Poznań University of Technology

Institute of Computing Science

Rule Models for Ordinal Classification in Variable Consistency Rough Set Approaches

Jerzy Błaszczyński

A dissertation submitted to
the Council of the Faculty of Computer Science and Management
in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Supervisor
Professor Roman Słowiński, PhD Dr Habil.

Politechnika Poznańska

Instytut Informatyki

Regułowe modele klasyfikacji porządkowej oparte na teorii zbiorów przybliżonych ze zmienną spójnością

Jerzy Błaszczyński

Rozprawa doktorska
Przedłożono Radzie Wydziału Informatyki i Zarządzania
Politechniki Poznańskiej

Promotor
prof. dr hab. inż. Roman Słowiński

Institute of Computing Science
Poznań University of Technology
Piotrowo 2, 60–965 Poznań, Poland
http://www.cs.put.poznan.pl

When one admits that nothing is certain one must, I think, also admit that some things are much more nearly certain than others.

Bertrand Russell

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

1.1 Problem setting

This work concerns *classification* problems, in which the aim is to assign an *object* (called also *example*, *case* or *observation*) to one of a finite number of discrete *classes* (or *categories*). The objects are described by attributes. In machine learning, classification is preceded by learning of a *classifier* on so-called *training* objects. For all training objects the value of *decision* (or *class*) attribute is known a priori. Then, the aim of learning is defined as to construct the classifier that predicts as accurately as possible the value of decision attribute (or class) for another set of so-called *testing* objects.

classification

To be more precise, we consider here *ordinal classification* problems, which further also include *monotonicity constraints*. Ordinal classification problems involve additional *domain knowledge* about attributes. This domain knowledge permits to an *expert* (or *decision maker*) to specify an ordering in the value sets of some attributes. The set of attributes is divided into condition attributes (independent variables) and decision attributes. In ordinal classification problems, the values of decision attributes correspond to ordered decision classes (or categories). Decision attribute is thus expressed on an ordinal scale. There are perhaps two major types of ordered discrete decision attributes (Anderson, 1984). A decision attribute of the first type is directly related to a single, underlying continuous attribute which is discretized. Example of such an attribute may be "income in dollars": $0 - 2000, 2001 - 3000$, and so on (McCullagh, 1980). Such attribute may be interpreted as grouped continuous attribute. A decision attribute of the second type is qualitative and expresses a grade or an ordered value. Here, example may be attribute "pain relief after treatment": *worse, same, slight im-*

ordinal classification

provement, moderate improvement, marked improvement or *complete relief* (Anderson and Philips, 1981). The expert, being a doctor, making the assessment uses several pieces of information making his decision on the observed class or category. In most of practical classification problems the set of decision attributes is a singleton.

monotonicity constraints

Informally, monotonicity constraints, which may be considered in the ordinal classification problem, mean that the predicted ordered values of the decision attribute are monotonically non-decreasing (or non-increasing) with the values of other attributes that are expressed on ordinal or numerical scales (Ben-David et al., 2009). Such a background knowledge is typical in describing various phenomena, e.g., "the larger the mass and the smaller the distance, the larger the gravity", "the more a tomato is red, the more it is ripe" or "the better the school marks of a pupil, the better his overall classification" (Greco et al., 2008b). Thus, monotonicity constraints relate condition and decision attributes having ordinal scales. Monotonicity constraints considered in the ordinal classification make this problem equivalent to *multiple criteria sorting* problem considered in *multiple criteria decision analysis* (MCDA) (Greco et al., 2010). In multiple criteria sorting, attributes with value sets ordered according to decreasing or increasing preference of a decision maker are called *criteria* (Roy, 1996). Nevertheless, attributes may be treated as criteria even though ordering of their value sets may not come from preferences of a decision maker. In problems described by criteria, monotonicity constraints are also called *semantic correlation*. Semantic correlation states that a better evaluation of an object on a *condition criterion*, with other evaluations being fixed, should not worsen its evaluation on *decision criterion*.

For instance, consider bond rating which consists in assigning bonds to ordered categories called grades: "D", "CC", ..., "B", "B+", ..., "A-", "A", "AA", "AAA". In this case, monotonicity constraint or semantic correlation means that a bond that is getting a better evaluation on any of financial indicators, with other values of indicators being fixed, should not get a worse grade. Another example may be pupil's results in mathematics, physics and its general achievement (Słowiński et al., 2002a). We may consider preferences of, e.g., members of school teachers' council in this example. Thus, the preference ordering of the attributes values is obvious: "good" is better than "medium" and "bad", and "medium" is better than "bad". It is known, moreover, that mathematics is semantically correlated with general achievement, as well as physics is semantically correlated with general achievement. In this example, improvement of pupil's score in mathematics or physics, with other values unchanged, should not worsen pupil's general achievement, but rather improve it.

Ordinal classification problems with monotonicity constraints are important, since they are common in everyday life. For instance, selecting the best route to work, where to shop, which product to buy, and where to live, are such examples of daily ordinal decision-making which may involve monotonic constraints (Ben-David et al., 2009). Moreover, even when the ordering seems irrelevant, the presence or absence of a property have an ordinal interpretation, because if two properties are related, the presence rather than the absence should make more (or less) probable the presence of the other property (Greco et al., 2010; Błaszczyński et al., accepted for publication 2010). The same is true when the presence or absence of a property is graded or fuzzy.

While multiple criteria sorting is widely considered in multiple criteria decision analysis, ordinal classification problems with monotonicity constraints are rarely considered in machine learning. This work is aiming to bridge these two approaches.

1.2 Background

In this section, we present the research background of this thesis. We start with comparison of the points of view of machine learning and decision support on the classification problems. Then, we continue with a brief presentation of existing approaches to ordinal classification with monotonicity constraints.

1.2.1 Machine learning and decision aiding perspective on classification

Looking at classification from machine learning perspective (Friedman, 2006), the goal is to predict (estimate) the unknown value of attribute d given a set of measured values of other attributes (characteristics or properties) of an object (observation) y. The attribute d is called the *output* or *response* variable, and y are referred to as the *input* or *predictor* variables. The prediction is a function on y and to achieve the goal one needs to produce a good predictive function. This requires a definition of the quality measure of any predictive function (classifier). The most commonly used measure of the lack of quality is a *loss* that reflects the cost of mistakes (i.e., the loss or cost of predicting a value $\bar{i}$ when the true value of d is i). Different types of loss functions (i.e., zero-one loss function, squared loss function, absolute loss function) allows to direct the search for a good classifier. The problem, while being easy to state is difficult to solve. The simple definition of the problem as an optimization problem with respect to one measure of quality of a classifier (one aspect of quality of the predictive function) is important for

the comparison of the view on classification problem from machine learning perspective with the view on this problem from decision aiding perspective.

The classification problem from the decision aiding point of view is not so easy to state. In decision aiding, classification corresponds to the sorting problematic (Roy, 1996), which can be seen as an activity aiming at revealing the unknown value of decision (i.e., value of the decision attribute) and at recommending, or simply favoring, a decision that will increase the consistency between the decision and description of objects by other attributes. Thus, in decision aiding, it is always about recommendation that needs to be interpretable to make the final decision. This is why the recommendation needs to be consistent and traceable. The recommendations are presented to a *decision maker* who is expecting them to be consistent with his/her preferences or expert knowledge. The final objective of decision aiding is, of course, to help make "better" decisions. However, the meaning of better depends, in part, on the context in which classification is being made. Moreover, in many cases, due to the limitation of the decision model (i.e., classifier) and imprecise, uncertain or ill-defined data, it is impossible to point the best decision objectively (Figueira et al., 2005). The decision maker is assessing the recommended decisions subjectively, on the basis of his/her expertise. Thus, while in machine learning the goal is to accurately estimate the value of the dependent variable, in decision aiding the goal is to find a convincing, consistent and traceable recommendation.

1.2.2 Existing approaches to ordinal classification problem with monotonicity constraints

We overview the existing approaches to ordinal classification with monotonicity constraints. We consider methods that originate from multiple criteria decision analysis (MCDA), machine learning and statistics. We start with UTADIS (UTilités Additives DIScriminantes) and its extensions, which are good representatives of MCDA. Then we present Dominance-based Rough Set Approach (DRSA), that is an important base on which we build upon in this thesis. DRSA can be considered as a framework that joins together MCDA approach and machine learning approaches. We continue the overview with methods that originate from machine learning and statistics: OLM (Ordinal Learning Model), OSDL (Ordinal Stochastic Dominance Learner), monotone decision trees, and other approaches.

1.2.2.1 UTADIS

UTilités Additives DIScriminantes (UTADIS) method is a variant of the well known UTilités Additives (UTA) method (Jacquet-Lagrèze and Siskos, 1982). UTADIS was designed to solve the ordinal classification with monotonicity constraints (Doumpos and Zopounidis, 2004). The objective of UTADIS method is to develop a classification function of an additive value form:

$$V(y) = \sum_{i=1}^{n} w_i v_i(y_i), \tag{1.1}$$

where, w_i is the weight of i-th attribute, y_i is the value of object y on i-th attribute, and $v_i(y_i)$ the marginal value function for i-th attribute.

The classification of objects into k categories introduced by value set of the decision attribute is performed in a straightforward way through the introduction of $k-1$ value cut-off threshold points $t_1, t_2, \ldots, t_{k-1}$, such that object y for which $V(y) \in (t_{k-1}, t_k]$ is classified to class k. The estimation of the additive value function and cut-off thresholds is performed through linear programing techniques. The objective of the problem is to develop the additive value model that can reproduce the classification of objects from the learning data set as accurately as possible. A detailed description of the linear programming formulation used in UTADIS can be found in (Zopounidis and Doumpos, 1999; Doumpos and Zopounidis, 2004).

UTADIS^{GMS} (Greco et al., 2009) is an extension of UTADIS whose characteristic feature is that it takes into account, in (1.1), the set of all value functions compatible with the assignment of training objects into k categories. It considers general non-decreasing marginal value functions instead of piecewise linear only that are typically used in UTADIS. Moreover, since it explores the whole space of compatible value functions it provides robust classification. These improvements involve, however, an increased computation cost. A method that selects the "most representative" value function among the set of compatible ones is presented in (Greco et al., to appear 2010). This function represents all other compatible value functions, which also do contribute to its definition. Furthermore, it highlights the possible assignments of objects to categories that correspond to the most stable part of the robust classification obtained by UTADIS^{GMS}.

1.2.2.2 Dominance-based Rough Set Approach

Dominance-based rough set approach (DRSA) (Greco et al., 1995, 1999b; Słowiński et al., 2005, 2009) is defined as an extension of the classical rough set approach (Pawlak, 1982),

which is also called indiscernibility-based rough set approach (IRSA). More detailed information about IRSA can be found in chapter 2. DRSA uses the *dominance relation* where IRSA uses the indiscernibility relation. Application of the dominance relation permits to take into account evaluations of objects by *criteria* (i.e., attributes with preference-ordered domains (scales)). Suppose, for simplicity, that set D of decision attributes is a singleton, $D = \{d\}$. Decision attribute d makes a partition of the universe of object U into finite number of classes $X_i, i = 1, \ldots, n$. Each $y \in U$ belongs to one and only one class X_i. The upward and downward unions of classes boil down, respectively, to:

$$X_i^{\geq} = \bigcup_{j \geq i} X_j, \quad X_i^{\leq} = \bigcup_{j \leq i} X_j, \quad i = 1, \ldots, n. \tag{1.2}$$

DRSA uses the dominance relation in order to enable granular computing with dominance cones (for more details, see chapter 3; particularly 3.2, and (Greco et al., 1998a, 1999b, 2001a, 2002b; Słowiński et al., 2000)).

Given a set of criteria $P \subseteq C$, the inclusion of an object $y \in U$ to the upward union of classes $X_i^{\geq}$ may be inconsistent with respect to the *dominance principle*. The dominance principle says that if evaluations of object $y_1 \in U$ on all considered criteria are not worse than evaluations of object $y_2 \in U$, then y_1 should be assigned to a class not worse than y_2. The dominance principle and more formally, the definition of the dominance relation as well as the definition of granules of knowledge that it introduces are discussed in section 3.2. If, given a set of criteria P, the inclusion of $y \in U$ to $X_i^{\geq}$, $i = 2, \ldots, n$, creates inconsistency in the sense of the dominance principle, we say that y belongs to $X_i^{\geq}$ with some ambiguity. Thus, y belongs to $X_i^{\geq}$ without any ambiguity with respect to $P \subseteq C$, if $y \in X_i^{\geq}$ and there is no violation of the dominance principle. This means that all objects P-dominating y belong to $X_i^{\geq}$, which can be also denoted as the P-dominating cone based on y being composed only of objects that belong to $X_i^{\geq}$, i.e., $D_P^+(y) \in X_i^{\geq}$. Furthermore, y possibly belongs to $X_i^{\geq}$ with respect to $P \subseteq C$, if y belongs to $X_i^{\geq}$ with or without ambiguity. Thus, y possibly belongs to $X_i^{\geq}$, with respect to $P \subseteq C$, if among the objects P-dominated by y there is an object x belonging to $X_i^{\geq}$, which can be also denoted as the P-dominated cone based on y being composed of at least one object that belongs to $X_i^{\geq}$, i.e., $D_P^-(y) \cap X_i^{\geq} \neq \emptyset$.

Analogous kind of reasoning may be made for object y and union $X_i^{\leq}$.

