rule with an antecedent at least the same weakness and with the same consequent exists (for example: rule r_1:'*if* $f(x,q_1)\geqslant 4$ and $f(x,q_2)\geqslant 5$, *then* $x \in Cl_3^{\geqslant}$' is not minimal with respect to r_2: '*if* $f(x,q_1)\geqslant 2$ and $f(x,q_2)\geqslant 5$, *then* $x \in Cl_3^{\geqslant}$)',

b) no rule with an antecedent of the same weakness and with a consequent at least the same strength exists (for example: rule r_1: '*if* $f(x,q_1)\geqslant 3$ and $f(x,q_2)\geqslant 3$, *then* $x \in Cl_2^{\geqslant}$' is not minimal with respect to r_2: '*if* $f(x,q_1)\geqslant 3$ and $f(x,q_2)\geqslant 3$, *then* $x \in Cl_3^{\geqslant}$)',

c) no rule with an antecedent of at least the same weakness and a consequent of at least the same strength exists (for example: rule r_1: '*if* $f(x,q_1)\geqslant 4$ and $f(x,q_2)\geqslant 4$, *then* $x \in Cl_2^{\geqslant}$' is not minimal with respect to r_2: '*if* $f(x,q_1)\geqslant 2$ and $f(x,q_2)\geqslant 3$, *then* $x \in Cl_3^{\geqslant}$)'.

Let us consider the first case (a). Assume that there are two robust rules: r_1 and r_2, induced from the set of positive examples Pos, using $FRGA$. The basis of the rules are objects $x, y \in Pos$, respectively. If $x = y$ then the rules are minimal by definition of the local reduct. The problem consists in testing if r_1 is minimal with respect to r_2, if $x \neq y$. Let us remark that r_2 would be more general only if r_1 and r_2 hav e the same length.

Let $r^+(x)$ and $r^+(y)$ be the dominating local reducts, which are bases of rules r_1 and r_2 and assume that $r^+(y) \subset r^+(x) \subseteq P \subseteq C$. The following situations may be considered:

1) x and y hav e the same values on $r^+(x)$,

2) x and y have the same values on $r^+(y)$,
3) x and y have values on $r^+(x)$, such that $xD_{r^+(x)}y$ and $\neg yD_{r^+(x)}x$,
4) x and y have values on $r^+(y)$, such that $xD_{r^+(y)}y$ and $\neg yD_{r^+(y)}x$,
5) x and y have values on $r^+(x)$, such that $yD_{r^+(x)}x$ and $\neg xD_{r^+(x)}y$,
6) x and y have values on $r^+(y)$, such that $yD_{r^+(y)}x$ and $\neg xD_{r^+(y)}y$,
7) x and y have such value, that none of the above mentioned situation does occur.

In the first two situations $r^+(x)$ is not a local reduct. If an object y is distinguished from all negative objects on $r^+(y)$, the same must occur for the object x (having the same values as y on the criteria belonging to $r^+(y)$). No such local reduct would be generated. In situations 3 and 4, $r^+(x)$ is also not a local reduct. Similarly, the correct local reduct for x and y is the set $r^+(y)$. In situations 5, 6 and 7, local reducts are constructed correctly, and rules r_1 and r_2 may be built using $r^+(x)$ and $r^+(y)$, respectively. The above leads to the conclusion that no minimal robust rule could be constructed using a dominated local reduct $r^+(x)$ for which the following holds: $\exists_{y \in Pos} : xD_{r^+(x)}y$ and $\neg\exists_{z \in Neg} : zD_{r^+(x)}y$. The second condition says that one of the dominating local reducts of y is the set $r^+(x)$.

