VC-DomLEM: Rule induction algorithm for variable consistency rough set approaches

Jerzy Błaszczyński[1], Roman Słowiński[1,2], Marcin Szeląg[1]

[1] Institute of Computing Science, Poznań University of Technology, 60-965 Poznań, Poland
{jblaszczynski, rslowinski, mszelag}@cs.put.poznan.pl
[2] Institute for Systems Research, Polish Academy of Sciences, 01-447 Warsaw, Poland

Abstract. We present a general rule induction algorithm based on sequential covering, suitable for variable consistency rough set approaches. This algorithm, called VC-DomLEM, can be used for both ordered and non-ordered data. In the case of ordered data, the rough set model employs dominance relation, and in the case of non-ordered data, it employs indiscernibility relation. VC-DomLEM generates a minimal set of decision rules. To this end, the following rule consistency measures are applied: ϵ-consistency, ϵ'-consistency, and μ-consistency. We analyze properties of induced decision rules, and discuss conditions of correct rule induction. Moreover, we show how to improve rule induction efficiency due to application of monotonic consistency measures.

1 Introduction

Rough set approach to reasoning about data consists of the following steps. In the first step, the data is checked for possible inconsistencies, by calculation of lower and upper approximations of considered sets of objects. In case of the original Rough Set Approach proposed by Pawlak [41–43, 45], the approximated sets are decision classes. Since this approach assumes that data are not ordered, and thus employs indiscernibility relation, we call it Indiscernibility-based Rough Set Approach (IRSA). In the Dominance-based Rough Set Approach (DRSA) proposed by Greco et al. [22, 23, 26, 55], where data are ordered and it is assumed that there exists a monotonic relationship between evaluations of objects and their assignment to ordered decision classes, one approximates upward and downward unions of decision classes. The classification problem handled by DRSA is called *ordinal classification with monotonicity constraints*. In both approaches, the approximations are built using granules of knowledge, which are either indiscernibility classes (IRSA) or dominance cones (DRSA). In IRSA and DRSA, the lower approximation of a set is defined using a strict inclusion relation of the granules of knowledge in the approximated set. The lower approximation is thus composed only of the granules that are subsets of the approximated set. This definition of the lower approximation appears to be too restrictive in practical applications. In consequence, lower approximations of sets are often empty, preventing generalization of data in terms of relative certainty. This observation has motivated research on probabilistic generalizations of rough sets. Different versions of probabilistic rough set approaches were proposed, starting from Variable Precision Rough Set (VPRS) model [59, 63, 64], Variable Consistency Dominance-based Rough Set Approaches (VC-DRSA) [7, 23], Bayesian Rough Set (BRS) model and Rough Bayesian (RB) model [58, 59], Decision Theoretic Rough Set model [29, 61, 62] and Parameterized Rough Sets (PRS) [30]. The probabilistic rough set approaches allow to extend lower approximation of a set by objects with sufficient evidence for membership to the set. In this paper, we rely on the monotonic Variable Consistency Indiscernibility-based Rough Set Approaches (VC-IRSA) and Variable Consistency Dominance-based Rough Set Approaches proposed in [8, 9]. The former approaches extend IRSA, while the latter are extensions of DRSA. We use different *object consistency measures* to quantify the evidence for membership to a set. They measure the overlap between a granule of knowledge based on an object and the approximated set or its complement.

In the second step, decision rules are generated in order to generalize description of objects contained in approximations. Objects from lower approximations of sets are the basis for induction of *certain rules*, objects from upper approximations of sets are used to obtain *possible rules*, and objects from boundaries of sets are used to generate *approximate rules*. In this paper, we present an algorithm for induction of a minimal set of minimal decision rules, based on sequential covering. This algorithm, called VC-DomLEM, generalizes the description of objects contained in probabilistic lower approximations defined according to VC-IRSA or VC-DRSA. When applied to the set of probabilistic lower approximations, it induces a set of *probabilistic rules*. Each such rule is supported by objects from a lower approximation and is allowed to cover objects from the respective positive region. To control the quality of the rules, we use three different *rule consistency measures* – ϵ-consistency, ϵ'-consistency, and μ-consistency. These measures have the same properties as corresponding object consistency measures used to calculate probabilistic lower approximations.

The overall goal of using VC-DomLEM is to find smallest possible set of rules with high predictive accuracy. In general, induction of decision rules is a complex problem and many algorithms have been introduced to solve it. Examples of rule induction algorithms that were presented for IRSA are the algorithms: by Grzymała-Busse [32], by Skowron [56], by Słowiński and Stefanowski [52], and by Stefanowski [57]. Algorithms defined for DRSA have been proposed: by Greco et al. [25], by Błaszczyński and Słowiński [6] and by Dembczyński et al. [15]. All these algorithms can be divided into three categories that reflect different induction strategies: generation of a minimal set of decision rules, generation of an exhaustive set of decision rules, and generation of a satisfactory set of decision rules. Algorithms from the first category focus on describing objects from lower approximations by minimal number of minimal rules that are necessary to cover all consistent objects from the decision table. Algorithms from the second category generate all minimal decision rules. The third category includes algorithms that generate all minimal rules that satisfy some a priori defined requirements (e.g., maximal rule length or minimal support). According to this classification, VC-DomLEM belongs to the first category.

Decision rules are considered to be a data model. Thus, in the case of classification problems addressed by IRSA, and in the case of more general ordinal classification problems with monotonicity constraints addressed by DRSA, they do not only describe the data, but they also can be used for prediction.

Classification of (new) objects by the induced decision rules is the third step of the rough set approach. In this step, recommendations of decision rules for classified objects are aggregated using classification strategies [5, 26, 27].

This paper is organized as follows. In Section 2, we remind basic definitions of the original Indiscernibility-based Rough Set Approach and the Dominance-based Rough Set Approach. We define monotonicity properties required for object consistency measures that are used in Monotonic Variable Consistency Rough Set Approaches. We present definitions of probabilistic lower approximations, followed by definitions of positive, negative and boundary regions of approximated sets of objects. In Section 3, we define syntax and semantics of decision rules. Section 4 introduces the properties of induced decision rules. In Section 5, we present VC-DomLEM, which is an algorithm that induces decision rules by sequential covering [35], also called separate and conquer [20]. In Section 6, we describe the setup of a computational experiment, performed to analyze the behavior of VC-DomLEM algorithm for different rule consistency measures. Section 7 contains the results of this experiment. In the last Section 8, we give final remarks and conclude the paper.

2 Rough set approximations and respective regions of evaluation space

In the rough set approach, classification of object y from universe U to a given set $X \subseteq U$ is based on available data. Data is presented as a decision table, where rows correspond to objects from U and columns correspond to attributes from a finite set A. Among attributes from set A there are attributes with preference-ordered value sets, called criteria, and regular attributes whose value sets are not preference-ordered. Moreover, the set of attributes A is divided into disjoint sets of condition attributes C and decision attributes D. The value set of attribute $q \in C \cup D$ is denoted by V_q. $V_P = \prod_{q=1}^{|P|} V_q$ is called P-evaluation space, where $P \subseteq C$. For simplicity, we assume set D to be a singleton $D = \{d\}$.

The decision attribute d makes a partition of set U into a finite number of disjoint sets of objects, called decision classes. Let $X \subseteq U$ be one of these decision classes. Decision about classification of object $y \in U$ to set X depends on its class label known from the decision table, and/or on its relation with other objects from the table. In IRSA, the considered relation is the *indiscernibility relation* in the evaluation space [41–43, 45]. Consideration of this relation is meaningful when set of attributes A is composed of regular attributes only. Indiscernibility relation makes a partition of universe U into disjoint blocks of objects that have the same description and are considered indiscernible. Such blocks are called *granules*. Moreover, $I_P(y)$ denotes a set of objects indiscernible with object y using set of attributes $P \subseteq C$. It is called a granule of P-indiscernible objects.

When condition attributes from C and decision attribute d have preference-ordered value sets, in order to make meaningful classification decisions, one has to consider the *dominance relation* instead of the indiscernibility relation in the evaluation space. It has been proposed in [22, 23, 26, 55] and the resulting approach was called Dominance-based Rough Set Approach (DRSA). Dominance relation makes a partition of universe U into granules being *dominance cones*. Moreover, for each object $y \in U$ two dominance cones are defined with respect to (w.r.t.) $P \subseteq C$. The P-positive dominance cone $D_P^+(y)$ is composed of all objects that for each $q_i \in P$ are not worse than y. The P-negative dominance cone $D_P^-(y)$ is composed of all objects that for each $q_i \in P$ are not better than y.

We consider a classification problem with n disjoint classes numbered by decision attribute d. While in IRSA, decision classes X_i, $i = 1, \ldots, n$, are not necessarily ordered, in DRSA, they are ordered, such that if $i < j$, then class X_i is considered to be worse than X_j. Moreover, DRSA takes into account monotonic relationship between evaluations of objects on particular criteria and assignment of these objects into decision classes. For example, the better the value of criterion $q_i \in C$ for object y, the better the decision class it may belong. From this follows the *dominance principle* which says that if evaluations of object y on all considered criteria are not worse than evaluations of object z, then y should be assigned to a class not worse than z. Violation of this principle causes *inconsistency* in the data table which is captured within DRSA by approximations of sets based on dominance. In order to handle preference orders, and monotonic relationships between evaluations on criteria and assignment to decision classes, approximations made in DRSA concern the following unions of decision classes: upward unions $X_i^{\geq} = \bigcup_{t \geq i} X_t$, where $i = 2, 3, \ldots, n$, and downward unions $X_i^{\leq} = \bigcup_{t \leq i} X_t$, where $i = 1, 2, \ldots, n-1$.

In order to avoid repetition of the same definitions and properties for IRSA and DRSA, from now on we will use a unique symbol X to denote a set of all objects belonging to class X_i, in the context of IRSA, or to union of classes $X_i^{\geq}$, $X_i^{\leq}$, in the context of DRSA. Moreover, we will use symbol $E_P(y)$ to denote any granule of the type $I_P(y)$, $D_P^+(y)$ or $D_P^-(y)$, $y \in U$. If both X and $E_P(y)$ will be used in the same equation, then for X representing class X_i, $E_P(y)$ denotes granule $I_P(y)$ and for X representing union of ordered classes $X_i^{\geq}$ (resp. $X_i^{\leq}$), $E_P(y)$ stands for dominance cone $D_P^+(y)$ (resp. $D_P^-(y)$).

As written above, different probabilistic rough set approaches aim to extend lower approximation of set X by inclusion of objects with sufficient evidence for membership to X. This evidence can be quantified by different object consistency measures. In [9], we distinguished gain-type and cost-type object consistency measures and specified conditions that must be satisfied by these measures. For a gain-type measure, the higher the value, the more consistent is the given object.

For a cost-type measure, the lower the value, the more consistent is the given object.

Let us give generic definition of probabilistic P-lower approximation of set X. For $P \subseteq C, X \subseteq U, y \in U$, given a gain-type (resp. cost-type) object consistency measure $\Theta_X^P(y)$ and a gain-threshold (resp. cost-threshold) θ_X, we get the following definition of the P-lower approximation of set X:

$$\underline{P}^{\theta_X}(X) = \{y \in X : \Theta_X^P(y) \propto \theta_X\}, \tag{1}$$

where $\propto$ denotes $\geq$ in case of a gain-type object consistency measure and a gain-threshold, or $\leq$ for a cost-type object consistency measure and a cost-threshold. In the above definition, $\theta_X \in [0, A_X]$ is a technical parameter influencing the degree of consistency of objects belonging to lower approximation of X. Values of θ_X and A_X depend on the interpretation of the object consistency measure.

The definition of P-upper approximation and the definition of P-boundary of set X, both making use of the complementarity property of rough approximations, are given in [9].