For $P \subseteq C$, the set of all objects belonging to $X_i^{\geq}, i = 2, \ldots, n$, without any ambiguity constitutes the P-lower approximation of $X_i^{\geq}$, denoted by $\underline{P}(X_i^{\geq})$, and the set of all objects that possibly belong to $X_i^{\geq}, i = 2, \ldots, n$ constitutes the P-upper approximation

of $X_i^{\geq}$, denoted by $\overline{P}(X_i^{\geq})$:

$$\underline{P}(X_i^{\geq}) = \Big\{y \in U : D_P^+(y) \subseteq X_i^{\geq}\Big\}, \quad \overline{P}(X_i^{\geq}) = \Big\{y \in U : D_P^-(y) \cap X_i^{\geq} \neq \emptyset\Big\}. \tag{1.3}$$

Analogously, we define P-lower approximation and P-upper approximation of $X_i^{\leq}$, $i = 1, \ldots, n-1$, as follows:

$$\underline{P}(X_i^{\leq}) = \Big\{y \in U : D_P^-(y) \subseteq X_i^{\leq}\Big\}, \quad \overline{P}(X_i^{\leq}) = \Big\{y \in U : D_P^+(y) \cap X_i^{\leq} \neq \emptyset\Big\}. \tag{1.4}$$

All the objects belonging to $X_i^{\geq}$ and $X_i^{\leq}$ with some ambiguity constitute the P-boundary of $X_i^{\geq}$ and $X_i^{\leq}$. P-boundaries are denoted by $Bn_P(X_i^{\geq})$ and $Bn_P(X_i^{\leq})$, respectively. They can be represented in terms of P-lower approximations and P-upper approximations as follows:

$$Bn_P(X_i^{\geq}) = \overline{P}(X_i^{\geq}) - \underline{P}(X_i^{\geq}), \quad Bn_P(X_i^{\leq}) = \overline{P}(X_i^{\leq}) - \underline{P}(X_i^{\leq}) \tag{1.5}$$

In DRSA, rules are induced from three types of approximated sets: P-lower approximations (certain rules), P-upper approximations (possible rules) and P-boundaries (approximate rules). Induction of decision rules is a complex problem and many algorithms have been introduced to solve it. Some examples of rule induction algorithms that were presented in the context of the rough set analysis are given in (Grzymała-Busse, 1992; Skowron, 1993; Grzymała-Busse, 1997; Grzymała-Busse and Zou, 1998; Bazan, 1998; Krawiec et al., 1998; Stefanowski, 1998; Susmaga et al., 2000). Algorithms proposed for DRSA are the following: (Greco et al., 2000a; Susmaga et al., 2000; Błaszczyński and Słowiński, 2003). 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, generation of a satisfactory set of decision rules. Algorithms from the first category focus on describing objects from approximations by minimal number of minimal rules that are necessary to cover all the objects from the decision table. Algorithms from the second category generate all possible minimal decision rules. The third category includes algorithms that generate all possible minimal rules that satisfy some a priori defined requirements (e.g. maximal rule length). It is known that algorithms from the first category generate sets of rules that perform best in classification. Sets of rules generated by algorithms from the second and the third category are useful for description and discovery purposes. More about rule induction algorithms applicable in DRSA classification can be found in chapter 4.

Once rules are induced they can be used for classification of objects. Rules classification methods for DRSA are described in chapter 5.

1.2.2.3 Ordinal learning model

Ordinal learning model (OLM) (Ben-David et al., 1989) is a simple algorithm that learns monotonic ordinal relations in data by eliminating non-monotonic pairwise inconsistencies. The algorithm is instance-based. This means that it stores the given learning objects into memory in some kind of format, and it is able to deduce from them the class labels of unseen objects by some usually local extrapolation technique. The learning objects are stored as rules (i.e., objects are transformed to rules) in a rule set. Initially, the rule set is empty. Then, during learning, each object is checked against each of the rules in the rule set. If the object is inconsistent with a rule in the rule set, the object or the rule is selected randomly while the other is discarded. If the object is selected it must be checked again against each of the rules in the rule set. If it passes this consistency test, it is added to the rule set as a rule. Thus, the rule set is always consistent.

Classification is made in a similar manner to learning, by checking the classified object against rules from the rule set. The rules are checked in decreasing order of decision attribute values (i.e., classes). The object is assigned to the class indicated by the first rule that cover it. This is equivalent to assigning maximal class i suggested by rules that cover object. If there is no rule in the rule set that covers object, two approaches are possible. In the first and simpler approach, the object is assigned to the worst class. In the second approach, the class is assigned by a nearest neighbor search of rules closest to the object according to the Euclidean distance.

OLM proved to produce very small rule set while learning form inconsistent data (Ben-David and Jagerman, 1997). The main weakness of this model lies in the fact that it does not make any accuracy checking during learning. The rule set depends on the order in which objects are checked against it. Moreover, classification by the nearest rules made for objects not covered by rules from rule set may lead to non-monotone classifications.

1.2.2.4 Ordinal Stochastic Dominance Learner

Ordinal Stochastic Dominance Learner (OSDL) (Cao-Van, 2003) provides an alternative to OLM since it is also an instance-based method. It uses the concept of ordinal stochastic dominance (OSD) to solve ordinal classification with monotonicity constraints. More specifically, the concept of dominance that is used in OSDL is first order stochastic dominance (FOSD). Definition of FOSD is as follows: object y is first-order stochastic dominating object x if for any class i, y has a higher probability of belonging to class i or better than x (which can be denoted $x \leq_{FOSD} y$). For more information on stochastic

dominance definitions please see (Altendorf et al., 2005). The goal of OSDL learning process is to find a mapping function F from the attribute space to class space such that:

$$\forall x \leq y \Rightarrow F(x) \leq_{FOSD} F(y).$$

OSDL constructs two mapping functions F_m and F_M: one that is based on the objects from the best class among those that are stochastically dominated by a given object x, and the second that is based on the objects from the worst class among those that stochastically dominate x. More precisely, for a given class i

$$F_m(x,i) = \min_{y \in (x]} \hat{F}_y(i), \qquad F_M(x,i) = \max_{y \in [x)} \hat{F}_y(i),$$

where $(x]$ is the set of objects stochastically dominated by x and $[x)$ is the set of objects stochastically dominating x and $\hat{F}_y(i)$ is an estimate of probability that y belongs to class at most i. Moreover, if $(x] = \emptyset$, then $F_m(x,i) = 1$ and if $[x) = \emptyset$, then $F_M(x,i) = 0$.

During classification, an interpolation between class assignments by mapping functions F_m and F_M is returned as the result. This interpolation involves a scaling parameter $s \in [0,1]$:

$$\tilde{F}(x,i) = (1-s)F_m(x,i) + sF_M(x,i).$$

Such interpolation has a drawback: to maintain the monotonicity of classification it is required to use the same fixed value of s for all classified objects.

In case when the data is consistent, then $F_m(x,i) \geq F_M(x,i)$ for each x. In order to treat inconsistent data and to overcome problem with scaling parameter s a balanced version of OSDL is proposed (Cao-Van, 2003). This version involves the following interpolation between the mapping functions:

$$\tilde{F}(x,i) = \begin{cases} (1-s)F_m(x,i) + sF_M(x,i) & \text{if } F_m(x,i) \geq F_M(x,i), \\ \frac{(1-s')N_m(x,i)F_m(x,i) + s'N_M(x,i)F_M(x,i)}{N_m(x,i)+N_M(x,i)} & \text{otherwise,} \end{cases}$$

where $s, s' \in [0,1]$, $N_m(x,i)$ is the number of objects from $(x]$ that belong to class better than i and $N_M(x,i)$ is the number of objects from $[x)$ that belong to class not worse than i. Thus, the balanced version of OSDL introduces weighting by the number of objects $N_m(x,i)$ and $N_M(x,i)$ that is made for inconsistent objects. It is meant to reduce the influence of inconsistent objects on classification. We should also remark that this approach is similar to Variable Consistency DRSA (see chapter 3).

1.2.2.5 Monotone decision trees

Decision tree models are one of the most popular in machine learning (Quinlan, 1992; Breiman et al., 1984). Decision tree models that are solving ordinal classification with

monotonicity constraints include: Positive Decision Tree (P-DT), Monotone Decision Trees (MDT), Variable Consistency Monotonic Decision Trees, and Rank Tree (RT).

Positive Decision Tree (P-DT) (Makino et al., 1996) is designed to solve two class problems only. It builds a binary tree, meaning that each of nodes of the tree splits in two sub nodes only. A slightly modified version of binary Shannon entropy measure to select the best split in the nodes. This measure does not however guarantee that the constructed tree classifies objects preserving monotonicity constraints. To achieve this property, the tree needs to be constructed on learning data sets that are gradually updated to be consistent by addition of new artificial objects (see (Cao-Van, 2003) for details).

Monotone Decision Trees (MDT) (Potharst et al., 1988; Potharst and Bioch, 2000; Potharst and Feelders, 2002), can be considered as a non-trivial extension of P-DT for multiple-class ordinal classification with monotonicity constraints. MDT uses the well known impurity measures such as Gini index and entropy to select the best splits in nodes. It also uses a procedure for adding new objects to keep the learning data set consistent. This procedure is called *cornering technique*. It consists in adding artificial objects to the set of objects covered by each node, one in the lower left corner of the node, and another in the upper right corner (notice that each node in the tree represents a subset of the learning data set and has the form of hyperrectangle). The lower left object obtains the highest possible class label which does not introduce additional inconsistency to the data set, while the upper left object – lowest possible label, respectively.

Both P-DT and MDT can be applied on consistent data set only. However, model that extends MDT for inconsistent data sets have been also proposed (Popova, 2004).

Variable Consistency Monotonic Decision Trees (Giove et al., 2002) are models that allow to construct decision trees in Variable Consistency DRSA (VC-DRSA, see chapter 3). Three variable consistency monotonic decision tree models were proposed:

1) *single class decision tree* that discriminates a union of decision classes only,

2) *progressively ordered decision tree* that is constructed again for a single union of decision classes, however, the union for which nodes are created is progressively changing as the tree is growing,

3) *full range tree* that is constructed for all unions of decision classes simultaneously.

Variable Consistency Monotonic Decision Trees use a measure similar to *rough membership* 3.1 to select splits in nodes. Since they are defined within VC-DRSA, they can handle inconsistent data sets.

Another model that constructs trees for ordinal classification with monotonic constraints problem is Rank Tree (RT) (Cao-Van, 2003). It is using an impurity measure based on the ranking error (number of reversed ranks) and in using a specific procedure for maintaining the monotonicity of the tree. It can handle inconsistent data sets.

1.2.2.6 Other approaches

There exist other approaches to the ordinal classification problem (with monotonicity constraints) that mainly originate from statistical learning. We do not present them here with care for details because they are harder to compare with the approach presented in this thesis. These approaches include: isotonic regression (Brunk, 1955; Ayer et al., 1955), isotonic separation (Ayer et al., 1955; Burdakov et al., 2006), monotone support vector machines (Chu and Keerthi, 2005; Le et al., 2006), monotone neural networks (Sill and Abu-Mostafa, 1997; Sill, 1998) and monotone ensembles of classifiers (Kotłowski and Słowiński, 2008, 2009).

1.3 Goal and scope of the thesis

The overall goal of this thesis is improvement of predictive abilities of rule classifiers used in ordinal classification with monotonicity constraints. Three objectives which are given below are associated with this goal.

1) Definition of probabilistic lower approximations of sets of objects characterized by consistency measures which have required properties.

2) Definition of decision rules and algorithms of their induction from probabilistic lower approximations of sets.

3) Application of decision rules in classifiers that aggregate suggestions of object assignments given by matching rules.

These objectives induce the structure of the chapters in the thesis. The objective 1) is achieved in chapters 2 and 3. We define two related probabilistic extensions of rough set approaches: Variable Consistency Indiscernibility-based Rough Set Approaches (VC-IRSA) and Variable Consistency Dominance-based Rough Set Approaches (VC-DRSA).

We chose to present these two approaches, the one that involves granules of knowledge defined by indiscernibility and the one that involves dominance relation because we want to show a general probabilistic extension of the rough set approaches. According to our best knowledge, no such a general extension was proposed so far. Such a choice seems also natural since the classical rough set approach was proposed for indiscernibility relation (Pawlak, 1982). Then, classical rough set approach was extended to Dominance-based Rough Set Approach (Greco et al., 1999a). Moreover, this structure of chapters simplifies the way of introducing definitions: first of indiscernibility based rough sets and then of dominance-based rough sets. Nevertheless, in the further chapters we focus on VC-DRSA since VC-IRSA are not directly applicable to the subject of the thesis.

The probabilistic rough set approaches allow to extend lower approximation of a set by objects with sufficient evidence for membership to the set. To quantify this evidence, we propose different measures of the overlap between a granule of knowledge based on a considered object and the approximated set or its complement. We call such measures *consistency measures*. The advantage of proposed definition of probabilistic lower approximation with consistency measure is that approaches proposed by other researchers can be represented in this definition with specific consistency measures. This allows us to compare other approaches to our proposals with respect to properties of the respective consistency measures.

The consistency measures are meant to be easy in interpretation so that one can directly specify properties of objects included in the probabilistic lower approximation. Different consistency measures are used to express different view on the consistency of objects. In this way, the lower approximation characterized by acceptable level of the selected consistency measure can be used to distinguish objects which are considered acceptably consistent from the selected point of view. Inspired by some basic properties of rough sets, we find it reasonable to require from consistency measures several properties of monotonicity that correspond directly to monotonicity properties of the lower approximation. These monotonicity properties guarantee that any object from a monotonic lower approximation will belong to this lower approximation after the data set is extended with respect to the set of attributes, set of objects or union of ordered classes. The monotonicity properties guarantee the same behavior of objects from lower approximation when improvement of evaluation of any object in the data set takes place. We show monotonicity properties of some of the compared consistency measures. These properties prove to be important in further stages of construction of the decision rules classifiers.