The second case (b) relates to a situation, in which the condition part of $r_1$is as general as the condition part of r_2, but the decision part of r_2 is more specific. Consider a lower approximation of class $Cl_l^{\geqslant}$ and the object x belonging to the class Cl_s, $s > l$. Let there exists a dominating local reduct $r^+(x)$ and let y denote an object that belongs to one of the classes Cl_l, Cl_{l+1},...,Cl_{s-1} and $yD_{r^+(x)}x$. To avoid this situation it is enough to check if such object as y does not exists. Let us remark that if the above is true then there exists dominating local reducts $r^+(x)$ for lower approximations of classes $Cl_k^{\geqslant}$, $s \geqslant k \geqslant l$. Only using the local reduct generated from lower approximation of $Cl_s^{\geqslant}$, a minimal rule could be founded. Then the consequent of the rule would be the strongest (most specific).

The third situation (c) never occurs. Let us consider two rules r_1 and r_2, where the consequences are $Cl_s^{\geqslant}$ and $Cl_l^{\geqslant}$, $s > l$, respectively, and r_1 has a more general antecedent than r_2. In this case r_1 may not exists, because there exists a basis of r_2 belonging to $Cl_{s-1}^{\leqslant}$ that dominates the basis of r_1 with respect to criteria involved in r_1.

The procedure of rule minimality testing is presented in Figure 2. Let us comment shortly on this algorithm. The whole test is executed using the dominance matrix, which makes the process more efficient. The notation 'if $r^+(x) \cap \delta^{\succeq}(x, y) = r^+(x)$' is equivalent to 'if $xD_{r^+(x)}y$'. In the third step we test if any, more general, rule was induced from the previously considered examples. The fourth step consists in checking if any, more general, rule could be induced from objects that still have not been considered. The test from step 3.2 and the second condition of the test from step 4.1 permit to

Input: rule r having the consequent $Cl_t^{\geqslant}$, the base object $x_l \in Pos$, where l means that it is l-th considered object, set of positive examples Pos and set of negative examples Neg the dominance matrix with elements $\delta^{\succeq}(x, y)$ $(x, y \in U)$.
Output: true if a rule r is minimal, otherwise: false

Minimality test of a robust D$\geqslant$-decision rule

Step 1: minimal = *true*
Step 2: $r^+(x_l) = UnTrace(r,\, x_l)$;
Step 3: for i = 1, ..., l -1
Step 3.1: if $r^+(x_l) \cap \delta^{\succeq}(x_l, x_i) = r^+(x_l)$, then
Step 3.2: if $r^+(x_l) \cap \delta^{\succeq}(x_i, x_l) = r^+(x_l)$, then minimal = *false*; go to End;
Step 3.3: for j = 1, ... $|Neg|$
Step 3.4: if $r^+(x_l) \cap \delta^{\succeq}(z_j, x_i) = r^+(x_l), z_j \in Neg$, then break loop;
Step 3.5: minimal = *false*; go to End;
Step 4: for $i = l$ + 1, ..., $|Pos|$
Step 4.1: if $r^+(x_l) \cap \delta^{\succeq}(x_l, x_i) = r^+(x_l) \wedge r^+(x_l) \cap \delta^{\succeq}(x_i, x_l) \neq r^+(x_l)$, then
Step 4.2: for j = 1, ... $|Neg|$
Step 4.3: if $r^+(x_l) \cap \delta^{\succeq}(z_j, x_i) = r^+(x_l), z_j \in Neg$, then break loop;
Step 4.4: minimal = *false*; go to End;
Step 5: if $x_l \in Cl_s, s > t$ and minimal =*true*, then
Step 5.1: minimal = *false*;
Step 5.2: for i = 1, ..., $|Cl_t|$
Step 5.3: if $r^+(x_l) \cap \delta^{\succeq}(x_i, x_l) = r^+(x_l)$, then minimal = *true*; go to End;
End: The final result is the value of variable minimal.

Fig. 2. Minimality test of a robust D$\geqslant$-decision rule

conclude that if two rules having exactly the same condition parts may be build on different objects, then only one rule will be induced on the basis of the previously considered object. The fifth step concerns the situation (b).