In [9], we introduced and motivated four *monotonicity properties* required from object consistency measures used in definition (1). We call a P-lower approximation *monotonic* when the object consistency measure used to define it fulfills relevant monotonicity properties. For IRSA and DRSA, we are interested in the following two properties:

(m1) Monotonicity w.r.t. set of attributes $P \subseteq C$. Formally, for all $P \subseteq P' \subseteq C$, $X \subseteq U$, $y \in U$, a gain-type measure $\Theta_X^P(y)$ is monotonically non-decreasing w.r.t. P, if and only if (iff)

$$\Theta_X^P(y) \leq \Theta_X^{P'}(y), \tag{2}$$

and a cost-type measure $\Theta_X^P(y)$ is monotonically non-increasing w.r.t. P, iff

$$\Theta_X^P(y) \geq \Theta_X^{P'}(y). \tag{3}$$

(m2) Monotonicity w.r.t. set of objects $X \subseteq U$, when set X is augmented by new objects. Formally, for all $P \subseteq C$, $X \subseteq U$, $X' = X \cup X^{\Delta}$, $X^{\Delta} \cap U = \emptyset$, $y \in U$, a gain-type measure $\Theta_X^P(y)$ is monotonically non-decreasing w.r.t. X, iff

$$\Theta_X^P(y) \leq \Theta_{X'}^P(y), \tag{4}$$

and a cost-type measure $\Theta_X^P(y)$ is monotonically non-increasing w.r.t. X, iff

$$\Theta_X^P(y) \geq \Theta_{X'}^P(y). \tag{5}$$

Moreover, for DRSA additional required properties are:

(m3) Monotonicity w.r.t. union of classes $X_i^{\geq} \subseteq U$ and $X_k^{\leq} \subseteq U$. Formally, for all $P \subseteq C$, $X_i^{\geq} \subseteq X_j^{\geq} \subseteq U$, $j \leq i$, $X_k^{\leq} \subseteq X_l^{\leq} \subseteq U$, $l \geq k$, $y \in U$, gain-type measures $\Theta^P_{X_i^{\geq}}(y)$ and $\Theta^P_{X_k^{\leq}}(y)$ are monotonically non-decreasing w.r.t. $X_i^{\geq}$ and $X_k^{\leq}$, respectively, iff

$$\Theta^P_{X_i^{\geq}}(y) \leq \Theta^P_{X_j^{\geq}}(y), \quad \Theta^P_{X_k^{\leq}}(y) \leq \Theta^P_{X_l^{\leq}}(y). \tag{6}$$

Analogously, a cost-type measures $\Theta^P_{X_i^{\geq}}(y)$ and $\Theta^P_{X_k^{\leq}}(y)$ are monotonically non-increasing w.r.t. $X_i^{\geq}$ and $X_k^{\leq}$, respectively, iff

$$\Theta^P_{X_i^{\geq}}(y) \geq \Theta^P_{X_j^{\geq}}(y), \quad \Theta^P_{X_k^{\leq}}(y) \geq \Theta^P_{X_l^{\leq}}(y). \tag{7}$$

(m4) Monotonicity w.r.t. P-dominance relation D_P, $P \subseteq C$. Formally, for all $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$, and $*$ standing for either $\geq$ or $\leq$ in every instance, a gain-type measure $\Theta^P_{X_i^*}(y)$ is monotonically non-decreasing w.r.t. P-dominance relation, iff

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \Theta^P_{X_i^*}(y_1) \geq \Theta^P_{X_i^*}(y_2), \tag{8}$$

and a cost-type measure $\Theta^P_{X_i^*}(y)$ is monotonically non-increasing w.r.t. P-dominance relation, iff

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \Theta^P_{X_i^*}(y_1) \leq \Theta^P_{X_i^*}(y_2). \tag{9}$$

Let us now remind some useful definitions of positive, negative and boundary regions of X in the evaluation space, introduced in [7]. First, let us note that each set X has its complement $\neg X = U - X$. P-positive region of X in P-evaluation space is defined as:

$$POS_P^{\theta_X}(X) = \bigcup_{y \in \underline{P}^{\theta_X}(X)} E_P(y), \tag{10}$$

where θ_X comes from (1).

Basing on the definition of the positive region of set X, we also define P-negative and P-boundary regions of the approximated set as follows:

$$NEG_P^{\theta_X}(X) = POS_P^{\theta_X}(\neg X) - POS_P^{\theta_X}(X), \tag{11}$$

$$BND_P^{\theta_X}(X) = U - POS_P^{\theta_X}(X) - NEG_P^{\theta_X}(X). \tag{12}$$

Finally, let us recall definitions and monotonicity properties of object consistency measures, which will be used in definition (1).

The first object consistency measure that we consider is a cost-type measure $\epsilon_X^P(y)$. For $P \subseteq C, X, \neg X \subseteq U$, where $\neg X = U - X$, $y \in U$, it is defined as

$$\epsilon_X^P(y) = \frac{|E_P(y) \cap \neg X|}{|\neg X|}. \tag{13}$$

As proved in [9], this measure has properties (m1), (m2) and (m4). To overcome the lack of property (m3) for $\epsilon_X^P(y)$ in the context of DRSA, we proposed a modified measure $\epsilon^{*P}_X(y)$, which has all four desirable monotonicity properties. For $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$, measures $\epsilon^{*P}_{X_i^{\geq}}(y)$ and $\epsilon^{*P}_{X_i^{\leq}}(y)$ are defined as

$$\epsilon^{*P}_{X_i^{\geq}}(y) = \max_{j \leq i} \epsilon^{P}_{X_j^{\geq}}(y), \tag{14}$$

$$\epsilon^{*P}_{X_i^{\leq}}(y) = \max_{j \geq i} \epsilon^{P}_{X_j^{\leq}}(y). \tag{15}$$

The third object consistency measure is a cost-type measure $\epsilon'^{P}_X(y)$. For $P \subseteq C, X, \neg X \subseteq U$, where $\neg X = U - X$, $y \in U$, it is defined as

$$\epsilon'^{P}_X(y) = \frac{|E_P(y) \cap \neg X|}{|X|}. \tag{16}$$

As proved in [9], this measure has all four desirable monotonicity properties.

The fourth object consistency measure, defined only in the context of DRSA, is a gain-type measure $\mu'^{P}_X(y)$ introduced in [7]. For $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$, measures $\mu'^{P}_{X_i^{\geq}}(y)$ and $\mu'^{P}_{X_i^{\leq}}(y)$ are defined as

$$\mu'^{P}_{X_i^{\geq}}(y) = \max_{z \in D_P^-(y) \cap X_i^{\geq}} \mu^{P}_{X_i^{\geq}}(z), \tag{17}$$

$$\mu'^{P}_{X_i^{\leq}}(y) = \max_{z \in D_P^+(y) \cap X_i^{\leq}} \mu^{P}_{X_i^{\leq}}(z), \tag{18}$$

where $\mu^P_X(z) = \frac{|E_P(z) \cap X|}{|E_P(z)|}$, denotes rough membership of object $z \in U$ to union of classes $X \subseteq U$, w.r.t. set $P \subseteq C$. As it was proved in [9], rough membership measure $\mu^P_X(y)$ has properties (m2) and (m3), but it lacks properties (m1) and (m4). Moreover, in [7] we showed that measure $\mu'^{P}_X(y)$, extending rough membership measure, gains property (m4) but still lacks property (m1). It can be also easily shown that this measure preserves properties (m2) and (m3).

3 The syntax and semantics of decision rules

In the variable consistency rough set approaches, we consider decision rules of the type:

if Φ then Ψ,

where Φ and Ψ denote *condition* and *decision* part of the rule, called also *premise* and *conclusion*, respectively. The condition part of the rule is a conjunction of elementary conditions concerning individual attributes/criteria, and the decision part of the rule suggests an assignment to a decision class or to a union of decision classes. A precise syntax of decision rules will be given later. Decision rules are induced so as to cover objects from probabilistic lower approximations of sets being classes or unions of decision classes. However, in some cases it is impossible for a rule to cover only objects from a probabilistic lower approximation. To handle these cases, the positive region of the considered set is computed.

The set $\underline{P}^{\theta_X}(X)$ of objects belonging to the P-lower approximation of X is the basis for induction of a set of decision rules $R_X^{\hat{\theta}_X}$. Each induced rule $r_X^{\hat{\theta}_X} \in R_X^{\hat{\theta}_X}$ is supported by at least one object from $\underline{P}^{\theta_X}(X)$, it covers object(s) from $POS_P^{\theta_X}(X)$, and it suggests an assignment to X. The elementary conditions (selectors) that form the decision rules from $R_X^{\hat{\theta}_X}$ are built using evaluations of objects belonging to $\underline{P}^{\theta_X}(X)$ only. Moreover, rule $r_X^{\hat{\theta}_X}$ is characterized by a value $\hat{\Theta}(r_X^{\hat{\theta}_X})$ of considered *rule consistency measure* $\hat{\Theta}$, not worse than threshold value $\hat{\theta}_X$. Rule consistency measures are adequate to object consistency measures used in the definition of probabilistic P-lower approximation. Different rule consistency measures are discussed in Section 4. The value of threshold $\hat{\theta}_X$ depends on the value of threshold θ_X, which is also shown in Section 4.

Below, we define a syntax of decision rule $r_X^{\hat{\theta}_X} \in R_X^{\hat{\theta}_X}$ for the most general classification problem, i.e., ordinal classification with monotonicity constraints:

$$\begin{aligned}
&\textit{if } q_{i_1}(y) \succeq r_{i_1} \wedge \ldots \wedge q_{i_p}(y) \succeq r_{i_p} \wedge q_{i_{p+1}}(y) = r_{i_{p+1}} \wedge \ldots \wedge q_{i_z}(y) = r_{i_z} \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \textit{then } y \in X_i^{\geq}, \qquad (19)\\
&\textit{if } q_{i_1}(y) \preceq r_{i_1} \wedge \ldots \wedge q_{i_p}(y) \preceq r_{i_p} \wedge q_{i_{p+1}}(y) = r_{i_{p+1}} \wedge \ldots \wedge q_{i_z}(y) = r_{i_z} \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \textit{then } y \in X_i^{\leq}, \qquad (20)
\end{aligned}$$

where $q_i, i \in \{i_1, i_2, \ldots, i_p\}$ denotes criterion and $q_i, i \in \{i_{p+1}, i_{p+2}, \ldots, i_z\}$ denotes regular attribute. Moreover, r_i denotes chosen value from the value set of attribute q_i. We use symbols $\succeq$ and $\preceq$ to indicate weak preference w.r.t. single criterion and inverse weak preference, respectively. If $q_i \in C$ is a gain (cost) criterion, then elementary condition $q_i(y) \succeq r_i$ denotes that the value of object $y \in U$ on condition criterion q_i is not smaller (not greater) than value r_i. Elementary conditions for regular attributes are of the type $q_i(y) = r_i$.

Decision rule $r_X^{\hat{\theta}_X}$ covers objects that fulfill its condition part and suggest their assignment to set X. Condition part of rule $r_X^{\hat{\theta}_X}$ can be denoted by $\Phi_{r_X^{\hat{\theta}_X}}$, while its decision part can be denoted by $\Psi_{r_X^{\hat{\theta}_X}}$. Moreover, we denote by $\|\Phi_{r_X^{\hat{\theta}_X}}\|$ or $\|\Psi_{r_X^{\hat{\theta}_X}}\|$ the set of objects fulfilling condition or decision part of the rule, respectively.

Decision rule $r_X^{\hat{\theta}_X} \in R_X^{\hat{\theta}_X}$ is characterized by the following basic measures:

$$\text{support of } r_X^{\hat{\theta}_X} : supp(r_X^{\hat{\theta}_X}) = \left| \|\Phi_{r_X^{\hat{\theta}_X}}\| \cap \|\Psi_{r_X^{\hat{\theta}_X}}\| \right|, \tag{21}$$

$$\text{strength of } r_X^{\hat{\theta}_X} : \sigma(r_X^{\hat{\theta}_X}) = \frac{supp(r_X^{\hat{\theta}_X})}{|U|}, \tag{22}$$

$$\text{certainty of } r_X^{\hat{\theta}_X} : cer(r_X^{\hat{\theta}_X}) = \frac{supp(r_X^{\hat{\theta}_X})}{\left|\|\Phi_{r_X^{\hat{\theta}_X}}\|\right|}, \tag{23}$$

$$\text{coverage of } r_X^{\hat{\theta}_X} : cov(r_X^{\hat{\theta}_X}) = \frac{supp(r_X^{\hat{\theta}_X})}{\left|\|\Psi_{r_X^{\hat{\theta}_X}}\|\right|}, \tag{24}$$

where $|\cdot|$ denotes cardinality of a set.

Objects that support rule $r_X^{\hat{\theta}_X}$ are those that satisfy both condition and decision part of the rule. The strength of a rule is defined as a ratio of its support and the number of all objects in the data set. The certainty of a rule is defined as a ratio of the number of objects that support the rule to the number of objects that satisfy condition part of the rule. Coverage of a rule is defined as a ratio of the number of objects that support the rule to the number of objects that satisfy decision part of the rule.

4 Characteristics and properties of decision rules

Decision rules should be short and accurate. Shorter decision rules are easier to understand. Shorter rules also allow to avoid *overfitting* the training data. Overfitting occurs when the learned model fits training data perfectly but is not performing well on new data. Rules induced in variable consistency rough set approaches avoid overfitting because they are not required to classify training data perfectly. Such a relaxation is typical for other machine learning rule induction algorithms [11–13, 60]. It allows to induce more general rules with less elementary conditions. The difference to other rule induction algorithms proposed in machine learning is that in case of the algorithms defined within variable consistency rough set approaches, it is known a priori which objects in the data set can be classified incorrectly, i.e., which objects from the P-positive region of X do not belong to the P-lower approximation of X. Relaxation of the requirement to cover only consistent objects involves a trade-off between accuracy and simplicity [36].