The objective 2) is achieved in chapter 4. Having defined the probabilistic lower approximations specified by required properties, we consider decision rule models which are induced on the basis of these lower approximations. The objective of rule induction is thus to construct a set of decision rules that expresses dependencies observable in the acceptably consistent part of the data set. Moreover, because we want the set of rules to be traceable, each of rules is characterized by consistency measure that corresponds to consistency measure used to define the probabilistic lower approximation. The induced rules must satisfy consistency requirements that are transformable to those that are imposed on objects included in the lower approximation. We show how to induce decision rules on the basis of lower approximations that have monotonicity properties, and on the basis lower approximations that don't have these properties. We prove that it is achievable to cover all objects form the probabilistic lower approximations in both cases. We also show that it is possible to induce rules more effectively when it is known that the consistency measure is monotonic.

Finally, the objective 3) is meet in chapter 5. We propose two types of classifiers that employ set of decision rules defined in chapter 4. The first type includes single classifiers whose results are easily interpretable while they may be not sufficiently accurate in some of applications. We apply the standard classification scheme defined for DRSA. We also propose a new classification scheme that is able to deal with imprecise and contradictory suggestions given by the matching rules. The second type of classifiers is an ensemble of classifiers that employ consistency measures and diversification between rule component classifiers to provide more accurate classification. Due to their complexity, these classifiers are, however, not as straightforward in interpretation as single classifiers.

We experimentally prove properties of the proposed decision rules classifiers in chapter 6. We compare our classifiers learned on objects belonging to probabilistic lower approximations characterized by consistency measures with well known methods proposed by other researchers.

A more detailed summary of results, conclusions, and plans for further research can be found in chapter 7.

CHAPTER 2

Variable Consistency Indescerinbility-based Rough Set Approaches (VC-IRSA)

2.1 Problem statement and basic definitions

Rough set theory in its classical definition (Pawlak, 1982, 1991), introduces a distinction of objects in the considered universe U into two categories of consistent and inconsistent objects. The objects in U are assigned to decision classes X_i, $(i = 1, 2, \ldots, n)$, according to the value of decision attributes from set D. We assume here, without loss of generality, that set of decision attributes D is a singleton $D = \{d\}$. The decision attribute d makes a partition of set U into finite number n of disjoint decision classes. When considering a single class X_i, we will drop index i, for simplicity, and thus the considered set of objects will be denoted by X. For each X_i, a lower approximation and an upper approximation are defined. In the classic definition of rough sets, the lower approximation is composed of consistent objects only. The upper approximation, on the other hand, includes all of objects from the approximated set and objects that are indiscernible with them. In other words, objects which in view of available knowledge, certainly belong to X_i are assigned to lower approximation of X_i and objects which possibly belong to X_i are assigned to upper approximation of X_i.

The inconsistency in the sense of classical rough sets occurs when indiscernible object are assigned by decision attribute d to different classes X. The *indiscernibility relation*, as well as the definition of granules of knowledge that it yields are discussed in section 2.2.

Inclusion of only consistent objects in lower approximation may lead to small lower approximations and large upper approximations. Even if objects that cause inconsistencies are in minority, they can affect lower approximations of all considered decision classes For this reason, various approaches that extend lower approximations were introduced. The extension is made by inclusion in a lower approximation of set X of those objects for which there is enough evidence for their membership in X. In practice, a consistency measure is used to quantify this evidence and a threshold on this measure permits to make decision about the inclusion. The approaches that allow to extend lower approximations include Variable Precision Rough Set (VPRS) model (Ziarko, 1993), Rough Bayesian (RB) model (Ślęzak, 2005; Ślęzak and Ziarko, 2005), Parameterized Rough Sets (Greco et al., 2005b) and Monotonic Variable Consistency Indiscernibility-based Rough Approaches (VC-IRSA) (Błaszczyński et al., 2007b). In this chapter, the Monotonic Variable Consistency Indiscernibility-based Rough Set Approaches are defined and justified. In section 2.2, we define granules of knowledge that are produced by the indiscernibility relation. Then in section 2.3, we focus on the domain knowledge that is taken into account in various definitions of the consistency measures. Further, in section 2.4, we define lower and upper approximations involving the consistency measures. We investigate properties of these approximations in sections 2.5 and 2.6. Particularly, monotonicity properties of rough approximations are of our special concern in section 2.5. The chapter is summarized in section 2.7.

2.2 Granules of knowledge

One of the elementary features of rough sets is reasoning in terms of the granules of data that are indistinguishable or indiscernible considering the knowledge that is available (Yao, 2003). The knowledge considered in this reasoning is represented by the set of attributes that describe objects and by the relation used for comparison of objects. It follows that the selection of different sets of attributes will yield different granulations of the analysed data. Similarly, the value set of considered attributes my affect the granulation, because richer value sets induce, in general, finer granulation. The latter is related to discretization of attributes. In other words, by projecting a data set U (value-attribute system) onto different sets of attributes or the same set of attributes but with more or less finer value sets, we get, in general, alternative sets of equivalence-classes in the data. These different sets will influence the extraction of relationships and regularities. It is common that the resolution of the attributes needs to be accustomed in order to extract meaningful regularities.

In this chapter, we do not consider the influence of discretization (Fayyad and Irani, 1993; Dougherty et al., 1995) on relationships between granules and decision classes. This aspect has been extensively considered in (Chlebus and Nguyen, 1998; Nguyen and Nguyen, 1998; Nguyen, 2006). In some cases it is assumed that the discretized data set is consistent (Nguyen and Nguyen, 1998; Nguyen, 2006). This is however a strong assumption, especially in the context of VC-IRSA. In our case it would suffice to assume that discretization cuts do not change the granules that include objects from different classes. Simply requiring that discretization maintains the quality of approximation is not sufficient in this case. One may easily come with an example of cuts that do not decrease the quality of approximation but changes the proportion of objects from different classes in a granule.

indiscernibility relation

In the rough set approach proposed by Pawlak (Pawlak, 1982), the objects are compared using the *indiscernibility relation*. For this reason, we call this approach Indiscernibility-based Rough Set Approach (IRSA). The indiscernibility relation is assumed to be an equivalence relation (Pawlak, 2004). Let V_{a_i} be the value set of attribute $a_i \in C$ and $f : U \times C \to V_{a_i}$ be a total function, such that $f(x, a_i) \in V_{a_i}$. Indiscernibility relation I_P is defined for a non-empty subset of attributes $P \subseteq C$ as

$$I_P = \{(y, z) \in U \times U : f(y, a_i) = f(z, a_i) \text{ for all } a_i \in P\}.$$

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*.

Example 2.2.1. *Consider the set U described by means of set P of two condition attributes a_1 and a_2, as presented in Figure 2.1. The indiscernibility granules are the following:*

$$G_1 = \{y_1, y_2, y_3\}, \quad G_2 = \{y_4, y_5, y_6\}.$$

Thus, the three objects y_1, y_2, y_3 within the first granule G_1, cannot be distinguished from one another based on the available attributes P, and the three objects y_4, y_5, y_6 within the second granule G_2, cannot be distinguished from one another based on the available attributes P. It is also worth noting that the first granule G_1 is consistent since all the objects that belong to it are from X_1. This is not the case for granule G_2. Object y_4 belongs to X_1 while the other objects y_5, y_6 belong to X_2. Granule G_2 shows the type inconsistency that is handled by rough set theory.

Now, let us consider an extension of the set of attributes P by attribute a_3. This extension may result in situation shown in Figure 2.2. Introduction of a new attribute

object	a_1	a_2	class
y_1	1	1	X_1
y_2	1	1	X_1
y_3	1	1	X_1
y_4	2	2	X_1
y_5	2	2	X_2
y_6	2	2	X_2

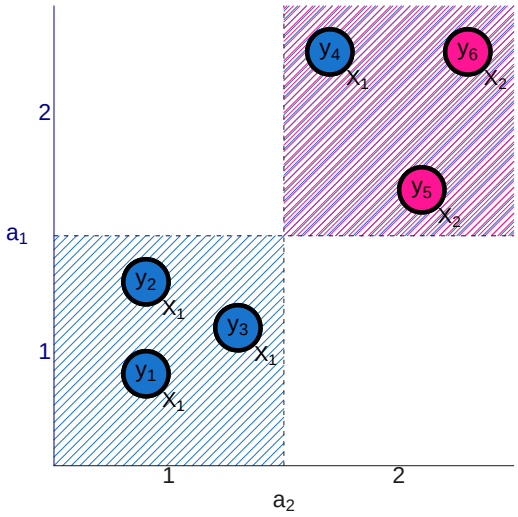


Figure 2.1: Exemplary set of objects described by means of set P of two condition attributes a_1 and a_2. Objects marked with 1 and 2 belong to class X_1 and X_2, respectively.

represents additional knowledge introduced to the analysed data set U. Then we obtain the following more precise structure of granules:

$$G_1 = \{y_1, y_3\}, \quad G_2 = \{y_2\}, \quad G_3 = \{y_4, y_6\}, \quad G_4 = \{y_5\}.$$

object	a_1	a_2	a_3	class
y_1	1	1	1	X_1
y_2	1	1	2	X_1
y_3	1	1	1	X_1
y_4	2	2	2	X_1
y_5	2	2	1	X_2
y_6	2	2	2	X_2

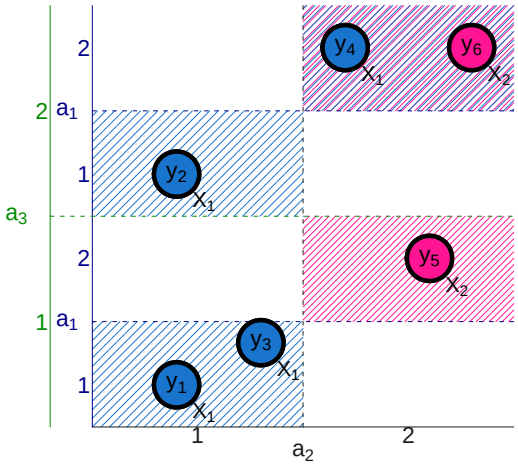


Figure 2.2: Exemplary set of objects described by means of set P' of three condition attributes a_1, a_2 and a_3. Objects marked with 1 and 2 belong to class X_1 and X_2, respectively.

For the extended set of attributes $P' = \{a_1, a_2, a_3\}$ objects y_1 and y_3 as well as y_4 and y_6 remain indiscernible. Objects y_2 and y_5 became incomparable with others and form two new granules. It is also worth noting that, the number of objects in inconsistent granules decreased. One can observe a tendency that with increase of precision, the incomparability of objects in U does not decrease while inconsistency does not increase. We will come back to this observation in sections 2.3 and 2.5.

Let us now consider a further extension of the set of attributes P' by attribute a_4. It may result in situation shown in Figure 2.3. This time, we obtain a fully consistent data set with the following singleton granules:

$$G_1 = \{y_1\}, \quad G_2 = \{y_2\}, \quad G_3 = \{y_3\}, \quad G_4 = \{y_4\}, \quad G_5 = \{y_5\}, \quad G_6 = \{y_6\}.$$

object	a_1	a_2	a_3	a_4	class
y_1	1	1	1	1	X_1
y_2	1	1	2	1	X_1
y_3	1	1	1	1	X_1
y_4	2	2	2	1	X_1
y_5	2	2	1	2	X_2
y_6	2	2	2	2	X_2

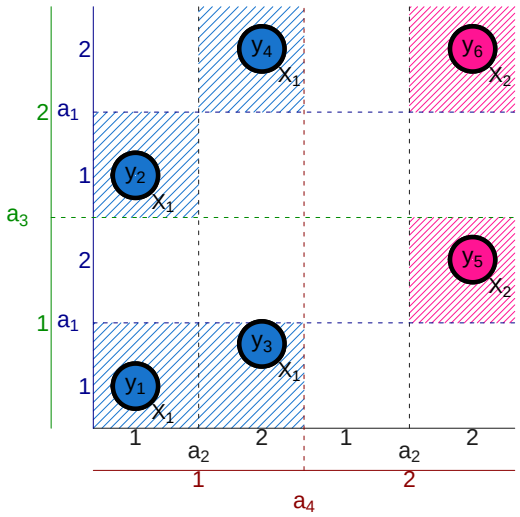


Figure 2.3: Exemplary set of objects described by means of set P'' of four condition attributes a_1, a_2, a_3 and a_4. Objects marked with 1 and 2 belong to class X_1 and X_2, respectively.

In practice, this type of precisiation is usually hard to obtain. That is why measures of consistency in granules are necessary to distinguish meaningful consistency of the objects in the analysed data.

2.3 Consistency principle and consistency measures

One of the ways in which inconsistencies in the data set can be handled is the precisiation by extension of the set of attributes A that describe the objects. Obviously, not always

this precisiation is desirable or achievable. That is why consistency measures are designed to quantify the level of consistency or inconsistency in granules of knowledge. These measures define how the inconsistencies discovered in the analyzed data are taken into account in further reasoning. To be more precise, the inconsistency in a granule is tested with respect to each of the classes that are represented by objects belonging to this granule. Each of the classes needs to be characterized by a threshold of acceptable consistency required for an object to be considered as sufficiently consistent member of the class. The measure of consistency constitutes a type of domain knowledge that is used in variable consistency rough set approaches.

consistency measures

Let us specify conditions that must be satisfied by consistency measures. We distinguish gain-type and cost-type consistency measures. First, let us consider $y_1, y_2 \in U$, $P \subseteq C$, $X \subseteq U$. Given description of y_1 and y_2 by P:

- a gain-type consistency measure $f_X^P(y)$ is any measure satisfying condition: $f_X^P(y_1) \geq f_X^P(y_2) \Leftrightarrow$ it is not less likely that y_1 belongs to X, than that y_2 belongs to X,
- a cost-type consistency measure $g_X^P(y)$ is any measure satisfying condition: $g_X^P(y_1) \leq g_X^P(y_2) \Leftrightarrow$ it is not less likely that y_1 belongs to X, than that y_2 belongs to X.