The next section contains an example that will clarify the presented idea.

4 Example

Let us consider a decision table S, presented in Figure 3. Objects x_1 through x_6, described by the set of criteria C={q_1, q_2, q_3}, belong to three classes: Cl_1, Cl_2, Cl_3; with objects from Cl_3 being preferred over objects from Cl_2 and objects from Cl_2 being preferred over objects from Cl_1. The dominance matrix DM corresponding to S is presented in Figure 4.

Objects	q_1	q_2	q_3	d_1	Objects	q_1	q_2	q_3	d_1
x_1	6	2	3	3	x_4	3	4	1	1
x_2	5	3	4	3	x_5	2	2	4	1
x_3	3	3	3	2	x_6	4	2	1	1

Fig. 3. Decision table S

Let us consider the process of inducing rules from $\underline{C}(Cl_2^{\geqslant})$, with $x_2 \in Cl_3$ as the basis. First, we compute all dominating reducts based on this object. The appropriate Boolean function and its representation in the form of prime

$\delta^{\succeq}(x,y)$	x_1	x_2	x_3	x_4	x_5	x_6
x_1	$\{q_1,q_2,q_3\}$	$\{q_1\}$	$\{q_1,q_3\}$	$\{q_1,q_3\}$	$\{q_1,q_2\}$	$\{q_1,q_2,q_3\}$
x_2	$\{q_2,q_3\}$	$\{q_1,q_2,q_3\}$	$\{q_1,q_2,q_3\}$	$\{q_1,q_3\}$	$\{q_1,q_2,q_3\}$	$\{q_1,q_2,q_3\}$
x_3	$\{q_2,q_3\}$	$\{q_2\}$	$\{q_1,q_2,q_3\}$	$\{q_1, q_3\}$	$\{q_1, q_2\}$	$\{q_2, q_3\}$
x_4	$\{q_2\}$	$\{q_2\}$	$\{q_1,q_2\}$	$\{q_1,q_2,q_3\}$	$\{q_1,q_2\}$	$\{q_2,q_3\}$
x_5	$\{q_2,q_3\}$	$\{q_3\}$	$\{q_3\}$	$\{q_3\}$	$\{q_1,q_2,q_3\}$	$\{q_2,q_3\}$
x_6	$\{q_2\}$	$\{\emptyset\}$	$\{q_1\}$	$\{q_1,q_3\}$	$\{q_1,q_2\}$	$\{q_1,q_2,q_3\}$

Fig. 4. Dominance matrix DM corresponding to S presented in Table 3.

implicants is presented below:

$Df_2^{x_2\geqslant}(C)=\prod\{\bigcup\neg\delta^{\succeq}(y,x_2):y\in\overline{C}(Cl_1^{\leqslant})\}=(q_1^*\vee q_3^*)\wedge(q_1^*\vee q_2^*)\wedge(q_1^*\vee q_2^*\vee q_3^*)$ $=q_1^*\vee q_2^*\wedge q_3^*$

The reducts are $r_1^+(x_2)=\{q_2, q_3\}$ and $r_2^+(x_2)=\{q_2, q_3\}$. Using the function 'trace' we can build two rules based on x_2: r_1: '*if* $f(x,q_1)\geqslant 5$, *then* $x\in Cl_2^{\geqslant}$', and r_2: '*if* $f(x,q_2)\geqslant 3$ and $f(x,q_3)\geqslant 4$, *then* $x\in Cl_2^{\geqslant}$.