Induced rules must satisfy similar constraints on consistency as objects from the lower approximation which serve as a base for rule induction. Thus, in addition to the measures specified in the previous section, a VC-DRSA decision rule $r_X^{\hat{\theta}_X}$ can be characterized by a value of chosen rule consistency measure $\hat{\Theta}$. We consider the following three rule consistency measures:

$$\epsilon\text{-consistency of } r_X^{\hat{\theta}_X} : \epsilon(r_X^{\hat{\theta}_X}) = \frac{\left|\|\Phi_{r_X^{\hat{\theta}_X}}\| \cap \neg\underline{P}^{\theta_X}(X)\right|}{|\neg\underline{P}^{\theta_X}(X)|}, \tag{25}$$

$$\epsilon'\text{-consistency of } r_X^{\hat{\theta}_X} : \epsilon'(r_X^{\hat{\theta}_X}) = \frac{\left|\|\Phi_{r_X^{\hat{\theta}_X}}\| \cap \neg\underline{P}^{\theta_X}(X)\right|}{|\underline{P}^{\theta_X}(X)|}, \tag{26}$$

$$\mu\text{-consistency of } r_X^{\hat{\theta}_X} : \mu(r_X^{\hat{\theta}_X}) = \frac{\left|\|\Phi_{r_X^{\hat{\theta}_X}}\| \cap \underline{P}^{\theta_X}(X)\right|}{\left|\|\Phi_{r_X^{\hat{\theta}_X}}\|\right|}, \tag{27}$$

where $\hat{\theta}_X = \frac{|\neg X|}{|\neg\underline{P}^{\theta_X}(X)|}\theta_X$ in definition (25), $\hat{\theta}_X = \frac{|X|}{|\underline{P}^{\theta_X}(X)|}\theta_X$ in definition (26), and $\hat{\theta}_X = \theta_X$ in definition (27). ϵ-consistency measure is related to cost-type object consistency measure ϵ defined as (13). ϵ'-consistency measure is related to cost-type object consistency measure ϵ' defined as (16). μ-consistency measure is related to gain-type rough membership measure μ used in definitions (17) and (18). It can be shown that each of the defined above rule consistency measures derives monotonicity properties from the corresponding object consistency measure.

As it will be shown in Section 5, ϵ-consistency measure can be used to induce decision rules from positive regions computed using object consistency measure ϵ^*. As it will be also shown in Section 5, it is possible, with some additional steps, to induce rules satisfying constraints on μ-consistency from positive regions computed using object consistency measure μ'. It should be noticed that there is a difference in the definitions of ϵ-consistency, ϵ'-consistency and μ-consistency, comparing to the corresponding definitions of object consistency measures ϵ, ϵ' and μ. In the definitions of rule consistency measures, $\underline{P}^{\theta_X}(X)$ is used instead of X. In this way, covered objects from X that do not belong to $POS_P^{\theta_X}(X)$ worsen the value of considered rule consistency measure. This is especially important when such objects belong to $NEG_P^{\theta_X}(X)$.

It is possible to induce decision rules from monotonic or non-monotonic lower approximations, i.e., probabilistic lower approximations computed using object consistency measures that have properties (m1), (m2), (m3), and (m4) or probabilistic lower approximations computed using measures that lack some of these properties, respectively. Monotonicity of rule consistency measure $\hat{\Theta}$ that is used in induction of set $R_X^{\hat{\theta}_X}$ affects the process of induction. Induction of rules from non-monotonic lower approximations requires additional steps to ensure desirable consistency of induced rules. As it will be shown in Section 5, it is computationally less expensive to induce rules from monotonic probabilistic lower approximations. Moreover, the rules induced from monotonic lower approximations may be more general since they explore larger elementary condition space, i.e., the set of possible elementary conditions that can be used in a rule is larger than in the non-monotonic case.

Now, let us introduce several concepts characteristic for machine learning and decision support approaches that apply a set of (decision) rules as a data model. We will also show how some of these concepts are adapted in rough set approaches, when one takes into account rough approximations of considered sets of objects.

Decision rule assigning to set X is *discriminant* if it covers only objects belonging to X. In IRSA and DRSA, a certain decision rule is discriminant if it covers only objects from $\underline{P}(X)$, while possible decision rule is discriminant if it covers only objects from $\overline{P}(X)$. Moreover, in variable consistency rough set approaches considered in this paper, rule is discriminant if it covers only objects belonging to positive region $POS_P^{\theta_X}(X)$. Rule is *minimal* if removing any of its elementary conditions causes that it is no more discriminant. We consider also minimality of a rule in the context of all rules from given set $\mathbf{R}$. In this context, rule r is minimal if there is no other rule r' with not less general conditions and not less specific decision. Using the notation introduced in Section 3, $r_X^{\hat{\theta}_X}$ is minimal if there does not exist other rule $r_Y^{\hat{\theta}_Y} \in \mathbf{R}$, $Y \subseteq U$, such that $\|\Phi_{r_Y^{\hat{\theta}_Y}}\| \supseteq \|\Phi_{r_X^{\hat{\theta}_X}}\|$ and $\|\Psi_{r_Y^{\hat{\theta}_Y}}\| \subseteq \|\Psi_{r_X^{\hat{\theta}_X}}\|$. Set of rules assigning to X is *complete* iff each object $y \in X$ is covered by at least one rule from this set. In the rough set approaches, however, we consider completeness of the set of rules from the view point of lower and/or upper approximation of X. In particular, in VC-IRSA and VC-DRSA, set of rules $R_X^{\hat{\theta}_X}$ is complete iff each object $y \in \underline{P}^{\theta_X}(X)$ is covered by at least one rule $r_X^{\hat{\theta}_X} \in R_X^{\hat{\theta}_X}$. Finally, rule r belonging to the set of rules assigning to X is *non-redundant*, if removing r causes that this set ceases to be complete.

According to the rule induction strategy used in AQ [39, 40], as well as in FOIL [48, 50], induced rules should be minimal and discriminant and the set of rules should be complete. These requirements are satisfied by most of decision rule induction algorithms proposed for rough set approaches, e.g., LEM2, Dom-LEM [25, 32–34, 57]. The requirement of completeness is, however, softened in case of pruned sets of rules induced by IREP [21], RIPPER [13] or SLIPPER [14]. In other cases, like Lightweight Rule Induction (LRI) [60], a given number of rules is induced for each set X which also leads to softening the requirement of completeness. This is also true for statistical approach to rule learning [51], where it is assumed that the number of induced rules is parameterized. Moreover, the requirement to use discriminant rules is usually softened in a voting setting. In this setting, a set of rules is typically seen as an ensemble of rules, i.e., one assigns a weight to each rule and uses a voting scheme for prediction. This is the case, e.g., for SLIPPER, LRI and a statistical approach to rule learning [51].

Rule induction methods that do not require discrimination of rules and/or completeness of the set of rules proved to be successful in classification. Thus, these features do not seem to be necessary to build an accurate classifier. On the other hand, classifiers that skip these requirements are less useful when it comes to comprehensibility or transparency of their responses. Inclination towards "glass-box" methods, as opposed to "black-box" approaches, is frequently postulated by researchers in many fields of artificial intelligence [18, 19, 31]. Not only a precise response of a classifier but also interpretable justification of presented suggestion is considered to be important.

5 Induction of decision rules by sequential covering in VC-DomLEM

So far, we have given the description of decision rules together with their characteristics and properties. The remaining task is to describe the algorithm for inducing rules. The proposed algorithm, called VC-DomLEM, induces rules for classification problems addressed in VC-IRSA and ordinal classification problems considered in VC-DRSA. It can be also easily adapted to induce certain, possible and approximate rules in IRSA, as well as certain and possible rules in DRSA. Moreover, it can be used to generate rules from *pairwise comparison table* considered in DRSA and VC-DRSA when solving choice or ranking problems [23, 55]. This algorithm heuristically searches for rules that satisfy given threshold value of one of rule consistency measures (25), (26) or (27). The applied heuristic strategy is called sequential covering [35] or separate and conquer [20, 38, 46]. It constructs a rule that covers a subset of training objects, removes the covered objects from the training set and iteratively learns another rule that covers some of the remaining objects, until no uncovered objects remain. This strategy has been previously applied in AQ family of algorithms, CN2, LEM, IREP, RIPPER and DomLEM.

VC-DomLEM induces a complete set of minimal and non-redundant decision rules $\mathbf{R}$. This algorithm operates at two levels. At the first level, presented as Algorithm 1, set of rules $R_X^{\hat{\theta}_X}$ is induced for each X by the $VC\text{-}SequentialCoverig^{mix}$ method, presented as Algorithm 2. $VC\text{-}SequentialCoverig^{mix}$ is inducting rules using elementary conditions constructed on attributes from set P (line 4). Value of chosen rule consistency measure $\hat{\Theta}$ has to be not worse than given threshold value $\hat{\theta}_X$. Moreover, each rule from set $R_X^{\hat{\theta}_X}$ is allowed to cover only those objects which belong to set $AO_P^{\theta_X}(X)$. This set is calculated according to one of three options coded by parameter $s \in \{1, 2, 3\}$ (line 3). We consider three reasonable options, indicated by the value of s: 1) $AO_P^{\theta_X}(X) = POS_P^{\theta_X}(X)$, 2) $AO_P^{\theta_X}(X) = POS_P^{\theta_X}(X) \cup BND_P^{\theta_X}(X)$, and 3) $AO_P^{\theta_X}(X) = U$. Option 1) is concordant with the spirit of DRSA. Option 3) implies induction of the most general rules. However, a rule induced for this option may cover objects from negative region $NEG_P^{\theta_X}(X)$. Option 2) is in between – it implies induction of more general rules than according to 1) and still prevents covering objects from the negative region. Set of rules $R_X^{\hat{\theta}_X}$ is added to set $\mathbf{R}$ in line 5. Minimality of set $\mathbf{R}$ is checked after each addition in line 6. In fact, minimality check is necessary only for VC-DRSA, where unions of ordered classes can overlap. Moreover, this step can be simplified if in line 2 upward or downward unions are considered from the most specific (i.e., containing the smallest number of objects) to the most general (i.e., containing the largest number of objects). In such a case, only rules from set $R_X^{\hat{\theta}_X}$ can be non-minimal.

Algorithm 1: *VC-DomLEM*

Input : set $\mathbf{X}$ of classes $X_i \in U$, upward unions of classes $X_i^{\geq} \in U$ or downward unions of classes $X_i^{\leq} \in U$,
set of attributes $P \subseteq C$,
rule consistency measure $\hat{\Theta}$,
set of rule consistency measure thresholds $\{\hat{\theta}_X : X \in \mathbf{X}\}$,
object covering option s.

Output: set of rules $\mathbf{R}$.

$\mathbf{R} := \emptyset$;
foreach *element* $X \in \mathbf{X}$ **do**
  $AO_P^{\theta_X}(X) := AllowedObjects(X, P, \theta_X, s)$;
  $R_X^{\hat{\theta}_X} := VC\text{-}SequentialCovering^{mix}(\underline{P}^{\theta_X}(X), AO_P^{\theta_X}(X), P, \hat{\Theta}, \hat{\theta}_X)$;
  $\mathbf{R} := \mathbf{R} \cup R_X^{\hat{\theta}_X}$;
  RemoveNonMinimalRules($\mathbf{R}$);
end

At the second level, rules for a given set X are induced by $VC\text{-}SequentialCoverig^{mix}$ method, presented as Algorithm 2. These rules consist of elementary conditions that are constructed using evaluations of objects from $\underline{P}^{\theta_X}(X)$ on attributes from set P (line 5). The word *mix* in the name of the algorithm is used to indicate that each elementary condition can be constructed from among evaluations of different positive objects (i.e., objects from set $\underline{P}^{\theta_X}(X)$). For regular attributes, elementary conditions involve relation $=$. In case of criteria, elementary conditions involve relation $\succeq$ or $\preceq$, for an upward or downward union of classes, respectively. The induction of rules is carried out as long as there are still some positive objects to be covered, i.e., there are uncovered objects from $\underline{P}^{\theta_X}(X)$ that can be used to construct elementary conditions (line 3). Each rule is constructed in a greedy search by adding new elementary conditions as long as consistency threshold $\hat{\theta}_X$ is not satisfied by the chosen rule consistency measure $\hat{\Theta}$, or rule $r_X^{\hat{\theta}_X}$ covers objects not belonging to set $AO_P^{\theta_X}(X)$ (line 6). The elementary condition added to rule $r_X^{\hat{\theta}_X}$ in line 8 is a new condition from set EC (i.e., condition that is not already present in the constructed rule) that is evaluated as the best in line 7. In order to evaluate elementary condition $ec \in EC$, the following two quality measures are used:

1. one of rule consistency measures (25), (26) or (27) of rule $r_X^{\hat{\theta}_X} \cup ec$,
2. $\left| \|\Phi_{r_X^{\hat{\theta}_X} \cup ec}\| \cap \underline{P}^{\theta_X}(X) \right|$,

where $r_X^{\hat{\theta}_X} \cup ec$ denotes a rule resulting from extension of rule $r_X^{\hat{\theta}_X}$ by new elementary condition ec.

The best elementary condition according to 1) is selected. In case of a tie between compared elementary conditions, the best one according to 2) is chosen. If this is not sufficient to determine the best condition, the order in which elementary conditions are checked decides. It is worth noting that it is possible to add a new elementary condition on an attribute which is already present in the rule. When such a new elementary condition is added, previous elementary condition on that attribute becomes redundant and is removed in line 11. This allows to start with a rule as general as possible, and then specialize it to meet constraint on rule consistency measure checked in line 6. After elementary condition is added to the rule (line 8), the set of candidate elementary conditions EC is updated (line 9). All elementary conditions that come from objects that do not support the growing rule are removed from EC. In this way, the search for new elementary conditions is narrowed to only these conditions that can be constructed from objects in $supp(r_X^{\hat{\theta}_X})$. This also causes that addition of a new elementary condition on the attribute already present in the rule can only result in a more specific rule (i.e., a rule that covers a subset of objects covered so far).