Second, let us consider $y \in U$, $P \subseteq C$, $X, Y \subseteq U$, where Y has the same interpretation as X (i.e., it denotes a class or a union of classes). Given description of y by P:

- a gain-type consistency measure $f_X^P(y)$ is any measure satisfying condition: $f_X^P(y) \geq f_Y^P(y) \Leftrightarrow$ it is not less likely that y belongs to X, than that it belongs to Y.
- a cost-type consistency measure $g_X^P(y)$ is any measure satisfying condition: $g_X^P(y) \leq g_Y^P(y) \Leftrightarrow$ it is not less likely that y belongs to X, than that it belongs to Y.

A consistency measure expresses the evidence for membership to set X. 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. The distinction between cost or gain type of consistency measure is important when the measure is applied to define rough approximations. Thus, we will consider it more thoroughly in sections 2.4 and 3.4.

μ measure

Rough membership measure was introduced in (Wong and Ziarko, 1987) and its properties were further investigated in (Pawlak and Skowron, 1994; Yao, 2008). It is used to control positive regions in Variable Precision Rough Set (VPRS) model (Ziarko, 1993)

as well as in Bayesian Rough Set Model (Ślęzak and Ziarko, 2005). Rough membership was also considered in the context of attribute reduction in rough sets (Inuiguchi, 2006).

In Indiscernibility-based Rough Set Approach (IRSA), rough membership of $y \in U$ to $X_i \subseteq U$ w.r.t. $P \subseteq C$ is defined as

$$\mu_{X_i}^P(y) = \frac{|I_P(y) \cap X_i|}{|I_P(y)|}, \tag{2.1}$$

where $I_p(y)$ denotes a set of objects indiscernible with object y when considering set of attributes P (i.e., granule of indiscernible objects). Rough membership is a gain-type measure. It captures a ratio of the number of objects that belong to granule $I_P(y)$ and to considered class X_i, to the number of all objects belonging to granule $I_P(y)$. For example, in case of a medical diagnosis, the value of rough membership would express the ratio of the number of patients that have the same medical signs and suffer from the considered disease to the number of all patients that have the same signs. This measure can also be treated as an estimate of conditional probability $Pr(x \in X_i | x \in I_P(y))$.

Other measures than rough membership have been used in variable consistency rough set approaches. For example, Bayesian confirmation measures (Fitelson, 2001; Greco et al., 2004) were considered together with rough membership in Parameterized Rough Sets (PRS) (Greco et al., 2005b). Bayesian confirmation measures quantify the degree to which membership of object y to given granule $I_P(y)$ provides "evidence for or against" or "support for or against" assignment to considered class X_i. They are thus gain-type measures.

confirmation measures

The Bayes factor is an consistency measure that has similar properties to Bayesian confirmation measures (in case of two class problems Bayesian confirmation measure l is a natural logarithm of Bayes factor (Fitelson, 2001)). It is used in Rough Bayesian (RB) model (Ślęzak, 2005; Ślęzak and Ziarko, 2005). Bayes factor for $y \in U$ and $X_i \subseteq U$ w.r.t. $P \subseteq C$ is defined as

Bayes factor

$$B_{X_i}^P(y) = \frac{|I_P(y) \cap X_i||\neg X_i|}{|I_P(y) \cap \neg X_i||X_i|}. \tag{2.2}$$

Bayes factor is a gain-type consistency measure. Coming back to the example with medical diagnosis, the Bayes factor would express, in this case, the ratio of the estimate of probability that a patient has the considered signs on condition that he suffers from the considered disease to the estimate of probability that he has these signs on condition that he does not suffer from this disease. The Bayes factor can also be seen as a ratio of estimates of two conditional probabilities $Pr(x \in I_P(y) | x \in X_i)$ and $Pr(x \in I_P(y) | x \in \neg X_i)$.

ϵ measure

Measure $\epsilon^P_{X_i}(y)$ is a consistency measure which posses some properties of a confirmation measure. This measure has been used in monotonic Variable Consistency Indiscernibility-based Rough Set Approaches (VC-IRSA) (Błaszczyński et al., 2007b). For $P \subseteq C, X_i, \neg X_i \subseteq U$, where $\neg X_i = U - X_i$, $y \in U$, it is defined as

$$\epsilon^P_{X_i}(y) = \frac{|I_P(y) \cap \neg X_i|}{|\neg X_i|}. \tag{2.3}$$

In the numerator of (2.3) there is the number of objects in U that do not belong to class X_i and are indiscernible with object y. In the denominator of (2.3) there is the number of objects in U that do not belong to class X_i. Measure ϵ is an example of cost-type consistency measure and for this reason it is also called a measure of inconsistency. The ratio $\epsilon^P_{X_i}(y)$ is an estimate of conditional probability $Pr(x \in I_P(y) | x \in \neg X_i)$, called also a catch-all likelihood (Fitelson, 2007). This measure is thus an estimate of of probability that object y belongs to granule $I_P(y)$ given that it does not belong to class X_i. It may result in low values of measure $\epsilon^P_{X_i}(y)$ for classes X_i that have low cardinality. In the example of medical diagnosis, the ϵ measure would express the ratio of the number of patients that have the same signs and does not suffer from the considered disease to the number of all known patients that do not have the considered disease. Even though it is easier to give intuition behind this measure by reference to prior probability, knowledge of priors is not necessary to estimate catch-all likelihoods. What is needed to use the ϵ measure is the conditional probability. This argument is further considered by Fitelson (Fitelson, 2007).

$\epsilon^{'}$ measure

Another consistency measure that we consider in VC-IRSA is a cost-type measure $\epsilon^{'P}_{X_i}(y)$. For $P \subseteq C, X_i, \neg X_i \subseteq U$, where $\neg X_i = U - X_i$, $y \in U$, it is defined as

$$\epsilon^{'P}_{X_i}(y) = \frac{|I_P(y) \cap \neg X_i|}{|X_i|}. \tag{2.4}$$

In the numerator of (2.4) there is the number of objects in U that do not belong to class X_i and are indiscernible with object y. In the denominator of (2.4) there is the number of objects in U that belong to class X_i. This measure represents the ratio of objects $z \in U$ that are counterexamples to the implication $z \in I_P(y)$ implies $z \in X_i$ to the total number of objects in X_i. It lacks the likelihood interpretation that we give for $\epsilon^P_{X_i}(y)$. It should be noticed that $\epsilon^{'P}_{X_i}(y)$ may have low values for classes X_i that have high cardinality. In the example of medical diagnosis, the $\epsilon^{'}$ measure would express the ratio of the number of patients that have the same signs and does not suffer from the considered disease to the number of all known patients that suffer from the considered disease.

A gain-type consistency measure that can be considered in VC-IRSA is ($\overline{\mu}$ measure
(Błaszczyński et al., 2007b) is $\overline{\mu}^P_{X_i}(y)$. For $P \subseteq C$, $X_i \subseteq U$, $y \in U$, it is defined as

$$\overline{\mu}^P_{X_i}(y) = max_{R \subseteq P} \frac{|I_R(y) \cap X_i|}{|I_R(y)|}. \tag{2.5}$$

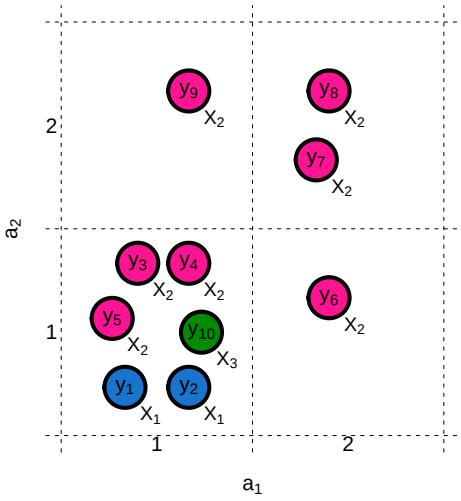
Consistency measure $\overline{\mu}^P_{X_i}(y)$ is calculated as a maximum rough membership to class X_i over all subsets G of the set of attributes P. An interpretation of this measure in the example of medical diagnosis would be that it expresses the maximal consistency measured by rough membership for any subset of signs and considered disease. Thus, this measure says which signs from all medical signs detected during examination of a given patient are the most relevant for the diagnosis that the patient suffers from the considered disease. The relevance is quantified by rough membership measure μ.

Example 2.3.1. *To observe differences between ϵ measure and measures that employ rough membership measure μ let us consider the example shown in Figure 2.4. There are ten objects from three classes $X_1 = \{y_1, y_2\}$, $X_2 = \{y_3, y_4, y_5, y_6, y_7, y_8, y_9\}$ and $X_3 = \{y_{10}\}$. These objects are described by attributes a_1 and a_2 ($P = \{a_1, a_2\}$) and form the following four granules:*

$$G_1 = \{y_1, y_2, y_3, y_4, y_5, y_{10}\}, \quad G_2 = \{y_6\}, \quad G_3 = \{y_7, y_8\}, \quad G_4 = \{y_9\}.$$

When we calculate the values of ϵ in granule G_1, we get the following values: $\epsilon^P_{X_1}(y_1) = \epsilon^P_{X_1}(y_2) = \frac{4}{8}$, $\epsilon^P_{X_2}(y_3) = \epsilon^P_{X_2}(y_4) = \epsilon^P_{X_2}(y_5) = 1$ and $\epsilon^P_{X_3}(y_{10}) = \frac{5}{9}$. For the same objects, we get the following values of rough membership: $\mu^P_{X_1}(y_1) = \mu^P_{X_1}(y_2) = \frac{2}{6}$, $\mu^P_{X_2}(y_3) = \mu^P_{X_2}(y_4) = \mu^P_{X_2}(y_5) = \frac{3}{6}$ and $\mu^P_{X_3}(y_{10}) = \frac{1}{6}$. One can observe that according to ϵ measure objects y_3 and y_4 are the most inconsistent in granule G_1 while according to rough membership measure μ they are the most consistent.

Rough membership takes into account local cardinalities in granule G_1 and its value reflects the fact that objects from class X_2 are the most frequent in G_1. Measure ϵ, on the contrary, takes into account that objects from class X_2 can be found in the granules surrounding granule G_1 while objects from classes X_1 and X_3 can be only found in granule G_1. Thus, in this sense, ϵ measure is global, while μ is local in the way in which objects are compared.



object	a_1	a_2	decision class
y_1	1	1	X_1
y_2	1	1	X_1
y_3	1	1	X_2
y_4	1	1	X_2
y_5	1	1	X_2
y_6	2	1	X_2
y_7	2	2	X_2
y_8	2	2	X_2
y_9	1	2	X_2
y_{10}	1	1	X_3

Figure 2.4: Illustration of difference between measures μ and ϵ in VC-IRSA.

2.4 Definition of lower and upper rough approximations using consistency measures

One of the most important features of the rough set reasoning about data is the separation of knowledge which is certainly consistent, from knowledge which is possibly inconsistent. The concepts of certain and possible correspond to lower and upper approximation of decision classes. In variable consistency rough set approaches, a key point is to find a sufficient evidence for assignment of objects to lower and upper approximations of a particular decision class.

Each set X, may include objects for which, due to inconsistency, we are unable to find enough evidence for their membership to X. In such a case, we can approximate set X by two sets, the *P-lower approximation* and the *P-upper approximation* of X, where $P \subseteq C$. Let us give generic definitions of P-lower approximations of set X. These definitions involve some *consistency measures* that express the evidence for membership to set X. These measures can be of gain or cost type. 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. In this thesis, we investigate desirable properties of consistency measures.

P-lower approximation

For $P \subseteq C, X \subseteq U, y \in U$, given a gain-type consistency measure $f_X^P(y)$ and a lower

threshold α_X, we get the following definitions of P-lower approximation of set X:

$$\underline{P}^{\alpha_X}(X) = \{y \in U : f_X^P(y) \geq \alpha_X\} \tag{2.6}$$

$$\text{or} \quad \underline{P}^{\alpha_X}(X) = \{y \in X : f_X^P(y) \geq \alpha_X\}. \tag{2.7}$$

Analogically, given a cost-type consistency measure $g_X^P(y)$ and an upper threshold β_X, we get the following definitions:

$$\underline{P}^{\beta_X}(X) = \{y \in U : g_X^P(y) \leq \beta_X\} \tag{2.8}$$

$$\text{or} \quad \underline{P}^{\beta_X}(X) = \{y \in X : g_X^P(y) \leq \beta_X\}. \tag{2.9}$$

Let us remark a fundamental difference between definitions 2.6 and 2.7 as well as 2.8 and 2.9. This difference concerns the source of objects considered for inclusion in the P-lower approximation of set X either from U or from X itself. This feature will be more thoroughly discussed in section 2.5.

In the above definitions, gain-threshold $\alpha_X \in [0, A_X]$ and cost-threshold $\beta_X \in [0, B_X]$. These thresholds are parameters depending on the interpretation of the gain-type or cost-type consistency measure, respectively. They play the role of technical parameters influencing the degree of consistency of objects belonging to lower approximation of X.

Thus, the values of A_X and B_X also depend on the interpretation of the corresponding consistency measure. For example, in case of probabilistic P-lower approximation defined using the rough membership measure, $A_X = 1$ and value of gain-threshold $\alpha_X \in [0, 1]$ can be calculated using method presented in (Greco et al., 2007; Yao, 2007). This method is based on application of the Bayesian decision procedure in transformation of risk into the value of α_X.