Now, we have to verify the minimality of these rules using the algorithm described in the previous section. Let us consider the rule r_1 first. In step 3 we examine if any, more general, rule had been induced before. In this case we have to test if $r_1^+(x_2)\cap\delta^{\succeq}(x_2,x_1) = r_1^+(x_2)$, i.e. if $x_2D_{\{q_1\}}x_1$. Because it is not true we enter step 4, which consists in checking if a more general rule could be induced from objects that have still not been considered. Here we test conditions $x_2D_{\{q_1\}}x_3$ and $\neg x_3D_{\{q_1\}}x_2$. It is easy to prove that both hold, which means that x_3 might be a basis of a rule that would render r_1 not minimal. To ensure the above we have to check if x_3 belongs to $\underline{\{q_1\}}(Cl_2^{\geqslant})$ (lower approximation of $Cl_2^{\geqslant}$ with respect to $\{q_1\}$), in other words, that there exists no $z\in Neg$ (i.e. $\overline{C}(Cl_1^{\leqslant})$) such that $zD_{\{q_1\}}x_3$. This is tested in steps 4.2—4.5. However, x_7 dominates x_3 with respect to $\{q_1\}$ and x_3 is not the basis of a rule built over q_1. Next, in step 5, we check if the decision part is not to general. In our example there exists no object y belonging to Cl_2, such that $yD_{\{q_1\}}x_2$, which suggests that a rule with the same condition part would be also induced from $\underline{C}(Cl_3^{\geqslant})$, i.e. *if* $f(x,q_1)\geqslant 5$, *then* $x\in Cl_3^{\geqslant}$. Because of this r_1 is not minimal.

Similar process may be conducted for r_2. Let us remark that in this case x_2 dominates both x_1 and x_3 with respect to $\{q_2, q_3\}$. However, it is impossible to build a rule based on x_1 because $x_5D_{\{q_2,q_3\}}x_2$. Because of an opposite situation concerning x_3, the minimal rule: '*if* $f(x,q_2)\geqslant 3$ and $f(x,q_3)\geqslant 3$, *then* $x\in Cl_2^{\geqslant}$' will be induced in the next iteration of the algorithm basing on x_3.

Below we present all certain rules, with the bases, induced from S:

if $f(x,q_1)\geqslant 5$, *then* $Cl_3^{\geqslant}$; basis: x_2.

if $f(x,q_2)\geqslant 3$ and $f(x,q_3)\geqslant 4$, *then* $Cl_3^{\geqslant}$; basis: x_2;

if $f(x,q_1)\geqslant 3$ and $f(x,q_2)\geqslant 3$, *then* $Cl_2^{\geqslant}$; basis: x_3;

if $f(x,q_2)\geqslant 3$ and $f(x,q_2)\geqslant 3$, *then* $Cl_2^{\geqslant}$; basis: x_3;

if $f(x,q_1)\leqslant 4$, *then* $Cl_2^{\leqslant}$; basis: x_4;

if $f(x,q_3)\leqslant 1$, *then* $Cl_1^{\leqslant}$; basis: x_6;

if $f(x,q_1) \leqslant 2$, *then* $Cl_1^{\leqslant}$; basis: x_5;
if $f(x,q_1) \leqslant 4$ and $f(x,q_2) \leqslant 2$, *then* $Cl_1^{\leqslant}$; basis: x_6.

5 Conclusions

We presented an alternative algorithm for inducing the exhaustive set of decision rules within DRSA. It is worth stating that this technique may be easily adopted to generate minimal sets of rules or satisfactory sets of rules. Both are other strategies of rule generation, which are often applied in real-life applications. The first strategy is focused on describing objects using the minimum number of necessary rules covering all objects from a decision table. The second category gives as its result the set of decision rules that satisfy some pre-defined user's requirements.

The main idea of the presented algorithm resolves itself into incorporating Boolean reasoning into DRSA. *ADRIA* utilizes the concepts of dominating/dominated local reducts, the dominance matrix and the dominance function, which are notions analogous to those of local reducts, discernibility matrix and discernibility function (all well-known in the context of CRSA). In one of its initial steps the algorithm employs the *FRGA*, the result of which, namely the set of all local reducts, serves to build rules out of objects. In further steps *ADRIA* uses a promising procedure for verifying the minimality of rules.