After the constructed rule satisfies necessary constraints from line 6, elementary conditions that became redundant are removed from that rule (line 11). This can be done in different ways (e.g., elementary conditions can be considered from the oldest to the newest ones). However, it needs to be assured that after this step the rule still satisfies constraints from line 6. Next, the rule is added to the set of rules induced so far (line 12). Objects supporting that rule are removed from set B, which is the base for building candidate elementary conditions (line 13).

Constructed set of rules $R_X^{\hat{\theta}_X}$ is checked for redundancy in line 15. The rules considered as redundant are removed. They are removed in an iterative procedure which consists of three steps. First, each rule that can be removed is put on a list. If the list is non-empty, then one of the rules can be removed without loosing completeness of $R_X^{\hat{\theta}_X}$. Otherwise, the checking is stopped. Second, one rule $r_X^{\hat{\theta}_X}$ is selected from the list according to the following measures, considered lexicographically:

1. the smallest value of $\left|\|\Phi_{r_X^{\hat{\theta}_X}}\| \cap \underline{P}^{\theta_X}(X)\right|$,
2. the worst value of $\hat{\Theta}(r_X^{\hat{\theta}_X})$,
3. the smallest index of $r_X^{\hat{\theta}_X}$ on the constructed list of rules.

Third, the selected rule is removed from set $R_X^{\hat{\theta}_X}$.

5.1 Induction of rules satisfying ϵ-consistency and ϵ'-consistency condition

Monotonicity properties of rule consistency measures: ϵ-consistency (25) and ϵ'-consistency (26), allow to increase efficiency of rule induction in $VC\text{-}SequentialCoverig^{mix}$ algorithm. These properties are derived from corresponding object consistency measures ϵ (13) and ϵ' (16).

There are two scenarios defined for $VC\text{-}SequentialCoverig^{mix}$ algorithm:

α) application of ϵ-consistency measure in order to induce rules covering objects from $\underline{P}^{\theta_X}(X)$ calculated using ϵ or ϵ^* object consistency measure,

Algorithm 2: *VC-SequentialCovering*^{mix}

Input : set of positive objects $\underline{P}^{\theta_X}(X) \subseteq U$,
set of objects that can be covered $AO_P^{\theta_X}(X) \subseteq U$, $AO_P^{\theta_X}(X) \supseteq \underline{P}^{\theta_X}(X)$
set of attributes $P \subseteq C$,
rule consistency measure $\hat{\Theta}$,
rule consistency measure threshold $\hat{\theta}_X$.

Output: set of rules $R_X^{\hat{\theta}_X}$ assigning objects to X.

$B := \underline{P}^{\theta_X}(X)$;
$R_X^{\hat{\theta}_X} := \emptyset$;
while $B \neq \emptyset$ **do**
  $r_X^{\hat{\theta}_X} := \emptyset$;
  $EC :=$ ElementaryConditions(B, P);
  while *($\hat{\Theta}(r_X^{\hat{\theta}_X})$ does not satisfy $\hat{\theta}_X$) or ($\|\Phi_{r_X^{\hat{\theta}_X}}\| \nsubseteq AO_P^{\theta_X}(X)$)* **do**
    $ec :=$ BestElementaryCondition(EC, $r_X^{\hat{\theta}_X}$, $\hat{\Theta}$, $\underline{P}^{\theta_X}(X)$);
    $r_X^{\hat{\theta}_X} := r_X^{\hat{\theta}_X} \cup ec$;
    $EC :=$ ElementaryConditions($B \cap supp(r_X^{\hat{\theta}_X})$, P);
  end
  RemoveRedundantElementaryConditions($r_X^{\hat{\theta}_X}$, $\hat{\Theta}$, $\hat{\theta}_X$, $AO_P^{\theta_X}(X)$);
  $R_X^{\hat{\theta}_X} := R_X^{\hat{\theta}_X} \cup r_X^{\hat{\theta}_X}$;
  $B := B \setminus supp(r_X^{\hat{\theta}_X})$;
end
RemoveRedundantRules($R_X^{\hat{\theta}_X}$, $\hat{\Theta}$, $\underline{P}^{\theta_X}(X)$);

β) application of ϵ'-consistency measure in order to induce rules covering objects from $\underline{P}^{\theta_X}(X)$ calculated using ϵ' object consistency measure.

Moreover,

γ) elementary condition ec is selected according to the following two measures, considered lexicographically:

1) rule consistency measure $\hat{\Theta}$ of rule $r_X^{\hat{\theta}_X} \cup ec$ being ϵ-consistency in scenario α) or ϵ'-consistency in scenario β),
2) $\big|\|\Phi_{r_X^{\hat{\theta}_X} \cup ec}\| \cap \underline{P}^{\theta_X}(X)\big|$.

Theorem 1. *For VC-SequentialCovering^{mix}, in scenario α) or β), and subject to γ), sequential addition of the best elementary condition always leads to decision rule $r_X^{\hat{\theta}_X}$ that has value of chosen rule consistency measure $\hat{\Theta}$ not worse than threshold $\hat{\theta}_X$, where $\hat{\theta}_X = \frac{|\neg X|}{|\neg \underline{P}^{\epsilon_X}(X)|}\epsilon_X$ (or $\hat{\theta}_X = \frac{|\neg X|}{|\neg \underline{P}^{\epsilon^*_X}(X)|}\epsilon^*_X$, respectively) in scenario α) or $\hat{\theta}_X = \frac{|X|}{|\underline{P}^{\epsilon'_X}(X)|}\epsilon'_X$ in scenario β).*

Proof. Let us assume that induced rule $r_X^{\hat{\theta}_X}$ does not satisfy yet the constraint on rule consistency measure from line 6 of Algorithm 2. Elementary conditions from set EC are constructed, in line 9, using evaluations of objects that belong to the set of positive objects B and that are covered by $r_X^{\hat{\theta}_X}$. Thus, in the worst case, this method constructs $r_X^{\hat{\theta}_X}$ that is composed of elementary conditions that use all evaluations from one object y belonging to B. This results in $r_X^{\hat{\theta}_X}$ that corresponds to the P-dominance cone based on y. Since y belongs to $\underline{P}^{\theta_X}(X)$, y has value of Θ not worse than θ_X. This implies that rule $r_X^{\hat{\theta}_X}$ has value of $\hat{\Theta}$ not worse than threshold $\hat{\theta}_X$. □

As it was proved in [9], both ϵ and ϵ' have property (m1). This property is also satisfied by related rule consistency measures ϵ-consistency and ϵ'-consistency. When combined with the greedy nature of the presented algorithm, it allows to consider for addition to rule $r_X^{\hat{\theta}_X}$ being constructed only new elementary conditions constructed on attributes that are not already present in the rule. New elementary condition constructed on an attribute already present in the rule decreases the quality of that rule, measured by its consistency and the number of covered objects from the probabilistic P-lower approximation of X, as shown by the following theorem.

Theorem 2. *For VC-SequentialCovering^{mix}, in scenario α) or β), and subject to γ), addition of a new (more specific) elementary condition on some attribute that is already present in the induced rule $r_X^{\hat{\theta}_X}$ does not change the value of rule consistency measure while it decreases support of that rule.*

Proof. Let us assume that induced rule $r_X^{\hat{\theta}_X}$ does not satisfy yet the constraint on rule consistency measure from line 6 of Algorithm 2. Moreover, let us assume that it already involves elementary conditions constructed on attributes from set R, $R \subset P \subseteq C$, $R \neq \emptyset$. At each step, best elementary condition ec was selected to extend the rule so that the resulting rule covered the lowest number of objects not belonging to $\underline{P}^{\theta_X}(X)$ (i.e., value of ϵ-consistency or ϵ'-consistency measure of the resulting rule was minimized) and, in case of a tie between considered elementary conditions, the highest number of objects from $\underline{P}^{\theta_X}(X)$. For attribute $q_i \in R$, next (more specific) elementary condition on that attribute has to decrease support of the induced rule. In order to prove that the new elementary condition on attribute $q_i \in R$ can not change the value of rule consistency measure, let us denote by ec_1 the first elementary condition on the considered attribute, and by ec_2 the new (more specific) elementary condition on that attribute. Let us observe that due to the greedy nature of the algorithm, at the time when ec_1 was chosen, ec_2 had to be evaluated as not better than ec_1 according to the value of rule consistency measure. This means that the difference DF between the set of objects covered by rule $r_X^{\hat{\theta}_X} \cup ec_1$ and the set of objects covered by rule $r_X^{\hat{\theta}_X} \cup ec_2$ could not contain any object not belonging to $\underline{P}^{\theta_X}(X)$. According to Algorithm 2, removal of elementary conditions from a rule is not permitted until it satisfies constraints from line 6. Thus, at any time after the rule is extended with elementary condition ec_1, we have $\|\Phi_{r_X^{\hat{\theta}_X}}\| - \|\Phi_{r_X^{\hat{\theta}_X} \cup ec_2}\| \subseteq DF$. Because $DF \cap \neg \underline{P}^{\theta_X}(X) = \emptyset$, value of rule consistency measure is not altered by addition of ec_2. $\square$

Theorem 2 shows that during rule induction by Algorithm 2, elementary conditions constructed on attributes that are already present in the rule are redundant from the viewpoint of ϵ-consistency and ϵ'-consistency measures. Moreover, such elementary conditions decrease the support of the rule. Thus, we can reduce the number of elementary conditions considered to be added to the constructed rule to only those on attributes that are not already present in the rule. The computational benefit coming from this reduction is hard to estimate. Anyway, this improvement does not involve any additional cost (i.e., it does not involve any additional steps to reduce the number of considered elementary conditions).

Measures ϵ and ϵ' both have property (m4). This allows us to further increase the efficiency of the rule induction algorithm. We can sort elementary conditions on each criterion $q_i \in P$, where $P \subseteq C$, according to the preference order on its values. Property (m4) assures that the order of elementary conditions after sorting reflects the order of values of object consistency measures ϵ and $\epsilon^{'}$. The remaining processing after the sorting is simple because we search for elementary conditions with the best value of chosen rule consistency measure. The additional computational cost of a one-time sort of each attribute is a fixed cost that is almost inconsequential when compared to the overall computational cost of induction of the rules. This improvement considerably reduces computational cost of rule induction. As it was shown in [60], a similar improvement resulted in computational complexity of induction approximately linear in the number of rules or objects.

ϵ-consistency measure can be used to induce decision rules for objects belonging to $\underline{P}^{\epsilon^*_{X_i^{\geq}}}(X_i^{\geq})$ (or $\underline{P}^{\epsilon^*_{X_i^{\leq}}}(X_i^{\leq})$). From definition (14), $\epsilon^{*P}_{X_i^{\geq}}(y) \geq \epsilon^{P}_{X_i^{\geq}}(y)$, $\forall y \in U, X_i^{\geq} \subseteq U, P \subseteq C$. If some object $y \in U$ belongs to $\underline{P}^{\epsilon^*_{X_i^{\geq}}}(X_i^{\geq})$, then it also belongs to $\underline{P}^{\epsilon_{X_i^{\geq}}}(X_i^{\geq})$, with $\epsilon_{X_i^{\geq}} = \epsilon^*_{X_i^{\geq}}$. In other words, for given object consistency measure threshold value $\theta_{X_i^{\geq}}$, probabilistic P-lower approximation of union $X_i^{\geq}$ calculated w.r.t. measure ϵ is a superset of probabilistic P-lower approximation of union $X_i^{\geq}$ calculated w.r.t. measure ϵ^*. Since it is possible to cover by rules all objects belonging to the former, it is also possible to cover by rules all objects belonging to the latter.

5.2 Induction of rules satisfying μ-consistency condition

VC-DomLEM algorithm needs some modifications to enable induction of rules satisfying a constraint on μ-consistency measure. These modifications are caused by lack of monotonicity property (m4) of μ-consistency measure. Notice that μ-consistency measure is also missing property (m1), however, this is already handled in VC-DomLEM algorithm by the possibility of adding a new elementary condition on the attribute which is already present in the induced rule. If an elementary condition covering too many objects not belonging to P-positive region of X is selected in some iteration, it can always be narrowed down later to cover fewer of them. Nevertheless, if this possibility is used in the algorithm frequently, it can increase the computational cost considerably.

Now, let us consider induction of rules, which satisfy constraint on μ-consistency measure, from probabilistic P-lower approximations calculated using object consistency measure $\mu^{'}$, defined as (17) or (18). The problem that can be faced by VC-DomLEM during induction of rules is presented in the following Example 1 and Fig. 1.

Example 1. When applying in equation (1) object consistency measure μ' defined as (17), and choosing gain-threshold $\theta_{X_2^{\geq}} = 0.75$, we obtain $\underline{P}^{0.75}(X_2^{\geq}) = \{y_1, y_2, y_3\}$, where $P = \{q_1, q_2\}$. One can observe that objects belonging to union $X_2^{\geq}$ are characterized by the following values of rough membership measure: $\mu(y_1) = 0.75$, $\mu(y_2) = 0.66$, $\mu(y_3) = 0.5$. Objects y_2 and y_3 belong to $\underline{P}^{0.75}(X_2^{\geq})$ because they dominate object y_1. Moreover, according to definition (10), $POS_P^{0.75}(X_2^{\geq}) = \{y_1, y_2, y_3, y_6\}$.