The above definitions of P-lower approximations of set X relax the non-parametric definitions. Precisely, the non-parametric definition is as follows:

$$\underline{P}(X) = \{y \in U : I_P(y) \subseteq X\} = \{y \in X : I_P(y) \subseteq X\},$$

An obvious condition of this relaxation is:

$$\underline{P}(X) \subseteq \underline{P}^{\alpha_X}(X), \tag{2.10}$$

$$\underline{P}(X) \subseteq \underline{P}^{\beta_X}(X). \tag{2.11}$$

The definition of P-upper approximation and of P-boundary of set X make use of the complementarity property of rough approximations.

For $P \subseteq C, X, \neg X \subseteq U$, where $\neg X = U - X$, P-upper approximation of set X is defined as

$$\overline{P}^{\alpha_X}(X) = U - \underline{P}^{\alpha_X}(\neg X), \quad \overline{P}^{\beta_X}(X) = U - \underline{P}^{\beta_X}(\neg X), \tag{2.12}$$

while P-boundary of set X is defined as

$$Bn_P^{\alpha_X}(X) = \overline{P}^{\alpha_X}(X) - \underline{P}^{\alpha_X}(X), \quad Bn_P^{\beta_X}(X) = \overline{P}^{\beta_X}(X) - \underline{P}^{\beta_X}(X). \tag{2.13}$$

Let us remark that the notion of consistency was also used in IRSA, to measure consistency of the whole decision table (Düntsch and Gediga, 1998; Hu et al., 2006; Qian et al., 2008b,a). In this case, different instances of the entropy measure were applied instead of the quality of approximation. Entropy measures were also applied to define consistency of a granule composed of P-indiscernible objects (Qian et al., 2008a). In the case of the whole decision table, as well as in the case of a single granule, consistency was considered with respect to all possible classes from the decision table.

We understand consistency in a different way. We consider consistency of particular objects with respect to the approximated sets. Latter, in chapter 4, we also consider consistency of decision rules with respect to the approximated sets.

2.5 Monotonicity of Lower Approximations

Our motivation for proposing Variable Consistency Indiscernibility-based Rough Set Approaches (VC-IRSA) comes from the need of ensuring monotonicity of lower approximations of sets w.r.t. set of attributes (Błaszczyński et al., 2007b). Due to definition of the upper approximation based on complementarity w.r.t. the lower approximation, the monotonicity property also concerns the upper approximation. The main difference between VC-IRSA and VPRS (Ziarko, 1993, 2006), RB model (Ślęzak, 2005; Ślęzak and Ziarko, 2005), and PRS (Greco et al., 2005b) is that in VC-IRSA one considers for inclusion to P-lower approximations only these objects which belong to the approximated set (definition (2.7) and (2.9)). In VPRS, RB model and PRS, whole granules of indiscernible objects are considered for inclusion to P-lower approximations (definition (2.6) and (2.8)). Inclusion of whole granules to lower approximations leads to a disturbance presented in the following example.

Example 2.5.1. *Three objects presented in Figure 2.5 are described by set of condition attributes $P = \{a_1\}$. All these object are P-indiscernible (i.e., they belong to the same granule), while y_1, y_2 belong to class X_2 and y_3 belongs to class X_1. Let us assume that objects y_1, y_2 and y_3 have sufficient consistency of belonging to decision class X_2. Thus,*

they all are included in P-lower approximation of X_2 defined according to (2.6) or (2.8). We note $\underline{P}(X_2) = \{y_1, y_2, y_3\}$.

object	a_1	class
y_1	0	X_2
y_2	0	X_2
y_3	0	X_1

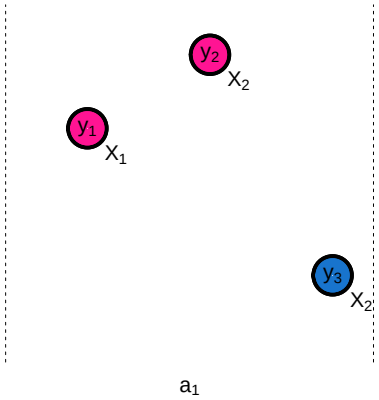


Figure 2.5: Illustration of non-monotonicity of definitions (2.6) or (2.8) on attribute a_1. Exemplary set of objects described by means of set P of one condition attribute a_1 as well as decision attribute d.

object	a_1	a_2	class
y_1	0	0	X_2
y_2	0	0	X_2
y_3	0	1	X_1

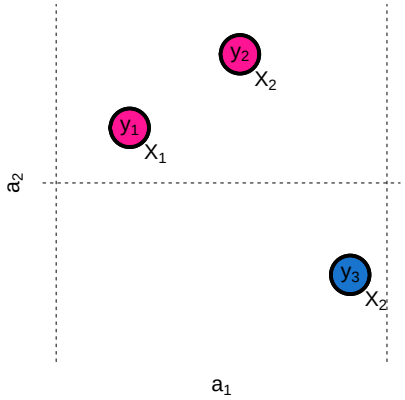


Figure 2.6: Illustration of non-monotonicity of definitions (2.6) or (2.8) on attribute a_1 and a_2. Exemplary set of objects described by means of set R of two condition attributes a_1 and a_2 as well as decision attribute d.

Now, the set of attributes is extended by attribute a_2 so that $R = \{a_1, a_2\}, R \supset P$. This situation is presented in Figure 2.6. Objects y_1 and y_2 are R-indiscernible with objects y_3. Nevertheless, to preserve monotonicity of P-lower approximation of class X_2 it would be necessary that object y_3 remains in R-lower approximation of X_2, so that $\underline{R}(X_2) = \{y_1, y_2, y_3\}$.

Remark that, as it is shown in Example 2.5.1, a granule included in a P-lower approximation may be composed of some inconsistent objects. Enlarging set P of attributes

to $R \supset P$, some P-indiscernible and inconsistent objects may become R-discernible and thus consistent. Monotonicity of lower approximation requires that if an object enters P-lower approximation it must also enter R-lower approximation. If we would like to preserve monotonicity of lower approximations defined according to (2.6) or (2.8), then we should keep in the R-lower approximation the R-discernible objects that do not belong to the approximated set. This, is not reasonable, however. Motivated by this remark, we consider the monotonicity properties only for approximations defined according to (2.7) or (2.9).

One can observe that properties of rough approximations defined in section 2.4 depend on properties of consistency measures $f_X^P(y)$ and $g_X^P(y)$. Thus, it is possible to formulate some properties with respect to these measures, which ensure desirable properties of rough approximations.

It is reasonable to require that measures $f_X^P(y)$ and $g_X^P(y)$ used to define the P-lower approximation according to (2.7) or (2.9) fulfill the following properties of monotonicity (henceforth called *monotonicity properties*):

(m1) Monotonicity with respect to (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 $f_X^P(y)$ is monotonically non-decreasing w.r.t. P, if and only if (iff)

$$f_X^P(y) \leq f_X^{P'}(y), \tag{2.14}$$

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

$$g_X^P(y) \geq g_X^{P'}(y). \tag{2.15}$$

(m2) Monotonicity w.r.t. set of objects $X \subseteq U$, when set X is augmented by a set of new objects X^Δ. 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 $f_X^P(y)$ is monotonically non-decreasing w.r.t. X, iff

$$f_X^P(y) \leq f_{X'}^P(y), \tag{2.16}$$

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

$$g_X^P(y) \geq g_{X'}^P(y). \tag{2.17}$$

Monotonicity properties (m1) and (m2) relate to the basic properties of rough sets. A rough set approach is called monotonic when the consistency measure used to define its lower approximation fulfills these monotonicity properties.

Property (m1) is particularly important. Property (m1) of measures $f_X^P(y)$ and $g_X^P(y)$ ensures monotonicity of P-lower approximation w.r.t. set of attributes $P \subseteq C$, defined according to (2.7) and (2.9), respectively. This property imposes that additional information about objects from U can only give more evidence for the observed assignment of objects to classes. In this case, additional information means a precisiation by more detailed description of considered objects using an extended set of attributes. Property (m1) is also concordant with the observation that additional attributes can only decrease comparability in the set of objects. When less objects are comparable, then also less inconsistent assignments to classes is observed.

Property (m2) of measures $f_X^P(y)$ and $g_X^P(y)$ ensures monotonicity of P-lower approximation w.r.t. set of objects $X \subseteq U$. Property (m2) states that when we consider two sets of objects $X' \supset X$, the evidence for membership to X' for objects from X should not be worse than the evidence for their membership to X. In other words, extension of class X_i by addition of new objects, should not negatively affect the evidence for membership of the objects to the extended class or union of classes.

In the following part of this section, we will show which of the monotonicity properties are held by measures defined in section 2.3.

2.5.1 Consistency measure μ

According to 2.1, gain-type measure rough membership μ is defined for $X_i \subseteq U$ w.r.t. $P \subseteq C$ as

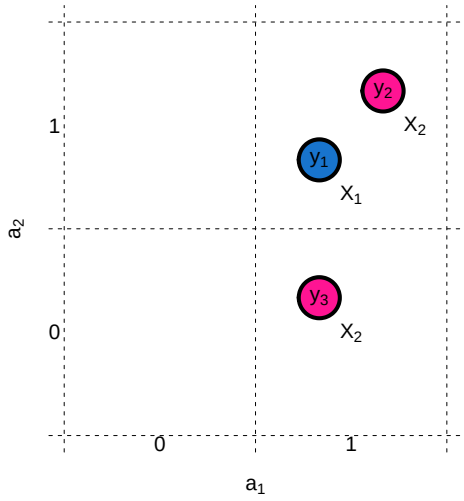
$$\mu_{X_i}^P(y) = \frac{|I_P(y) \cap X_i|}{|I_P(y)|}.$$

Theorem 2.5.1. *Measure $\mu_{X_i}^P(y)$ does not have property (m1), i.e., for all $P' \subseteq P'' \subseteq C, X_i \subseteq U, y \in U$.*

Proof. 2.5.1. The proof will be made by checking the situation presented in Figure 2.7. Measure $\mu_{X_i}^P(y)$ has property (m1) iff for all $P' \subseteq P'' \subseteq C, X_i \subseteq U, y \in U$, $\mu_{X_i}^P(y)$:

$$\mu_{X_i}^{P'}(y) \leq \mu_{X_i}^{P''}(y)$$

First, we consider attribute a_1 only, $P' = \{a_1\}$. All objects have the same value on that attribute (i.e., they all belong to the same granule). Thus, $\mu_{X_2}^{P'}(y_1) = \mu_{X_2}^{P'}(y_2) = \mu_{X_2}^{P'}(y_3) = 0.66$. Second, we consider set $P'' = \{a_1, a_2\}$. Then, we have two granules. The first one consists of objects y_1, y_2 and the other one is composed of object y_3. The value of rough membership to class X_2, $\mu_{X_i}^{P''}(y_1) = \mu_{X_i}^{P''}(y_2) = 0.5$. The value in the first granule has dropped after the extension of the set of attributes. □



object	a_1	a_2	decision class
y_1	1	1	X_1
y_2	1	1	X_2
y_3	1	0	X_2

Figure 2.7: Illustration of measure μ not having property (m1). Exemplary set of objects described by means of set P of attributes. Objects marked with 1 and 2 belong to class X_1 and X_2, respectively.

Even though property (m2) holds for rough membership (see proof 2.5.11), we refrain from using it to define VC-IRSA P-lower approximations.

2.5.2 Bayes Factor

According to 2.2, gain-type measure Bayes factor is defined for $y \in U$ and $X_i \subseteq U$ w.r.t. $P \subseteq C$ as

$$B^P_{X_i}(y) = \frac{|I_P(y) \cap X_i||\neg X_i|}{|I_P(y) \cap \neg X_i||X_i|}.$$

Theorem 2.5.2. *Measure $B^P_{X_i}(y)$ does not have property (m1), i.e., for all $P' \subseteq P'' \subseteq C, X_i \subseteq U, y \in U$, $B^P_{X_i}(y)$.*

Proof. 2.5.2. Let us come back to the example presented in Figure 2.7. For all $P' \subseteq P'' \subseteq C, X_i \subseteq U, y \in U$, measure $B^P_{X_i}(y)$ has property (m1) iff

$$B^{P'}_{X_i}(y) \leq B^{P''}_{X_i}(y).$$

We can observe that $B^{\{a_1\}}_{X_2}(y_2) = 1$, while $B^P_{X_2}(y_2) = 0.5$. □

Theorem 2.5.3. *Measure $B^P_{X_i}(y)$ does not have property (m2), i.e, for all $P \subseteq C$, $X_i \subseteq U$, $X_i' = X_i \cup X_i^\Delta$, $X_i^\Delta \cap U = \emptyset$, $y \in U$.*

Proof. 2.5.3. Measure $B^P_{X_i}(y)$ has property (m2) iff for all $P \subseteq C$, $X_i \subseteq U$, $X_i' = X_i \cup X_i^{\Delta}$, $X_i^{\Delta} \cap U = \emptyset$, $y \in U$

$$B^P_{X_i}(y) \leq B^P_{X_i'}(y).$$

In the example presented in Figure 2.7, we extend the set of objects by one new object y_4, which belongs to class X_2 and has the following description: $a_1 = 0$, $a_2 = 1$. We can notice that $B^P_{X_2}(y_2) = 0.5$ and $B^P_{X_2'}(y_2) = \frac{1}{3}$, where $X_2' = \{y_2, y_3, y_4\}$. □

Therefore, we refrain from using Bayes Factor to define VC-IRSA P-lower approximations because it is neither (m1) nor (m2) monotonic.