6 Acknowledgement

This paper has been supported by the research grant no. 8T11F-006-19 from the Polish Research Committee (KBN). The authors are grateful to the anonymous reviewer of this paper for very apt and productive comments contained in the review.

References

[1] Greco S., Matarazzo, B. and Slowinski, R., Rough approximation of a preference relation by dominance relations. ICS Research Report 16/96, Warsaw University of Technology, Warsaw, and 1996 in: *European Journal of Operational Research* **117**, 63-83 1999.

[2] Greco S., Matarazzo, B. and Slowinski, R., A new rough set approach to evaluation of bankruptcy risk. In: Zopounidis, C. (ed.), *Operational Tools in the Management of Financial Risks*. Kluwer, Dordrecht, 121-136, 1998.

[3] Greco S., Matarazzo, B. and Slowinski, R., Rough sets theory for multicriteria decision analysis. *European Journal of Operational Research*, **129**, 1, 1-47, 2001

[4] Greco S., Matarazzo, B. and Slowinski, R., Rough sets methodology for sorting problems in presence of multiple attributes and criteria. *European Journal of Operational Research*, **138**, 247-259, 2002

[5] Grzymala-Busse, J.W., LERS - a system for learning from examples based on rough sets. In: Slowinski, R. (ed.), *Intelligent Decision Support. Handbook of Applications and Advances of the Rough Sets Theory*. Kluwer, Dordrecht, 3-18, 1992.

[6] Pawlak, Z., Rough sets. *International Journal of Information & Computer Sciences*, **11**, 341-356, 1982.

[7] Pawlak, Z., *Rough Sets. Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, Dordrecht, 1991.

[8] Pindur, R., Susmaga, R., Fast Rule Extraction with Binary-Coded Relations, *Intelligent Data Analysis*, 7 **1**, 2003.

[9] Roy, B. *Méthodologie Multicritère d'Aide à la Décision*. Economica, Paris, 1985.

[10] 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*, Kluwer Academic Publishers, Dordrecht 331-362, 1992.

[11] Skowron, A., Boolean reasoning for decision rules generation. In: Komorowski, J. and Ras, Z.W. (eds.), *Methodologies for Intelligent Systems*, *Lecture Notes in Artificial Intelligence*, **689**, Springer-Verlag, Berlin, 295-305, 1993.

[12] Skowron, A., Extracting laws form decision table. *Computational Intelligence: An International Journal* **11/2**, 371-388, 1995.

[13] Stefanowski, J., On rough set based approaches to induction of decision rules. In: Polkowski, L. and Skowron, A. (eds.), *Rough Sets in Data Mining and Knowledge Discovery*, **1**, Physica-Verlag, Heidelberg, 500-529, 1998.

[14] Stefanowski, J., *Algorytmy indukcji regul decyzyjnych w odkrywaniu wiedzy* Wydawnictwo Politechniki Poznanskiej, Poznan, **361**, 2001.

[15] Stefanowski J., Zurawski M., *Incremental learning of decision rules for classification problems* In Proceedings of the Conference IIS-2003: New Trends in Intelligent Information Processing and Webmining, Zakopane, June 2003, (accepted).

[16] Susmaga, R., Experiments in Incremental Computation of Reducts. In: Polkowski, L., Skowron, A. (eds.), *Rough Sets in Knowledge Discovery*, Physica-Verlag, Heidelberg, New York, 530-553, 1998.

[17] Susmaga, R., New Tests for Inclusion Minimality in Reduct Generation. *Foundations of Computing and Decision Sciences*, Poznan, Poland, 25, **2**, 121-146, 2000.

[18] Susmaga, R., Generation of Exhaustive Rule Sets Using a Reduct generating Algorithm, In: Kopotek, M., Michalewicz, M., Wierzchoń, S. (eds.), *Intelligent Information Systems*, *series Advanes in Soft Computing*, Physica-Verlag, 65-74, 2000.