Now, we intend to construct decision rules assigning to union of classes $X_2^{\geq}$. For this purpose, we apply rule μ-consistency measure, defined as (27). We take $\hat{\theta}_{X_2^{\geq}} = \theta_{X_2^{\geq}} = 0.75$ and construct elementary conditions using evaluations of objects belonging to $\underline{P}^{0.75}(X_2^{\geq})$, in order to cover objects from $POS_P^{0.75}(X_2^{\geq})$ only (i.e., we assume the most restrictive object covering option, corresponding to $s = 1$). For attribute q_1, considered elementary conditions have the following values of μ-consistency measure: 0.6 for $q_1(y) \geq 2$, 0.6(6) for $q_1(y) \geq 4$ and 0.5 for $q_1(y) \geq 5$. It is visible that μ-consistency measure does not have property (m4) since it is a gain-type measure and its value for $q_1(y) \geq 5$ is lower than for $q_1(y) \geq 4$. The first elementary condition selected by VC-DomLEM for rule $r^{0.75}_{X_2^{\geq}}$ is $q_1(y) \geq 4$. This elementary condition has value of μ-consistency measure equal to 0.6(6). The constraint on rule consistency from line 6 of VC-$SequentialCoverig^{mix}$ is not satisfied. Unfortunately, any elementary condition that can be further added to the induced rule does not help to satisfy that constraint. The best elementary condition that can be added in the second iteration is $q_2(y) \geq 4$, resulting in a rule *if* $q_1(y) \geq 4 \wedge q_2(y) \geq 4$ *then* $y \in X_2^{\geq}$, with μ-consistency of 0.6(6). Thus, in the current form, it is impossible to construct by VC-DomLEM algorithm a rule that satisfies threshold on μ-consistency measure. Such rule would be *if* $q_1(y) \geq 2 \wedge q_2(y) \geq 2$ *then* $y \in X_2^{\geq}$, with μ-consistency 0.75.

Note that the possibility to add elementary condition on an attribute already present in the rule does not solve the problem resulting from the lack of property (m4). It allows only to specialize elementary conditions already present in the rule. To overcome the lack of property (m4) of μ-consistency measure, we propose a reduction of the set of objects considered when creating elementary conditions.

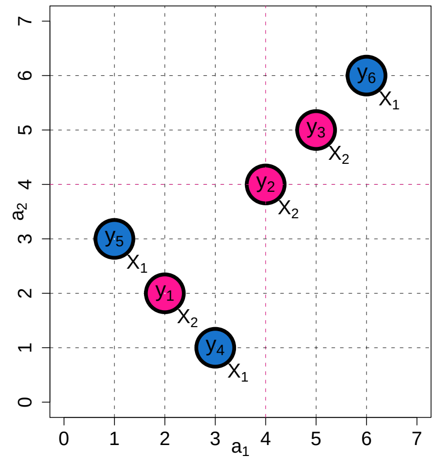

Fig. 1. Illustration of VC-DomLEM problems with induction of rules satisfying μ-consistency condition, caused by lack of property (m4).

We define P-edge regions of unions of classes $X_i^{\geq}$ and $X_i^{\leq}$. For $P \subseteq C, X_i^{\geq}, X_i^{\leq} \subseteq U$, $y, z \in U$, $\theta_{X_i^{\geq}} \in [0, A_{X_i^{\geq}}], \theta_{X_i^{\leq}} \in [0, A_{X_i^{\leq}}]$, P-edge regions are defined as follows:

$$EDGE_P^{\theta_{X_i^{\geq}}}(X_i^{\geq}) = \{y \in \underline{P}^{\theta_{X_i^{\geq}}}(X_i^{\geq}) : z \in D_P^-(y) \cap \underline{P}^{\theta_{X_i^{\geq}}}(X_i^{\geq}) \Rightarrow z \in D_P^+(y)\}, \tag{28}$$

$$EDGE_P^{\theta_{X_i^{\leq}}}(X_i^{\leq}) = \{y \in \underline{P}^{\theta_{X_i^{\leq}}}(X_i^{\leq}) : z \in D_P^+(y) \cap \underline{P}^{\theta_{X_i^{\leq}}}(X_i^{\leq}) \Rightarrow z \in D_P^-(y)\}. \tag{29}$$

It should be noticed that the P-edge region of union $X_i^{\geq}$ is a subset of probabilistic P-lower approximation of that union. This subset contains only objects that do not (at least) weakly dominate any other object belonging to $\underline{P}^{\theta_{X_i^{\geq}}}(X_i^{\geq})$. Analogically, the P-edge region of union $X_i^{\leq}$ contains only objects that are not (at least) weakly dominated by any other object belonging to $\underline{P}^{\theta_{X_i^{\leq}}}(X_i^{\leq})$. We say that object y *weakly dominates* object z iff y is not worse than z on each criterion $q_i \in P$, for at least one criterion $q_i \in P$ is strictly better, and for each regular attribute $q_i \in P$ is indifferent to z. We say that object y is *weakly dominated* by object z iff y is not better than z on each criterion $q_i \in P$, for at least one criterion $q_i \in P$ is strictly worse, and for each regular attribute $q_i \in P$ is indifferent to z.

Let us consider the following scenario for VC-$SequentialCoverig^{mix}$ algorithm:

α') application of μ-consistency measure in order to induce rules covering objects from $\underline{P}^{\theta_X}(X)$ calculated using μ' object consistency measure.

In order to adjust $VC\text{-}SequentialCoverig^{mix}$ algorithm for μ-consistency measure, we need the following modifications:

γ') elementary condition ec is selected according to the following two measures, considered lexicographically:
   (a) μ-consistency measure of rule $r_X^{\hat{\theta}_X} \cup ec$,
   (b) $\left|\|\Phi_{r_X^{\hat{\theta}_X} \cup ec}\| \cap \underline{P}^{\theta_X}(X)\right|$,

δ) P-edge region of set X is used instead of the probabilistic P-lower approximation of X.

Theorem 3. *For $VC\text{-}SequentialCoverig^{mix}$ method, in scenario α'), and subject to γ') and δ), sequential addition of the best elementary condition always leads to decision rule $r_X^{\hat{\theta}_X}$ that has value of μ-consistency measure not lower than threshold $\hat{\theta}_X = \theta_X$.*

Proof. Because of the definition of object consistency measure μ', the objects that are included in the P-edge region of X are only those that have value of rough membership not lower than the specified threshold θ_X. Proposed reduction of the set of objects, together with the possibility to add next elementary condition on an attribute that is already present in the induced rule, guarantee that each rule induced for set X can finally reach the value of μ-consistency measure not worse than threshold $\hat{\theta}_X = \theta_X$. It is true because one can always construct a rule that has all elementary conditions generated from exactly one of the objects belonging to the P-edge region of X. □

Presented modification of VC-DomLEM algorithm implies additional computational cost because P-edge regions must be calculated. On the other hand, smaller set of objects is considered to generate elementary conditions used to construct a rule, which compensates the computational cost of selecting these objects. Moreover, as we have shown in Example 1, without this modification it might be impossible to induce rules having value of μ-consistency measure not lower than specified threshold $\hat{\theta}_X$.

6 Experimental Setup

The main aim of the experiment presented in this paper is to evaluate the usefulness of VC-DomLEM algorithm in terms of its predictive accuracy. To this end, we compared our algorithm to other methods on twelve ordinal data sets listed in Table 1. Data sets: employee rejection/acceptance (ERA), employee selection (ESL), lectures evaluation (LEV) and social workers decisions (SWD) were taken from [2]. Other data sets come from the UCI repository[3] and other public repositories (as in case of data sets: bank of Greece (bank-g) and financial analysis made easy (fame)).

[3] see http://www.ics.uci.edu/˜mlearn/MLRepository.html

Table 1. Characteristics of data sets

Id	Data set	Objects	Attributes	Classes
1	breast-c	286	8	2
2	breast-w	699	9	2
3	car	1296	6	4
4	cpu	209	6	4
5	bank-g	1411	16	2
6	fame	1328	10	5
7	denbosch	119	8	2
8	ERA	1000	4	9
9	ESL	488	4	9
10	LEV	1000	4	5
11	SWD	1000	10	4
12	windsor	546	10	4

In general, it is not always the case that ordinal classifiers that preserve monotonicity constraints perform better than non-ordinal classifiers [4] in predictive accuracy. This is mainly attributed to the fact that monotonicity constraints that need to be satisfied bias the classifier. The unbiased classifier may generalize the data more effectively. Taking this into account, we compared VC-DomLEM to other ordinal classifiers that preserve monotonicity constraints as well as to non-ordinal classifiers.

Let us now describe briefly the classifiers that we considered in the experiment. We used the implementation of VC-DomLEM from jRS and jMAF frameworks[4]. We considered VC-DomLEM in two variants: monotonic (i.e., with monotonic ϵ-consistency or ϵ'-consistency measure) and non-monotonic (i.e., with non-monotonic μ-consistency measure). Moreover, we used two ordinal classifiers that preserve monotonicity constraints, namely: Ordinal Learning Model (OLM) [1, 3] and Ordinal Stochastic Dominance Learner (OSDL) [10]. We also used some well known non-ordinal classifiers: Naive Bayes, Support Vector Machine (SVM) with linear kernel [47], decision rule classifier RIPPER [13], and decision tree classifier C4.5 [49].

7 Results of Experiment and Discussion

In the experiment, the predictive accuracy was estimated by stratified 10-fold cross-validation, which was repeated several times. We measured mean absolute error (MAE), which is a standard measure used for ordinal classification problems. Additionally, we measured the average percentage of correctly classified objects. These results are shown in Tables 2 and 3, respectively. In both cases, tables with results contain the value of measure and its standard deviation for each data set and each classifier. Moreover, for each data set we calculated a

[4] see http://www.cs.put.poznan.pl/jblaszczynski/Site/jRS.html

rank of the result of a classifier when compared to the other classifiers. The rank is presented in brackets (the smaller the rank, the better). We show these ranks because they are used in statistical test described further. Last row of each table shows the average rank obtained by a given classifier. Moreover, for each data set, the best value of the predictive accuracy measure, and those values which are within standard deviation of the best one, are marked as bold.

Table 2. Mean absolute error (MAE) results

Id	monotonic VC-DomLEM	non-monotonic VC-DomLEM	Naive Bayes	SVM	RIPPER	C4.5	OLM	OSDL
1	**0.2331** (1) ±0.003297	0.2436 (3) ±0.007185	0.2564 (4) ±0.005943	0.3217 (7) ±0.01244	0.2960 (5) ±0.01154	0.2424 (2) ±0.003297	0.324 (8) ±0.01835	0.3065 (6) ±0.001648
2	0.03720 (2) ±0.002023	0.04578 (6) ±0.003504	0.03958 (3) ±0.0006744	**0.03243** (1) ±0.0006744	0.04483 (5) ±0.004721	0.05532 (7) ±0.00751	0.1764 (8) ±0.00552	0.04149 (4) ±0.001168
3	**0.03421** (1) ±0.0007275	0.03524 (2) ±0.0009624	0.1757 (7) ±0.002025	0.08668 (4) ±0.002025	0.2029 (8) ±0.01302	0.1168 (6) ±0.003108	0.09156 (5) ±0.005358	0.04141 (3) ±0.0009624
4	**0.08293** (1) ±0.01479	**0.0925** (2) ±0.01579	0.1707 (5) ±0.009832	0.4386 (8) ±0.01579	0.1611 (4) ±0.01372	0.1196 (3) ±0.01790	0.3461 (7) ±0.02744	0.3158 (6) ±0.01034
5	**0.04559** (1) ±0.001456	0.04867 (2) ±0.000884	0.1146 (6) ±0.01371	0.1280 (7) ±0.001205	0.0489 (3) ±0.00352	0.0515 (4) ±0.005251	0.05528 (5) ±0.001736	0.1545 (8) ±0
6	**0.3406** (1.5) ±0.001878	0.3469 (3) ±0.004	0.4829 (6) ±0.002906	**0.3406** (1.5) ±0.001775	0.3991 (5) ±0.003195	0.3863 (4) ±0.005253	1.577 (7) ±0.03791	1.592 (8) ±0.007555
7	**0.1232** (1) ±0.01048	**0.1289** (2.5) ±0.01428	**0.1289** (2.5) ±0.01428	0.2129 (7) ±0.003961	0.1737 (6) ±0.02598	0.1653 (5) ±0.01048	0.2633 (8) ±0.02206	0.1541 (4) ±0.003961
8	1.307 (2) ±0.002055	1.376 (7) ±0.002867	1.325 (5) ±0.003771	1.318 (3) ±0.007257	1.681 (8) ±0.01558	1.326 (6) ±0.006018	1.321 (4) ±0.01027	**1.280** (1) ±0.00704
9	0.3702 (3) ±0.01352	0.4146 (5) ±0.005112	**0.3456** (2) ±0.003864	0.4262 (6) ±0.01004	0.4296 (7) ±0.01608	0.3736 (4) ±0.01089	0.474 (8) ±0.01114	**0.3422** (1) ±0.005019
10	0.4813 (6) ±0.004028	0.5213 (7) ±0.002055	0.475 (5) ±0.004320	0.4457 (4) ±0.003399	0.4277 (3) ±0.00838	0.426 (2) ±0.01476	0.615 (8) ±0.0099	**0.4033** (1) ±0.003091
11	0.454 (4) ±0.004320	0.498 (7) ±0.004546	0.475 (6) ±0.004320	0.4503 (2) ±0.002867	0.452 (3) ±0.006481	0.4603 (5) ±0.004497	0.5707 (8) ±0.007717	**0.433** (1) ±0.002160
12	**0.5024** (1) ±0.006226	0.5201 (3) ±0.003956	0.5488 (4) ±0.005662	0.5891 (6) ±0.02101	0.6825 (8) ±0.03332	0.652 (7) ±0.03721	0.5757 (5) ±0.006044	0.5153 (2) ±0.006044
	2.04	4.12	4.62	4.71	5.42	4.58	6.75	3.75

We used a statistical approach to compare differences in predictive accuracy between classifiers in variants which we mentioned above. First, we applied Friedman test to globally compare performance of eight different classifiers on multiple data sets [16, 37]. The null-hypothesis in this test was that all compared classifiers perform equally well. It was tested using the ranks of each of the classifiers on each of the data sets. We do not present complete post-hoc analysis [16] of differences between classifiers, however, we show the average rank of each classifier in the last row of the tables with results.