2.5.3 Consistency measure ϵ

According to (2.3), cost-type consistency measure ϵ is defined for $P \subseteq C, X_i, \neg X_i \subseteq U$, where $\neg X_i = U - X_i$, $y \in U$, as

$$\epsilon^P_{X_i}(y) = \frac{|I_P(y) \cap \neg X_i|}{|\neg X_i|}.$$

Theorem 2.5.4. *Measure $\epsilon^P_{X_i}(y)$ has property (m1), i.e., for all $P' \subseteq P'' \subseteq C, X_i \subseteq U, y \in U$, $\epsilon^P_{X_i}(y)$:*

$$\epsilon^{P'}_{X_i}(y) \geq \epsilon^{P''}_{X_i}(y).$$

Proof. 2.5.4. From the definition of rough granules $I_{P'}(y)$ and $I_{P''}(y)$, $P' \subseteq P'' \subseteq C$, $y \in U$,

$$I_{P'}(y) \supseteq I_{P''}(y)$$

for $X_i, \neg X_i \subseteq U$ being both independent of sets of considered attributes P' and P''. This implies:

$$\frac{|I_{P'}(y) \cap \neg X_i|}{|\neg X_i|} \geq \frac{|I_{P''}(y) \cap \neg X_i|}{|\neg X_i|} \Leftrightarrow \epsilon^{P'}_{X_i}(y) \geq \epsilon^{P''}_{X_i}(y).$$

□

Theorem 2.5.5. *Measure $\epsilon^P_{X_i}(y)$ has property (m2). More precisely, for all $P \subseteq C$, $X_i \subseteq U$, $X_i' = X_i \cup X_i^{\Delta}$, $X_i^{\Delta} \cap U = \emptyset$, $y \in U$:*

$$\epsilon^P_{X_i}(y) = \epsilon^P_{X_i'}(y).$$

Proof. 2.5.5. Since new objects are introduced to the universe U and to class $X_i \subseteq U$, thus for all sets of objects $X_i \subseteq U$, $X_i' \subseteq U'$, where $X_i' = X_i \cup X_i^{\Delta}$, $X_i^{\Delta} \cap U = \emptyset$,

$$\neg X_i = \neg X_i'.$$

For all $P \subseteq C$, $y \in U$, this implies:

$$\frac{|I_P(y) \cap \neg X_i|}{|\neg X_i|} = \frac{|I'_P(y) \cap \neg X'_i|}{|\neg X'_i|} \quad \Leftrightarrow \quad \epsilon^P_{X_i}(y) = \epsilon^P_{X'_i}(y),$$

where $I'_P(y)$ denotes a set of objects indiscernible with object y when considering set of attributes P and universe U'. □

Definition of a monotonic P-lower approximation of X_i using measure ϵ, requires that (2.9) takes the following form:

$$\underline{P}^{\beta_{X_i}}(X_i) = \{y \in X_i : \epsilon^P_{X_i}(y) \leq \beta_{X_i}\}, \tag{2.18}$$

where parameter β_{X_i} in $[0, 1]$ reflects the greatest degree of consistency acceptable to include object y in the P-lower approximation of set X_i.

Theorem 2.5.6. *Lower approximation defined according to (2.18) satisfies condition (2.11):*

$$\underline{P}(X_i) \subseteq \underline{P}^{\beta_{X_i}}(X_i).$$

Proof. 2.5.6. For each object $y \in X_i$, $I_P(y) \subseteq X_i$ iff $\epsilon^P_{X_i}(y) = 0$. □

2.5.4 Consistency measure ϵ'

According to (2.4), cost-type consistency measure $\epsilon'^P_{X_i}(y)$ is defined for $P \subseteq C, X_i, \neg X_i \subseteq U$, where $\neg X_i = U - X_i$, $y \in U$ as:

$$\epsilon'^P_{X_i}(y) = \frac{|I_P(y) \cap \neg X_i|}{|X_i|}. \tag{2.19}$$

Theorem 2.5.7. *Measure $\epsilon'^P_{X_i}(y)$ has property (m1), i.e., for all $P \subseteq P' \subseteq C, X_i \subseteq U, y \in U$:*

$$\epsilon'^P_{X_i}(y) \geq \epsilon'^{P'}_{X_i}(y).$$

Proof. 2.5.7. Analogous to proof 2.5.4 for measure $\epsilon^P_{X_i}(y)$ - only the common denominators in fractions are changed from $|\neg X_i|$ to $|X_i|$. □

Theorem 2.5.8. *Measure $\epsilon^P_{X_i}(y)$ has property (m2). More precisely, for all $P \subseteq C$, $X_i \subseteq U$, $X'_i = X_i \cup X_i^{\Delta}$, $X_i^{\Delta} \cap U = \emptyset$, $y \in U$:*

$$\epsilon'^P_{X_i}(y) = \epsilon'^P_{X'_i}(y).$$

Proof. 2.5.8. New objects are introduced to class $X_i \subseteq U$. Thus, for all sets of objects $X_i \subseteq U$, $X_i^{'} = X_i \cup X_i^{\Delta}$, where $X_i^{\Delta} \cap U = \emptyset$,

$$\neg X_i = \neg X_i^{'}, \quad |X_i| < |X_i^{'}|.$$

This implies that for all $P \subseteq C$, $y \in U$:

$$\frac{|I_P(y) \cap \neg X_i|}{|X_i|} > \frac{|I_P^{'}(y) \cap \neg X_i^{'}|}{|X_i^{'}|} \Leftrightarrow \epsilon_{X_i}^{'P}(y) > \epsilon_{X_i^{'}}^{'P}(y),$$

where $I_P^{'}(y)$ denotes a set of objects indiscernible with object y when considering set of attributes P and universe $U \cup X_i^{\Delta}$. □

Using consistency measure $\epsilon_{X_i}^{'P}(y)$, definition (2.9) takes the form:

$$\underline{P}^{\beta'_{X_i}}(X_i) = \{y \in X_i : \epsilon_{X_i}^{'P}(y) \leq \beta'_{X_i}\}, \tag{2.20}$$

where parameter $\beta'_{X_i} \in \left[0, \frac{|\neg X_i|}{|X_i|}\right]$ reflects the highest degree of consistency acceptable to include object y to the P-lower approximation of class X_i.

Theorem 2.5.9. *Lower approximation defined according to (2.20) satisfies condition (2.11):*

$$\underline{P}(X_i) \subseteq \underline{P}^{\beta'_{X_i}}(X_i). \quad \square$$

Proof. 2.5.9. For each object $y \in X_i$, $I_P(y) \subseteq X_i$ iff $\epsilon_{X_i}^{'P}(y) = 0$. □

2.5.5 Consistency measure $\overline{\mu}$

According to (2.5), gain-type consistency measure $\overline{\mu}_{X_i}^{P}(y)$ is calculated as a maximum rough membership to class X_i over all subsets R of the set of attributes P. For $P \subseteq C$, $X_i \subseteq U$, $y \in U$,

$$\overline{\mu}_{X_i}^{P}(y) = max_{R \subseteq P} \frac{|I_R(y) \cap X_i|}{|I_R(y)|}.$$

Theorem 2.5.10. *Measure $\overline{\mu}_{X_i}^{P}(y)$ has property (m1), i.e., for all $P' \subseteq P'' \subseteq C, X_i \subseteq U, y \in U$:*

$$\overline{\mu}_{X_i}^{P'}(y) \leq \overline{\mu}_{X_i}^{P''}(y).$$

Proof. 2.5.10. For all $P' \subseteq P'' \subseteq C, X_i \subseteq U, y \in U$, obviously,

$$\overline{\mu}_{X_i}^{P'}(y) = max_{R \subseteq P'} \frac{|I_R(y) \cap X_i|}{|I_R(y)|} \leq max_{R \subseteq P''} \frac{|I_R(y) \cap X_i|}{|I_R(y)|} = \overline{\mu}_{X_i}^{P''}(y).$$

□

Theorem 2.5.11. *Measure $\overline{\mu}^P_{X_i}(y)$ has property (m2), i.e., for all $P \subseteq C$, $X_i \subseteq U$, $X_i' = X_i \cup X_i^{\Delta}$, $U' = U \cup X_i^{\Delta}$, $X_i^{\Delta} \cap U = \emptyset$, $y \in U$:*

$$\overline{\mu}^P_{X_i}(y) \leq \overline{\mu}^P_{X_i'}(y).$$

Proof. 2.5.11. Let us consider all $P \subseteq C$, $X_i \subseteq U$, $X_i' = X_i \cup X_i^{\Delta}$, $U' = U \cup X_i^{\Delta}$, $X_i^{\Delta} \cap U = \emptyset$, $y \in U$. Since all new objects are added to class X_i, both numerator and denominator of fraction

$$\frac{|I_P(y) \cap X_i|}{|I_P(y)|} = \mu^P_{X_i}(y)$$

can increase only with the same number $k \geq 0$, equal to difference $|I'_P(y)| - |I_P(y)|$:

$$\frac{|I_P(y) \cap X_i| + k}{|I_P(y)| + k} = \frac{|I'_P(y) \cap X_i'|}{|I'_P(y)|} = \mu^P_{X_i'}(y),$$

where $I'_P(y)$ denotes a set of objects indiscernible with object y when considering set of attributes P and universe U'. Further, let us introduce the following notation: $a = |I_P(y) \cap X_i|$, $b = |I_P(y)|$, and let us notice that $a \leq b$. We can observe that

$$\mu^P_{X_i}(y) \leq \mu^P_{X_i'}(y), \tag{2.21}$$

which is proved in the following way:

$$\frac{a}{b} \leq \frac{a+k}{b+k} \Leftrightarrow a(b+k) \leq b(a+k) \Leftrightarrow ab + ak \leq ab + bk \Leftrightarrow ak \leq bk \Leftrightarrow a \leq b.$$

Thus,

$$\overline{\mu}^P_{X_i}(y) = max_{R \subseteq P}\, \mu^R_{X_i}(y) \leq max_{R \subseteq P}\, \mu^R_{X_i'}(y) = \overline{\mu}^P_{X_i'}(y).$$

□

We define the P-lower approximation of class X_i by means of $\overline{\mu}^P_{X_i}(y)$ and lower threshold $\overline{\alpha}_{X_i} \in [0, 1]$, as

$$\underline{P}^{\overline{\alpha}_{X_i}}(X_i) = \{y \in X_i : \overline{\mu}^P_{X_i}(y) \geq \overline{\alpha}_{X_i}\}. \tag{2.22}$$

Theorem 2.5.12. *Lower approximation defined according to (2.22) satisfies condition (2.10):*

$$\underline{P}(X_i) \subseteq \underline{P}^{\overline{\alpha}_{X_i}}(X_i).$$

Proof. 2.5.12. For each object $y \in X_i$, $I_P(y) \subseteq X_i$ iff $\overline{\mu}^P_{X_i}(y) = 1$. □

2.6 Properties of rough approximations from the viewpoint of rule induction

Distinction between P-lower and P-upper approximation of decision classes X_i, $(i = 1, 2, \ldots, n)$, constitutes the first step of rough set reasoning about data. The next step is induction of decision rules discussed in chapter 4. In this step information contained in approximations is transformed into knowledge represented by decision rules. From this perspective, the properties of rough set approximations that allow to induce effectively decision rules are of crucial importance.

In the definitions of P-lower approximations (2.7) and (2.9), that we used for monotonic VC-IRSA, only objects y that belong to X_i are included to P-lower approximation of class X_i. As it was already shown in Example 2.5.1 it is an important feature from the viewpoint of monotonicity of lower approximations in VC-IRSA. This feature however results in not all objects belonging to granule $I_P(y)$ being included in P-lower approximation.

A decision rule that assigns to a given class X_i, covers object y and objects that are P-indiscernible with y, i.e., if it covers object y it also covers all objects from granule $I_P(y)$. When we create a rule covering object y belonging to P-lower approximation of X_i and $I_P(y)$ happens to be composed of objects that do not belong to X_i there is no possibility to cover y while not covering objects from $I_P(y)$ that do not belong to X_i. This shows that P-lower approximations are not sufficient to define sets of objects covered by rules in VC-IRSA. P-lower approximation of class X_i does not include all objects that are covered by rule assigning to X_i. For this reason, we define P-positive, P-negative and P-boundary regions of class X_i in P-evaluation space, i.e., in $V_P = \prod_{j:a_j \in P} V_{a_j}$.

In IRSA, rules are induced from three types of approximations: lower approximations (certain rules), upper approximations (possible rules) and boundaries (approximate rules). In VC-IRSA, objects belonging to the P-positive regions are basis for induction of decision rules.

P-positive region

For $P \subseteq C, X_i, \neg X_i \subseteq U$, where $\neg X_i = U - X_i$, $y \in U$ and $\alpha_{X_i} \in [0, 1]$, $\beta_{X_i} \in [0, 1]$, P-positive regions of a class X_i are defined as:

$$POS_P^{\alpha_{X_i}}(X_i) = \bigcup_{y \in \underline{P}^{\alpha_{X_i}}(X_i)} I_P(y), \tag{2.23}$$

$$POS_P^{\beta_{X_i}}(X_i) = \bigcup_{y \in \underline{P}^{\beta_{X_i}}(X_i)} I_P(y), \tag{2.24}$$

where $\underline{P}^{\alpha_{X_i}}(X_i)$ is defined according to (2.7) and $\underline{P}^{\beta_{X_i}}(X_i)$ is defined according to (2.9). From (2.23 and 2.24), positive regions $POS_P^{\alpha_{X_i}}(X_i)$ and $POS_P^{\beta_{X_i}}(X_i)$ are composed of all objects y from P-lower approximation of X_i and objects that belong to granule $I_P(y)$ (i.e., all objects indiscernible from y). This can be denoted as property of P-positive regions:

$$\begin{aligned} &POS_P^{\alpha_{X_i}}(X_i) = \\ &= \left\{y \in X_i : f_{X_i}^P(y) \geq \alpha_{X_i}\right\} \cup \left\{y \in I_P(x) : x \in \underline{P}^{\alpha_{X_i}}(X_i) \wedge f_{X_i}^P(y) \geq \alpha_{X_i}\right\} = \\ &= \underline{P}^{\alpha_{X_i}}(X_i) \cup \left\{y \in I_P(x) : x \in \underline{P}^{\alpha_{X_i}}(X_i) \wedge f_{X_i}^P(y) \geq \alpha_{X_i}\right\}, \end{aligned} \tag{2.25}$$

$$\begin{aligned} &POS_P^{\beta_{X_i}}(X_i) = \\ &= \left\{y \in X_i : f_{X_i}^P(y) \leq \beta_{X_i}\right\} \cup \left\{y \in I_P(x) : x \in \underline{P}^{\beta_{X_i}}(X_i) \wedge g_{X_i}^P(y) \leq \beta_{X_i}\right\} = \\ &= \underline{P}^{\beta_{X_i}}(X_i) \cup \left\{y \in I_P(x) : x \in \underline{P}^{\beta_{X_i}}(X_i) \wedge g_{X_i}^P(y) \leq \beta_{X_i}\right\}. \end{aligned} \tag{2.26}$$

Lemma 2.6.1. *P-positive regions defined according to (2.23) and (2.24) differ in general from P-lower approximations defined according to (2.6) and (2.8).*

Let us observe that according to definitions (2.6) and (2.23), using property (2.25):

$$\begin{aligned} &\underline{P}^{\alpha_{X_i}}(X_i) = \left\{y \in U : f_{X_i}^P(y) \geq \alpha_{X_i}\right\} = \\ &= \left\{y \in X_i : f_{X_i}^P(y) \geq \alpha_{X_i}\right\} \cup \left\{y \in \neg X_i : f_{X_i}^P(y) \geq \alpha_{X_i}\right\}, \text{ while} \\ &POS_P^{\alpha_{X_i}}(X_i) = \bigcup_{y \in \underline{P}^{\alpha_{X_i}}(X_i)} I_P(y) = \\ &= \left\{y \in X_i : f_{X_i}^P(y) \geq \alpha_{X_i}\right\} \cup \left\{y \in I_P(x) : x \in \underline{P}^{\alpha_{X_i}}(X_i) \wedge f_{X_i}^P(y) \geq \alpha_{X_i}\right\}. \end{aligned}$$

P-lower approximation defined according to (2.6) contains all objects satisfying condition on consistency of belonging to a given class X_i. P-positive region contains only these objects that satisfy the condition and are indiscernible from objects belonging to the lower approximation of class X_i. The same can be shown for definitions (2.8) and (2.24).

Moreover, from the same reason, if one would consider a P-positive regions composed of objects indiscernible from object belonging to P-lower approximations defined by (2.6), (2.8) would differ from (2.23), (2.24).

We define P-negative and P-boundary regions of approximated sets, for $P \subseteq C$, X_i, $\neg X_i \subseteq U$, and $\alpha_{X_i} \in [0,1]$, $\beta_{X_i} \in [0,1]$, as the following:

P-negative region

$$NEG_P^{\alpha_{X_i}}(X_i) = POS_P^{\alpha_{X_i}}(\neg X_i) - POS_P^{\alpha_{X_i}}(X_i), \tag{2.27}$$

$$NEG_P^{\beta_{X_i}}(X_i) = POS_P^{\beta_{X_i}}(\neg X_i) - POS_P^{\beta_{X_i}}(X_i), \tag{2.28}$$

P-boundary region

$$BND_P^{\alpha_{X_i}}(X_i) = (U - POS_P^{\alpha_{X_i}}(X_i)) - NEG_P^{\alpha_{X_i}}(X_i) \tag{2.29}$$

$$BND_P^{\beta_{X_i}}(X_i) = (U - POS_P^{\beta_{X_i}}(X_i)) - NEG_P^{\beta_{X_i}}(X_i). \tag{2.30}$$

The following properties hold the P-positive, the P-negative and the P-boundary regions of class X_i and its complement $\neg X_i$.

Theorem 2.6.1. *For all $P \subseteq C, X_i, \neg X_i \subseteq U$, where $\neg X_i = U - X_i$, $y \in U$ and $\alpha_{X_i} \in [0,1]$, $\beta_{X_i} \in [0,1]$:*

$$BND_P^{\alpha_{X_i}}(X_i) = BND_P^{\alpha_{X_i}}(\neg X_i),$$
$$BND_P^{\beta_{X_i}}(X_i) = BND_P^{\beta_{X_i}}(\neg X_i).$$

Proof 2.6.1.

$$BND_P^{\alpha_{X_i}}(X_i) = (U - POS_P^{\alpha_{X_i}}(X_i)) - NEG_P^{\alpha_{X_i}}(X_i),$$
$$BND_P^{\alpha_{X_i}}(\neg X_i) = (U - POS_P^{\alpha_{X_i}}(\neg X_i)) - NEG_P^{\alpha_{X_i}}(\neg X_i).$$

Since for any sets A, B

$$A - B = (A - B) \cup (A - A) = A - (B \cap A),$$

then

$$NEG_P^{\alpha_{X_i}}(X_i) = POS_P^{\alpha_{X_i}}(\neg X_i) - POS_P^{\alpha_{X_i}}(X_i) =$$
$$= POS_P^{\alpha_{X_i}}(\neg X_i) - (POS_P^{\alpha_{X_i}}(X_i) \cap POS_P^{\alpha_{X_i}}(\neg X_i)),$$

and

$$NEG_P^{\alpha_{X_i}}(\neg X_i) = POS_P^{\alpha_{X_i}}(X_i) - POS_P^{\alpha_{X_i}}(\neg X_i) =$$
$$= POS_P^{\alpha_{X_i}}(X_i) - (POS_P^{\alpha_{X_i}}(\neg X_i) \cap POS_P^{\alpha_{X_i}}(X_i)).$$

Thus, if we consider that all together,

$$BND_P^{\alpha_{X_i}}(X_i) = (U - POS_P^{\alpha_{X_i}}(X_i)) - \big(POS_P^{\alpha_{X_i}}(\neg X_i) - (POS_P^{\alpha_{X_i}}(X_i) \cap POS_P^{\alpha_{X_i}}(\neg X_i))\big),$$

$$BND_P^{\alpha_{X_i}}(\neg X_i) = (U - POS_P^{\alpha_{X_i}}(\neg X_i)) - \big(POS_P^{\alpha_{X_i}}(X_i) - (POS_P^{\alpha_{X_i}}(\neg X_i) \cap POS_P^{\alpha_{X_i}}(X_i))\big),$$

and

$$BND_P^{\alpha_{X_i}}(X_i) = BND_P^{\alpha_{X_i}}(\neg X_i).$$

The same can be shown for $BND_P^{\beta_{X_i}}(X_i)$ and $BND_P^{\beta_{X_i}}(\neg X_i)$. □

The boundary region of approximated set and the boundary region of its complement are equal. This property seems natural since boundary regions consist of objects that we are uncertain to assign either to lower approximation of considered set or to lower approximation of its complement. The objects that lie in the boundary may be in general covered by rules assigning to the set or to its complement.

Theorem 2.6.2. *For all $P \subseteq C, X_i, \neg X_i \subseteq U$, where $\neg X_i = U - X_i$, $y \in U$ and $\alpha_{X_i} \in [0,1]$, $\beta_{X_i} \in [0,1]$:*

$$NEG_P^{\alpha_{X_i}}(X_i) \cap NEG_P^{\alpha_{X_i}}(\neg X_i) = \emptyset, \tag{2.31}$$

$$NEG_P^{\beta_{X_i}}(X_i) \cap NEG_P^{\beta_{X_i}}(\neg X_i) = \emptyset. \tag{2.32}$$

Proof 2.6.2.

$$NEG_P^{\alpha_{X_i}}(X_i) = POS_P^{\alpha_{X_i}}(\neg X_i) - POS_P^{\alpha_{X_i}}(X_i),$$
$$NEG_P^{\alpha_{X_i}}(\neg X_i) = POS_P^{\alpha_{X_i}}(X_i) - POS_P^{\alpha_{X_i}}(\neg X_i),$$

and

$$NEG_P^{\beta_{X_i}}(X_i) = POS_P^{\beta_{X_i}}(\neg X_i) - POS_P^{\beta_{X_i}}(X_i),$$
$$NEG_P^{\beta_{X_i}}(\neg X_i) = POS_P^{\beta_{X_i}}(X_i) - POS_P^{\beta_{X_i}}(\neg X_i).$$

□

Intersection of negative region of approximated set and negative region of its complement is an empty set. This is an important property from both rough set theory perspective and rule induction perspective. The negative region contains objects for which we are sure that they don't belong to the considered set. Thus, one should expect that negative regions of complementary sets do not have any common part.

Once decision rules are learned, they can be applied by a classifier (see chapter 5) to suggest assignment of objects to classes. The rules are learned from P-positive regions of the decision classes. This type of structuring of the data involves an a priori restriction

of the set of objects, on which the classifier is learned. The rough set analysis enables estimation of the attainable predictive accuracy before learning of a classifier occurs. A classifier learned on P-positive regions of decision classes *may* correctly assign object $y \in X_i$ to class X_i if y belongs to the P-positive region of X_i.

We define λ measure that estimates the predictive accuracy that may be attained by the classifier: λ measure

$$\lambda_P^{\boldsymbol{\alpha}_X} = \frac{\bigcup_{i=1}^{n} \left| X_i \cap POS_P^{\alpha_{X_i}}(X_i) \right|}{|U|}, \tag{2.33}$$

$$\lambda_P^{\boldsymbol{\beta}_X} = \frac{\bigcup_{i=1}^{n} \left| X_i \cap POS_P^{\beta_{X_i}}(X_i) \right|}{|U|}, \tag{2.34}$$

where n is the number of the decision classes. This measure estimates the ratio of objects in U that may be learned by the classifier. It can be thus used to characterize the data set on which the classifier is learned.

It is worth noting that in Ziarko and Ślęzak also defined positive, negative and boundary regions in Variable Precision Rough Sets (Ziarko, 1993), Bayesian rough set model (Ślęzak and Ziarko, 2005; Ziarko, 2006) and Bayesian Rough Sets (Ślęzak, 2005). However, their definitions differ from those presented here. In their definitions, an object that has a consistency of belonging to a given set X higher than a given threshold enters positive region. An object that has the consistency lower than the threshold enters negative region. The rest of objects (i.e., those that have the consistency equal the threshold) are counted in the boundary region.

2.7 Summary

In this chapter, we presented definitions of several consistency measures that can be used to define VC-IRSA. Two monotonicity properties (m1), (m2) were considered for these measures. We have stressed the importance of some monotonicity properties of the consistency measure used in the definition of a lower approximation. The monotonicity properties of the considered measures are summarized in table 2.1. We have proposed two types of measures enjoying the above monotonicity properties. The first type stems from consistency measure ϵ, which is a catch-all likelihood measure. This consistency measure has a comprehensible probabilistic explanation. It has also a close relation with the Bayes factor and confirmation measure l. We proposed a kind of complementary measure to ϵ denoted by ϵ'. One can observe that for ϵ, there is a tendency of including relatively more objects to lower approximations when the approximated class or union of classes has low cardinality. On the other hand, one can observe that for ϵ', there

Table 2.1: Monotonicity of consistency measures considered for VC-IRSA.

consistency measure	(m1)	(m2)
$\mu_X^P(y)$	no	yes
$B_X^P(y)$	no	no
$\epsilon_X^P(y)$	yes	yes
$\epsilon_X^{'P}(y)$	yes	yes
$\overline{\mu}_X^P(y)$	yes	yes

is a tendency of including relatively more objects to lower approximations when the approximated class or union of classes has high cardinality.

Monotonic measures of the second type stem from consistency measure μ. They require to take into account all subsets of the set of considered attributes. Computation of lower approximations defined by means of monotonic measure $\overline{\mu}$ is a computationally intensive problem, equivalent to induction of a set of all rules. On the other hand, computation of such approximations and rule induction can be combined, and thus the total time would be of the same order as the time for induction of all rules.

We defined monotonic lower approximations for those of consistency measures. These lower approximations have all considered monotonicity properties. Further, the monotonic lower approximations were used to define positive, negative and boundary regions which, as it was presented, are more desirable basis for the induction of the decision rules. Moreover, we defined a measure that estimate the predictive accuracy attainable to a classifier learned on positive regions.

As a conclusion, we can recommend using consistency measure ϵ or $\epsilon^{'}$. These measures have all required monotonicity properties and are much less computationally intensive than the monotonic measures of the second type.

CHAPTER 3

Variable Consistency Dominance-based Rough Set Approaches (VC-DRSA)

3.1 Problem statement and basic definitions

Dominance-based rough set approach (DRSA) (Greco et al., 1995, 1999b; Słowiński et al., 2009) has been proposed as an extension of the Indiscernibility-based rough set approach (IRSA) reminded in chapter 2. The type of inconsistency handled by DRSA is more general than inconsistency handled by IRSA. DRSA uses the *dominance relation* where IRSA uses the indiscernibility relation. Application of the dominance relation enables reasoning about data sets described by *criteria* (i.e., attributes with preference-ordered domains (scales)). For example, a quality measure can be considered as a gain-type criterion. If we consider this gain-type criterion and two objects y_1 and y_2, where object y_1 has "good" quality while object y_2 has "poor" quality then object y_1 dominates (i.e., is not worse than) object y_2 taking into account the quality criterion. If no order on quality scale is considered, as it is in case of IRSA, than we can only state that objects y_1 and y_2 are discernible (i.e, different) on the quality attribute.