We analyzed the ranks of MAE, which are presented in Table 2. The p-value in Friedman test performed for this comparison is 0.00017. Then, we analyzed ranks of percentage of correctly classified objects, which are presented in Table 3. The p-value in Friedman test is in this case 0.00018. The results of Friedman test

Table 3. Percentage of correctly classified objects results

Id	monotonic VC-DomLEM	non-monotonic VC-DomLEM	Naive Bayes	SVM	RIPPER	C4.5	OLM	OSDL
1	**76.69** (1) ±0.3297	75.64 (3) ±0.7185	74.36 (4) ±0.5943	67.83 (7) ±1.244	70.4 (5) ±1.154	75.76 (2) ±0.3297	67.6 (8) ±1.835	69.35 (6) ±0.1648
2	96.28 (2) ±0.2023	95.42 (6) ±0.3504	96.04 (3) ±0.06744	**96.76** (1) ±0.06744	95.52 (5) ±0.4721	94.47 (7) ±0.751	82.36 (8) ±0.552	95.85 (4) ±0.1168
3	**97.15** (1) ±0.063	**97.1** (2) ±0.1311	84.72 (7) ±0.1667	92.18 (4) ±0.2025	84.41 (8) ±1.309	89.84 (6) ±0.1819	91.72 (5) ±0.4425	96.53 (3) ±0.063
4	**91.7** (1) ±1.479	**90.75** (2) ±1.579	83.41 (5) ±0.9832	56.62 (8) ±1.579	84.69 (4) ±1.409	88.52 (3) ±1.409	68.58 (7) ±2.772	72.41 (6) ±1.479
5	**95.44** (1) ±0.1456	95.13 (2) ±0.0884	88.54 (6) ±1.371	87.2 (7) ±0.1205	95.11 (3) ±0.352	94.85 (4) ±0.5251	94.47 (5) ±0.1736	84.55 (8) ±0
6	**67.55** (1) ±0.4642	**67.1** (2.5) ±0.4032	56.22 (6) ±0.2328	**67.1** (2.5) ±0.2217	63.55 (5) ±0.5635	64.33 (4) ±0.5844	27.43 (7) ±0.7179	22.04 (8) ±0.128
7	**87.68** (1) ±1.048	**87.11** (2.5) ±1.428	**87.11** (2.5) ±1.428	78.71 (7) ±0.3961	82.63 (6) ±2.598	83.47 (5) ±1.048	73.67 (8) ±2.206	84.6 (4) ±0.3961
8	26.9 (2) ±0.3742	21.43 (7) ±0.1700	25.03 (3) ±0.2494	24.27 (5) ±0.2494	20 (8) ±0.4243	**27.83** (1) ±0.4028	23.97 (6) ±0.4643	24.7 (4) ±0.8165
9	66.73 (3) ±1.256	62.43 (6) ±1.139	67.49 (2) ±0.3483	62.7 (5) ±0.6693	61.61 (7) ±1.555	66.33 (4) ±0.6966	55.46 (8) ±0.7545	**68.3** (1) ±0.3483
10	55.63 (6) ±0.3771	52.43 (7) ±0.2055	56.17 (5) ±0.3399	58.87 (4) ±0.3091	60.83 (2) ±0.6128	60.73 (3) ±1.271	45.43 (8) ±0.8179	**63.03** (1) ±0.2625
11	56.43 (6) ±0.4643	51.67 (7) ±0.4497	56.57 (5) ±0.4784	**58.23** (2) ±0.2055	57.63 (3) ±0.66	57.1 (4) ±0.4320	47.83 (8) ±0.411	**58.6** (1) ±0.4243
12	54.58 (2) ±0.7913	52.93 (4) ±1.427	53.6 (3) ±0.2284	51.83 (5) ±1.813	44.08 (8) ±0.8236	47.99 (7) ±2.888	49.15 (6) ±0.7527	**55.37** (1) ±0.3763
	2.25	4.25	4.29	4.79	5.33	4.17	7	3.92

and observed differences in average ranks allow us to state with high confidence that there is a significant difference between compared classifiers.

We continued our experimental comparison with examination of importance of difference in predictive accuracy for each pair of classifiers. We applied Wilcoxon test [37] with null-hypothesis that the medians of results on all data sets of the two compared classifiers are equal. Let us remark, that in the paired tests ranks are assigned to the value of difference in the predictive accuracy between the two compared classifiers. First, we applied this test to MAE from Table 2. We can observe significant difference (p-values lower than 0.05) between monotonic VC-DomLEM and any other classifier except OSDL. The same is true for the following pairs: non-monotonic VC-DomLEM and OLM, Naive Bayes and OLM, C4.5 and RIPPER, C4.5 and OLM, OSDL and OLM. Then, we applied Wilcoxon test to percentage of correctly classified objects from Table 3. These results indicate significant differences between monotonic VC-DomLEM and any other classifier except C4.5 and OSDL. The same is true for following pairs: non-monotonic VC-DomLEM and OLM, Naive Bayes and OLM, RIPPER and OLM, C4.5 and RIPPER, C4.5 and OLM, OSDL and OLM.

It follows from the results of the experiment that monotonic VC-DomLEM is better than the other compared classifiers. It has the best value of the average rank of both predictive accuracy measures. However, when we compared monotonic VC-DomLEM to other classifiers in pairs, we were not able to show significant difference in predictive accuracy with respect to OSDL and also C4.5 (but only in case of percentage of correctly classified objects). On the other hand, non-monotonic VC-DomLEM is comparable to other classifiers except OLM. OLM is clearly the worst classifier in our experiment.

Table 4. Comparison of mean strength and length of rules induced by monotonic and non-monotonic versions of VC-DomLEM

	monotonic VC-DomLEM		non-monotonic VC-DomLEM	
Id	strength	length	strength	length
1	**0.243**	**1.857**	0.179	2.636
2	**0.306**	**2.600**	0.298	2.917
3	**0.129**	**3.836**	**0.129**	**3.836**
4	0.300	**1.968**	**0.301**	2.033
5	**0.129**	**2.216**	0.124	2.742
6	**0.143**	**2.430**	0.133	3.122
7	0.230	**2.182**	**0.260**	2.769
8	0.070	2.341	**0.262**	**2.000**
9	**0.356**	**1.864**	0.346	2.319
10	0.181	**2.377**	**0.207**	2.395
11	0.164	**2.602**	**0.179**	2.857
12	0.144	3.644	**0.149**	**3.534**

It is generally acknowledged that decision rules are relatively easy to interpret by users. Stronger and shorter rules are particularly relevant since they represent strongly established relationships between causes and effects. From this point of view, it is thus interesting to compare our two versions of VC-DomLEM – the monotonic and the non-monotonic ones. Table 4 summarizes this comparison. In Fig. 2 and Fig. 3, we present accumulative characteristics of the strength of induced rules. These figures show how many rules (in percentage of the respective set of rules) have at least given strength (in percentage). Moreover, in Fig. 4 and Fig. 5, we present accumulative characteristics of the length of induced rules. These figures show how many rules (in percentage of the respective set of rules) have at most given length. It can be observed that for majority of data sets the accumulative strength of induced rules is comparable. There are few exceptions from this rule: for brest-c data set, monotonic VC-DomLEM induced stronger rules than non-monotonic VC-DomLEM. On the other hand, for denbosch and ERA data sets, non-monotonic VC-DomLEM induced stronger rules than monotonic one. The figures of accumulative length of induced rules show that for majority of data sets monotonic VC-DomLEM induced shorter rules. Only for ERA data set, shorter rules were obtained by non-monotonic VC-DomLEM.

Finally, we compared mean execution times of both versions of VC-DomLEM over all runs on the twelve data sets. Induction of rules with monotonic VC-DomLEM was on average 3.3 times faster than induction of rules with non-monotonic VC-DomLEM. Thus, the results showed that monotonic VC-DomLEM is more efficient than non-monotonic VC-DomLEM.

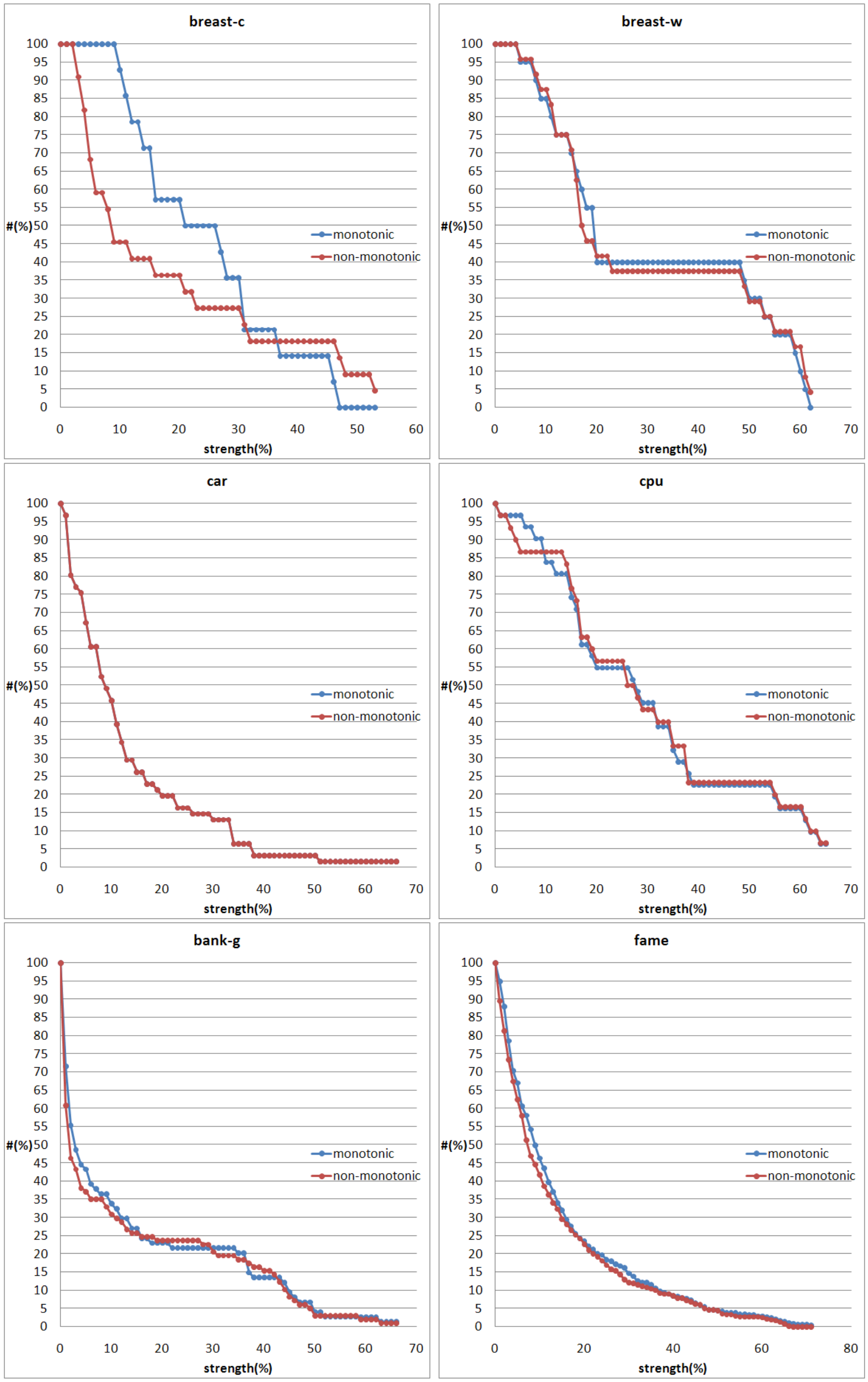

Fig. 2. Accumulative strength characteristics of decision rules induced by monotonic and non-monotonic VC-DomLEM.

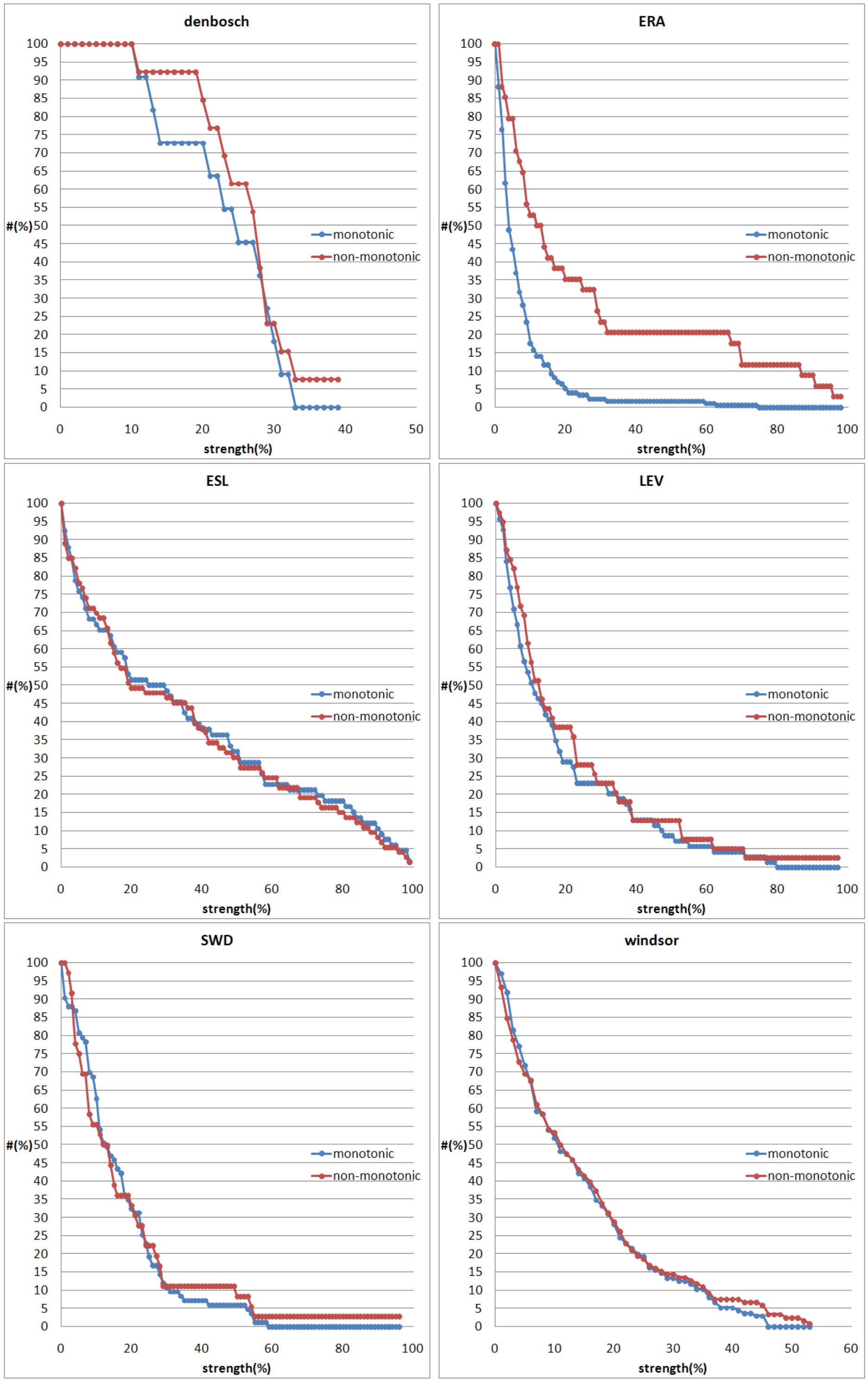

Fig. 3. Accumulative strength characteristics of decision rules induced by monotonic and non-monotonic VC-DomLEM.

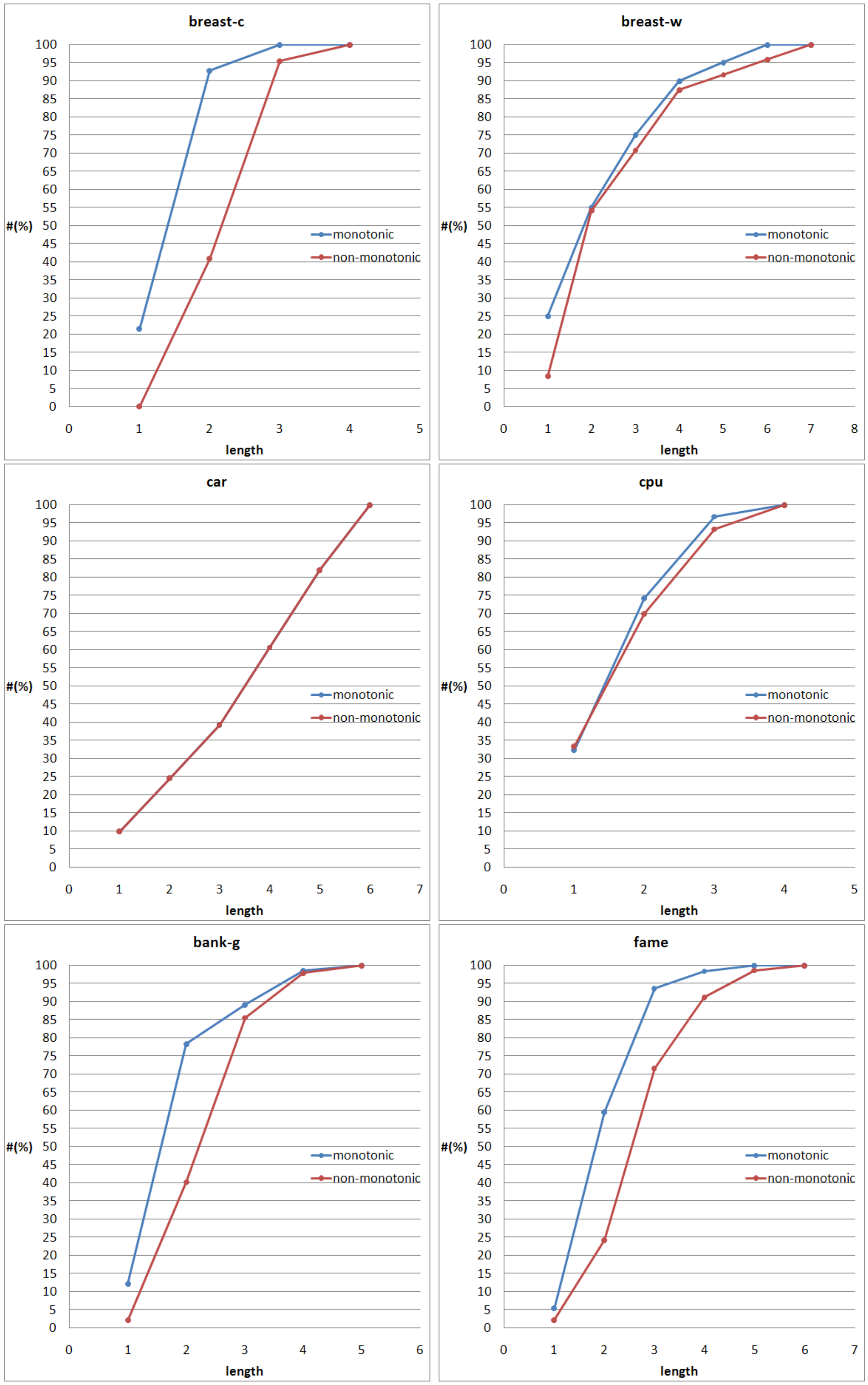

Fig. 4. Accumulative length characteristics of decision rules induced by monotonic and non-monotonic VC-DomLEM.

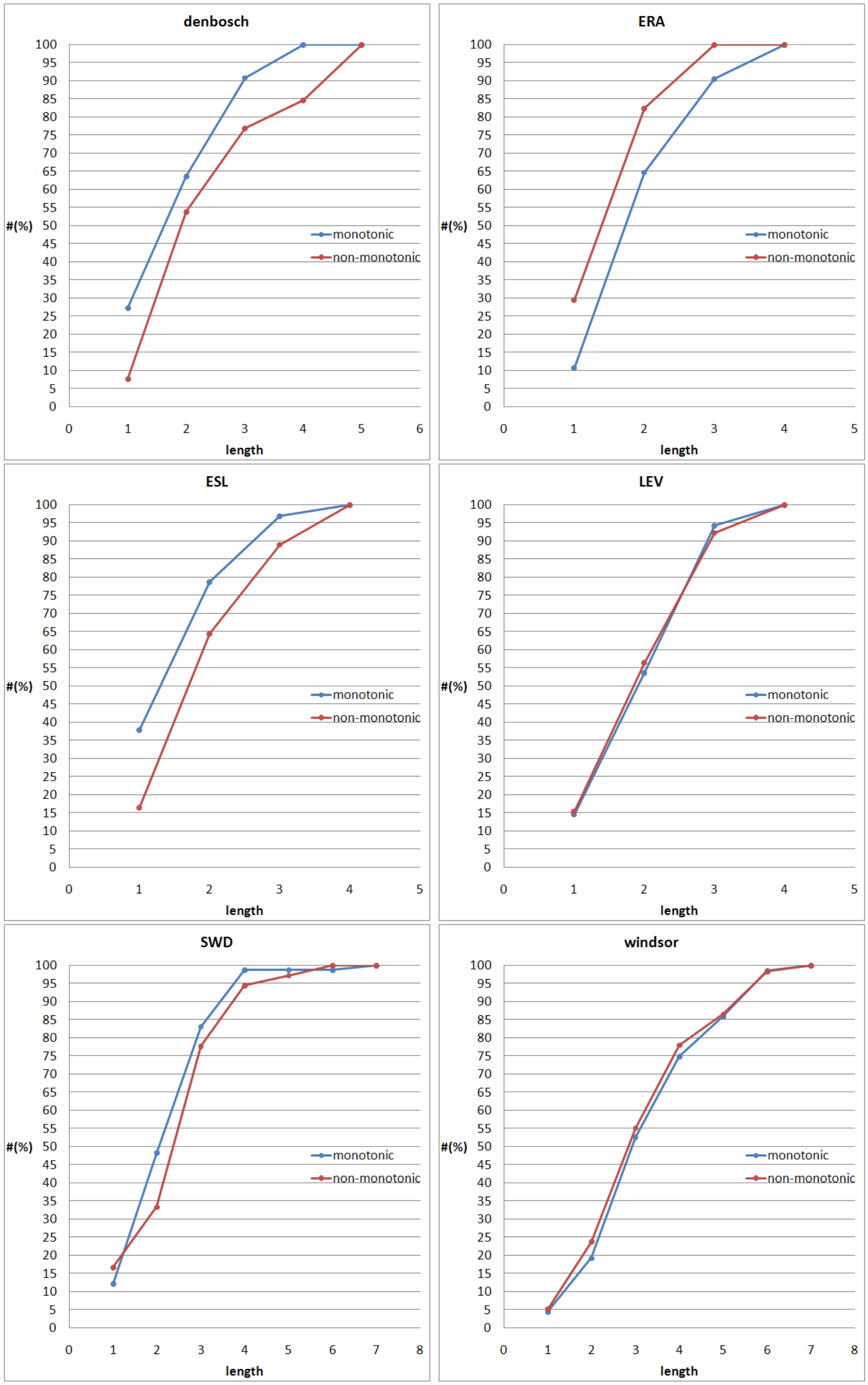

Fig. 5. Accumulative length characteristics of decision rules induced by monotonic and non-monotonic VC-DomLEM.

8 Conclusions

In this paper, we have presented a rule induction algorithm based on sequential covering, called VC-DomLEM. This algorithm can be used for both ordered and non-ordered data. It generates a minimal set of non-redundant decision rules. We have proposed three rule consistency measures which can be applied during rule induction: ϵ-consistency, ϵ'-consistency and μ-consistency. In Theorems 1 and 3, we have proved that the presented algorithm is correct, i.e., it can always induce rules that are consistent to a required degree. Moreover, we have analyzed properties of induced rules, and we have shown how to improve rule induction efficiency due to application of monotonic rule consistency measures: ϵ-consistency or ϵ'-consistency (Theorem 2).

Computational experiment presented in Section 7, concerning twelve ordinal classification data sets, showed good performance of VC-DomLEM. In particular, monotonic VC-DomLEM (i.e., VC-DomLEM using monotonic ϵ-consistency or ϵ'-consistency measure) produced the best results with respect to mean absolute error and percentage of correctly classified objects. We have verified that, in general, decision rules produced by monotonic VC-DomLEM are shorter than rules induced by non-monotonic VC-DomLEM (i.e., VC-DomLEM using non-monotonic μ-consistency measure). We have also observed that induction of rules with monotonic VC-DomLEM is significantly faster than with non-monotonic VC-DomLEM. This observation is concordant with our remarks expressed in Section 5.

References

1. A. Ben-David, L. Sterling, Y.-H. Pao, Learning and classification of monotonic ordinal concepts, Computational Intelligence 5(1) (1989) 45–49.
2. A. Ben-David, Monotonicity maintenance in information-theoretic machine learning algorithms, Machine Learning 19(1) (1995) 29–43.
3. A. Ben-David, D. L. Jagerman, Evaluation of the number of consistent multiattribute classification rules, Engineering Applications of Artificial Intelligence 10(2) (1997) 205–211.
4. A. Ben-David, L. Sterling, T. Tran, Adding monotonicity to learning algorithms may impair their accuracy, Expert Systems with Applications 36(3) (2009) 6627–6634.
5. J. Błaszczyński, S. Greco, R. Słowiński, Multi-criteria classification – A new scheme for application of dominance-based decision rules, European Journal of Operational Research 181(3) (2007) 1030–1044.
6. J. Błaszczyński, R. Słowiński, Incremental Induction of Decision Rules from Dominance-based Rough Approximations, Electronic Notes in Theoretical Computer Science 82(4) (2003) 40–51.
7. J. Błaszczyński, S. Greco, R. Słowiński, M. Szeląg, On Variable Consistency Dominance-Based Rough Set Approaches, In: S. Greco, Y. Hata, S. Hirano, M. Inuiguchi, S. Miyamoto, H. S. Nguyen, R. Słowiński (Eds.), Rough Sets and Current Trends in Computing, LNAI 4259, Springer-Verlag, Berlin, 2006, pp. 191–202.

8. J. Błaszczyński, S. Greco, R. Słowiński, M. Szeląg, Monotonic Variable Consistency Rough Set Approaches, In: J. Yao, P. Lingras, W. Wu, M. Szczuka, N. J. Cercone, D. Ślęzak (Eds.), Rough Sets and Knowledge Technology, LNAI 4481, Springer-Verlag, Berlin, 2007, pp. 126–133.
9. J. Błaszczyński, S. Greco, R. Słowiński, M. Szeląg, Monotonic Variable Consistency Rough Set Approaches, International Journal of Approximate Reasoning 50(7) (2009) 979–999.
10. K. Cao-Van, Supervised ranking – from semantics to algorithms, Ph.D. thesis, Ghent University, CS Department, 2003.
11. P. Clark, T. Niblett, The cn2 induction algorithm, Machine Learning 3(4) (1989) 261–283.
12. P. Clark, R. Boswell, Rule induction with cn2: some recent improvements, In: Y. Kodratoff (Ed.), Proceedings of the Fifth European Working Session on Learning, Springer-Verlag, Berlin, 1991, pp. 151–163.
13. W. W. Cohen, Fast effective rule induction, In: A. Priedities, S. Russell (Eds.), Proceedings of the Twelfth International Conference on Machine Learning, Morgan Kaufmann, CA, 1995, pp. 115–123.
14. W. W. Cohen, Y. Singer, A simple, fast, and effective rule learner, In: J. Hendler, D. Subramanian, R. Uthurusamy, B. Hayes-Roth (Eds.), Proceedings of the Sixteenth National Conference on Artificial Intelligence, AAAI Press, Menlo Park, CA, 1999, pp. 335–342.
15. K. Dembczyński, R. Pindur, R. Susmaga, Generation of Exhaustive Set of Rules within Dominance-based Rough Set Approach, Electronic Notes in Theoretical Computer Science 82(4) (2003) 96–107.
16. J. Demsar, Statistical Comparisons of Classifiers over Multiple Data Sets, Journal of Machine Learning Research 7 (2006) 1–30.
17. P. Fortemps, S. Greco, R. Słowiński, Multicriteria decision support using rules that represent rough-graded preference relations, European Journal of Operational Research 188(1) (2008) 206–223.
18. J. H. Friedman, Recent advances in predictive (machine) learning, Journal of Classification 23(2) (2006) 175–197.
19. J. H. Friedman, B. E. Popescu, Predictive learning via rule ensembles, Annals of applied statistics 2(3) (2008) 916–954.
20. J. Fürnkranz, Separate-and-conquer rule learning, Artificial Intelligence Review 13 (1999) 3–54.
21. J. Fürnkranz, G. Widmer, Incremental reduced error pruning, In: W.W. Cohen, H. Hirsh (Eds.), Proc. 11th International Conference on Machine Learning, Morgan Kaufmann, CA, 1994, pp. 70–77.
22. S. Greco, B. Matarazzo, R. Słowiński, The use of rough sets and fuzzy sets in MCDM, Chapter 14 in: T. Gal, T. Hanne, T. Stewart (Eds.), Advances in Multiple Criteria Decision Making, Kluwer Academic Publishers, Boston, 1999, pp. 14.1–14.59.
23. S. Greco, B. Matarazzo, R. Słowiński, Rough sets theory for multicriteria decision analysis, European Journal of Operational Research 129(1) (2001) 1–47.
24. S. Greco, B. Matarazzo, R. Słowiński, J. Stefanowski, Variable Consistency Model of Dominance-based Rough Sets Approach, In: W. Ziarko, Y. Yao (Eds.), Rough Sets and Current Trends in Computing, LNAI 2005, Springer-Verlag, Berlin, 2001, pp. 170–181.
25. S. Greco, B. Matarazzo, R. Słowiński, J. Stefanowski, An algorithm for induction of decision rules consistent with dominance principle, In: W. Ziarko, Y. Yao

(Eds.), Rough Sets and Current Trends in Computing, LNAI 2005, Springler-Verlag, Berlin, 2001, pp. 304–313.
26. S. Greco, B. Matarazzo, R. Słowiński, Rough approximation by dominance relations, International Journal of Intelligent Systems 17 (2002) 153–171.
27. S. Greco, B. Matarazzo, R. Słowiński, Rough sets methodology for sorting problems in presence of multiple attributes and criteria, European Journal of Operational Research 138 (2002) 247–259.
28. S. Greco, B. Matarazzo, R. Słowiński, Rough Membership and Bayesian Confirmation Measures for Parameterized Rough Sets, In: D. Ślęzak, G. Wang, M. S. Szczuka, I. Düntsch, Y. Yao (Eds.), Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, LNAI 3641, Springer-Verlag, Berlin, 2005, pp. 314–324.
29. S. Greco, R. Słowiński, Y.Y. Yao, Bayesian Decision Theory for Dominance-Based Rough Set Approach, In: J. Yao, P. Lingras, W. Wu, M. Szczuka, N. J. Cercone, D. Ślęzak (Eds.), Rough Sets and Knowledge Technology, LNAI 4481, Springer-Verlag, Berlin, 2007, pp. 134–141.
30. S. Greco, B. Matarazzo, R. Słowiński, Parameterized rough set model using rough membership and Bayesian confirmation measures, International Journal of Approximate Reasoning 49(2) (2008) 285–300.
31. S. Greco, B. Matarazzo, R. Słowiński, Dominance-based rough set approach to interactive multiobjective optimization, In: J. Branke, K. Deb, K. Miettinen, R. Słowiński (Eds.), Multiobjective Optimization, LNCS 5252, Springer, Berlin, 2008, pp. 121–155.
32. J. W. Grzymała-Busse, LERS – A system for learning from examples based on rough sets, In: R. Słowiński (Ed.), Intelligent Decision Support. Handbook of Applications and Advances of the Rough Sets Theory, Kluwer Academic Publishers, Dordrecht, 1992, pp. 3–18.
33. J. W. Grzymała-Busse, A new version of the rule induction system LERS, Fundam. Inform. 31(1) (1997) 27–39.
34. J. W. Grzymała-Busse, J. Stefanowski, Three discretization methods for rule induction, Int. J. Intell. Syst. 16(1) (2001) 29–38.
35. J. Han, M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, San Francisco, CA, 2006.
36. W. Iba, J. Wogulis, P. Langley, Trading off simplicity and coverage in incremental concept learning, In: John E. Laird (Ed.), Proceedings of the 5th International Conference on Machine Learning, Morgan Kaufmann, CA, 1988, pp. 73–79.
37. I. Kononenko, M. Kukar, Machine Learning and Data Mining, Horwood Publishing, Coll House, Westergate, Chichester, West Sussex, 2007.
38. R. S. Michalski, On the quasi-minimal solution of the general covering problem, In: Proceedings of the 5th International Symposium on Information Processing, FCIP-69, vol. A3 (Switching Circuits), Bled, 1969, pp. 125–128.
39. R. S. Michalski, A theory and methodology of inductive learning, In: B. G. Buchanan, D. C. Wilkins (Eds.), Readings in knowledge acquisition and learning: automating the construction and improvement of expert systems, Morgan Kaufmann, San Francisco, CA, 1993, pp. 323–348.
40. R. S. Michalski, K. A. Kaufman, Data mining and knowledge discovery: A review of issues and a multistrategy approach, In: R. S. Michalski, I. Bratko, A. Bratko (Eds.), Machine Learning and Data Mining: Methods and Applications, John Wiley & Sons, New York, NY, 1998, pp. 71–112.
41. Z. Pawlak, Rough sets, International Journal of Information & Computer Sciences 11 (1982) 341–356.

42. Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Norwell, MA, USA, 1991.
43. Z. Pawlak, Rough Sets and Intelligent Data Analysis, Information Sciences 147 (1–4) (2002) 1–12.
44. Z. Pawlak, A. Skowron, Rough membership functions, In: R. R. Yaeger, J. Kacprzyk, M. Fedrizzi (Eds.), Advances in the Dempster Shafer Theory of Evidence, John Wiley & Sons, New York, NY, 1994, pp. 251–271.
45. Z. Pawlak, A. Skowron, Rudiments of rough sets, Information Sciences 177(1) (2007) 3–27.
46. G. Pagallo, D. Haussler, *Boolean feature discovery in empirical learning. Machine Learning*, 5, 1990, pp. 71–99.
47. J. Platt, Machines using Sequential Minimal Optimization, In: B. Schoelkopf, C. Burges, A. Smola (Eds.), Advances in Kernel Methods – Support Vector Learning, MIT Press, Cambridge, MA, 1998.
48. J. R. Quinlan, Learning logical definitions from relations, Machine Learning 5(3) (1990) 239–266.
49. J. R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann, San Mateo, CA, 1992.
50. J. R. Quinlan, R. M. Cameron-Jones, Foil: A midterm report, In: P.B. Brazdil (Ed.), Proceedings of the European Conference on Machine Learning, ECML '93, LNCS 667, Springer-Verlag, Berlin, 1993, pp. 3–20.
51. U. Ruckert, S. Kramer, A statistical approach to rule learning, In: W.C. Cohen, A. Moore (Eds.), Proceedings of the 23rd international conference on Machine learning, ICML'06, ACM, New York, NY, 2006, pp. 785–792.
52. R. Słowiński, J. Stefanowski, RoughDAS and RoughClass software implementations of rough sets approach, In: R. Słowiński (Ed.), Intelligent Decision Support. Handbook of Applications and Advances of the Rough Sets Theory, Kluwer Academic Publishers, Dordrecht, 1992, pp. 445–456.
53. R. Słowiński, S. Greco, B. Matarazzo, Mining decision-rule preference model from rough approximation of preference relation, In: Proc. 26^{th} IEEE Annual International Conference on Computer Software & Applications, IEEE Computer Society Press, Los Alamitos, CA, 2002, pp. 1129–1134.
54. R. Słowiński, S. Greco, Inducing Robust Decision Rules from Rough Approximations of a Preference Relation, In: L. Rutkowski, J. Siekmann, R. Tadeusiewicz, L.A. Zadeh (Eds.), Artificial Intelligence and Soft Computing, LNAI 3070, Springer-Verlag, Berlin, 2004, pp. 118–132.
55. R. Słowiński, S. Greco, B. Matarazzo, Rough Set Based Decision Support, Chapter 16 in: E. Burke, G. Kendall (Eds.), Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques, Springer-Verlag, New York, 2005, pp. 475–527.
56. A. Skowron, Boolean Reasoning for Decision Rules Generation, In: H. J. Komorowski, Z. W. Raś (Eds.), Proceedings of Methodologies for Intelligent Systems, 7th International Symposium, ISMIS '93, Springer-Verlag, London, 1993, pp. 295–305.
57. J. Stefanowski, On rough set based approaches to induction of decision rules, In: L. Polkowski, A. Skowron (Eds.), Rough Sets in Data Mining and Knowledge Discovery, 1, Physica-Verlag, Heidelberg, 1998, pp. 500–529.
58. D. Ślęzak, Rough Sets and Bayes Factor, Transactions on Rough Sets III (2005) 202–229.
59. D. Ślęzak, W. Ziarko, The investigation of the Bayesian rough set model, International Journal of Approximate Reasoning 40 (2005) 81–91.

60. S. M. Weiss, N. Indurkhya, Lightweight rule induction, In: P. Langley (Ed.), International Conference on Machine Learning, Morgan Kaufmann, CA, 2000, pp. 1135–1142.
61. Y.Y. Yao, Decision-theoretic rough set models, In: J. Yao, P. Lingras, W. Wu, M. Szczuka, N. J. Cercone, D. Ślęzak (Eds.), Rough Sets and Knowledge Technology, LNAI 4481, Springer-Verlag, Berlin, 2007, pp. 11–12.
62. Y.Y. Yao, Probabilistic rough set approximations, International Journal of Approximate Reasoning 49(2) (2008) 255–271.
63. W. Ziarko, Variable Precision Rough Set Model, Journal of Computer and System Sciences 46(1) (1993) 39–59.
64. W. Ziarko, Stochastic Approach to Rough Set Theory, In: S. Greco, Y. Hata, S. Hirano, M. Inuiguchi, S. Miyamoto, H. S. Nguyen, R. Słowiński (Eds.), Rough Sets and Current Trends in Computing, LNAI 4259, Springer-Verlag, Berlin, 2006, pp. 38–48.