In classification problems considered by DRSA, analogously to IRSA, objects from U are assigned to decision classes X_i, $(i = 1, 2, \ldots n)$, according to the value of decision attribute d. For simplicity, we assume here, without loss of generality, that set of decision attributes D is a singleton $D = \{d\}$ and that the domain of decision attribute d is ordered such that a higher level value is better than any lower level value. 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 relationships 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 y may belong. 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 $i = 1, 2, \ldots, n-1$. All objects from a particular upward or downward union of objects $X_i^{\geq}$ and $X_i^{\leq}$ can also be referred to as set of objects $X^{\geq}$ and $X^{\leq}$. Also, for given unions $X_i^{\geq}$, $X_i^{\leq}$, all objects belonging to the opposite unions of classes $X_{i-1}^{\leq}$, $X_{i+1}^{\geq}$ respectively, are denoted as $\neg X^{\geq}$, $\neg X^{\leq}$.

The inconsistency detected by DRSA comes from the violation of the *dominance principle* which says that if evaluations of object y_1 on all considered criteria are not worse than evaluations of object y_2, then y_1 should be assigned to a class not worse than y_2. The dominance principle, and more formally, the definition of the dominance relation, as well as the definition of granules of knowledge that it implies, are discussed in section 3.2.

DRSA makes distinction of objects from any set $X^{\geq}$ and $X^{\leq}$ into two disjoint sets of objects: those consistently belonging to $X^{\geq}$ or $X^{\leq}$, and inconsistent objects. The set of objects consistently belonging to $X^{\geq}$ or $X^{\leq}$ is called lower approximation of $X^{\geq}$ or $X^{\leq}$. The upper approximation, on the other hand, includes all objects from the approximated set and objects that are inconsistent. In other words, objects which, according to available knowledge, certainly belong to $X^{\geq}$ or $X^{\leq}$ are assigned to lower approximation of $X^{\geq}$ or to lower approximation of $X^{\leq}$. Objects which may possibly belong to $X^{\leq}$ or $X^{\geq}$ are classified to upper approximation of $X^{\leq}$ or $X^{\geq}$.

The motivation for introduction of Variable Consistency Dominance-base Rough Set Approaches (VC-DRSA) is the same as in case of Variable Consistency Indiscernibility-based Rough Set Approaches (VC-IRSA). These approaches relax the definition of lower approximation of a union of classes and admit those objects for which there is enough evidence for their membership to the union of classes. The aim of this relaxation is not to allow inconsistent objects to make further reasoning difficult when sufficient evidence for their membership can be found. The evidence for membership is estimated by consistency measures (see sections 2.3 and 3.3). The approaches that provide this mechanism include Variable Consistency Dominance-based Rough Set Approaches (VC-DRSA) (Greco et al., 2000b, 2005a; Błaszczyński et al., 2006, 2007b; Greco et al., 2008b;

Błaszczyński et al., 2009a) and Variable-Precision Dominance-based Rough Set Analysis (VP-DRSA) (Inuiguchi and Yoshioka, 2006).

A statistical approach to treatment of inconsistent objects within DRSA has also been considered (Kotłowski et al., 2008; Kotłowski and Słowiński, 2008). This approach originates from statistical learning and statistical decision theory (Hastie et al., 2009). It uses the notion of stochastic dominance and probabilities of object belonging to unions of classes which are further replaced by their maximum likelihood (ML) estimators. Stochastic lower approximations are composed of objects for which values of estimators are higher then a given threshold. In this sense, this approach is similar to probabilistic approaches considered in the thesis. However, the ML estimation of the probabilities involves solving optimization problems. It makes the whole approach not as traceable as VC-DRSA, which, for the same purpose, employs relatively simple (and thus easy to interpret) consistency measures. Moreover, statistical approach involves relabeling of objects (i.e., change of class to which object belongs to the more probable one) which further reduces traceability. Relabeling is not considered in VC-DRSA. The described above features make statistical DRSA hard to comparable to the approaches considered in this thesis.

In this chapter, the Monotonic Variable Consistency Dominance-based Rough Set Approaches are defined and justified. In section 3.2, we define the granules of knowledge that are implied by the dominance relation. Then, in section 3.3, we focus on the consistency measures. Further, in section 3.4, we define the lower and the upper approximations. In sections 3.5 and 3.6, we investigate properties of these approximations. In section 3.5, the monotonicity properties of rough approximations are of our special concern. The chapter is summarized in section 3.7.

3.2 Granules of knowledge

As it was already mentioned, DRSA provides a mechanism for reasoning about data through granules of knowledge that are more general than granules used in IRSA. Indiscernibility granules are bounded sets in the attribute space corresponding to condition attributes C and decision attributes D (Słowiński et al., 2002b). An indiscernibility granule is a point in the C space. When condition attributes from C and decision attributes 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 (Greco et al., 1999b, 2001a; Słowiński et al., 2005). Dominance relation makes a partition of universe U into granules being *dominance cones*. The dominance relation

dominance relation

D_P is defined for a non-empty subset of criteria $P \subseteq C$ as

$$D_P = \{(y, z) \in U \times U : f(y, a_i) \succeq f(z, a_i) \text{ for all } a_i \in P\},$$

where $f(y, a_i) \succeq f(z, a_i)$ means "y is at least as good as z w.r.t. criterion a_i". Dominance relation D_P is a partial preorder (i.e. reflexive and transitive). For each object $y \in U$ two dominance cones are defined with respect to subset $P \subseteq C$. The P-positive dominance cone $D_P^+(y)$ is composed of all objects that are dominating y. The P-negative dominance cone $D_P^-(y)$ is composed of all objects that are dominated by y. Formal definitions of dominance cones are as follows:

dominance cones

$$D_p^+(y) = \{z \in U : zD_Py\},$$
$$D_p^-(y) = \{z \in U : yD_Pz\}.$$

Dominance cones are open sets in the attribute spaces corresponding to condition attributes C and decision attributes D. With each point in the C space two dominance cones D_P^+ and D_P^- are associated. Both of these dominance cones have their origin in the point of the space by which they are defined, but they are not bounded by any end.

Similarly as in case of VC-IRSA, we do not consider in this chapter discretization of of attributes. Nevertheless, all assumptions made in this subject for VC-IRSA in Section 2.2 are valid for VC-DRSA.

Example 3.2.1. *Consider the following example of U described by means of set P of two condition criteria q_1 and q_2 in in Table 3.1.*

object	q_1	q_2	d
y_1	4	4	2
y_2	4	8	2
y_3	7	2	2
y_4	2	7	1
y_5	5	5	1
y_6	2	2	1

Table 3.1: Exemplary set of objects described by means of set P of two gain-type condition criteria q_1 and q_2 as well as decision gain-type criterion d.

First, let us consider criterion q_1 alone. Further, let us consider P-positive dominance conses only. This example is illustrated in Figure 3.1. The P-positive dominance cones are the following:

$$D^+_{q_1}(y_4) = D^+_{q_1}(y_6) = \{y_1, y_2, y_3, y_4, y_5, y_6\},$$
$$D^+_{q_1}(y_1) = D^+_{q_1}(y_2) = \{y_1, y_2, y_3, y_5\},$$
$$D^+_{q_1}(y_5) = \{y_3, y_5\},$$
$$D^+_{q_1}(y_3) = \{y_3\}.$$

Positive dominance cones $D^+_{q_1}$ based on object y_4 and object y_6 are the same (Figure 3.1a). They contain all objects in the analysed data set. Positive dominance cones cones $D^+_{q_1}$ based on indiscernible on criterion q_1 objects y_1 and y_2 are also the same. In general, this dependency holds for indiscernible objects. The following relation can be observed: $D^+_{q_1}(y_4) = D^+_{q_1}(y_6) \supset D^+_{q_1}(y_1) = D^+_{q_1}(y_2) \supset D^+_{q_1}(y_5) \supset D^+_{q_1}(y_3)$. It is worth noting that inconsistency introduced by violation of the dominance principle between criteria q_1 and d occurs in case of cones $D^+_{q_1}(y_1)$ and $D^+_{q_1}(y_2)$ (Figure 3.1b). They contain object y_5 belonging to class X_1 which is worse than X_2.

The negative dominance cones are shown in Figure 3.2. They are the following:

$$D^-_{q_1}(y_4) = D^-_{q_1}(y_6) = \{y_4, y_6\},$$
$$D^-_{q_1}(y_1) = D^-_{q_1}(y_2) = \{y_1, y_2, y_4, y_6\},$$
$$D^-_{q_1}(y_5) = \{y_1, y_2, y_4, y_5, y_6\},$$
$$D^-_{q_1}(y_3) = \{y_1, y_2, y_3, y_4, y_5, y_6\}.$$

For P-negative dominance conses in Figure 3.2 the following relation can be observed: $D^-_{q_1}(y_4) = D^+_{q_1}(y_6) \subset D^+_{q_1}(y_1) = D^+_{q_1}(y_2) \subset D^+_{q_1}(y_5) \subset D^+_{q_1}(y_3)$. Violation of the dominance principle occurs only in case of cone $D^+_{q_1}(y_5)$ (Figure 3.1c). Objects dominated by y_5 which belongs to class X_1 include y_1 and y_2 which are belonging to class X_2. Thus, objects y_1, y_2 and y_5 are inconsistent.

Example 3.2.2. *When we consider both q_1 and q_2 condition criteria from Table 3.1,*

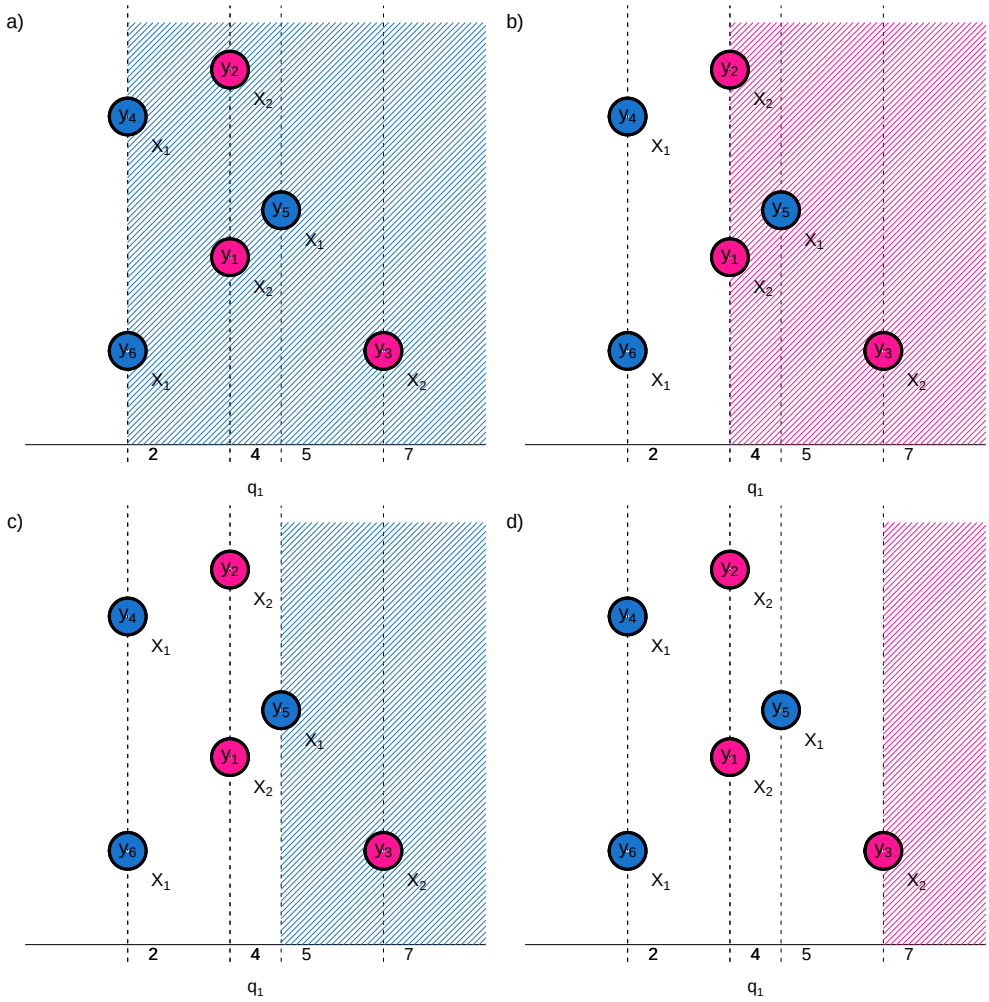


Figure 3.1: Dominance cones $D^+_{q_1}$ in exemplary set of objects described by means of one gain-type condition criterion q_1.

we get dominating cones illustrated in Figure 3.3. They are the following :

$$D^+_{q_1,q_2}(y_6) = \{y_1, y_2, y_3, y_4, y_5, y_6\},$$
$$D^+_{q_1,q_2}(y_1) = \{y_1, y_2, y_5\},$$
$$D^+_{q_1,q_2}(y_4) = \{y_2, y_4\},$$
$$D^+_{q_1,q_2}(y_5) = \{y_5\},$$
$$D^+_{q_1,q_2}(y_2) = \{y_2\},$$
$$D^+_{q_1,q_2}(y_3) = \{y_3\}.$$

Only object y_6 remains dominated by all other objects. The inconsistency is observed between objects y_1 and y_5. Object y_1 remains dominated by object y_5.

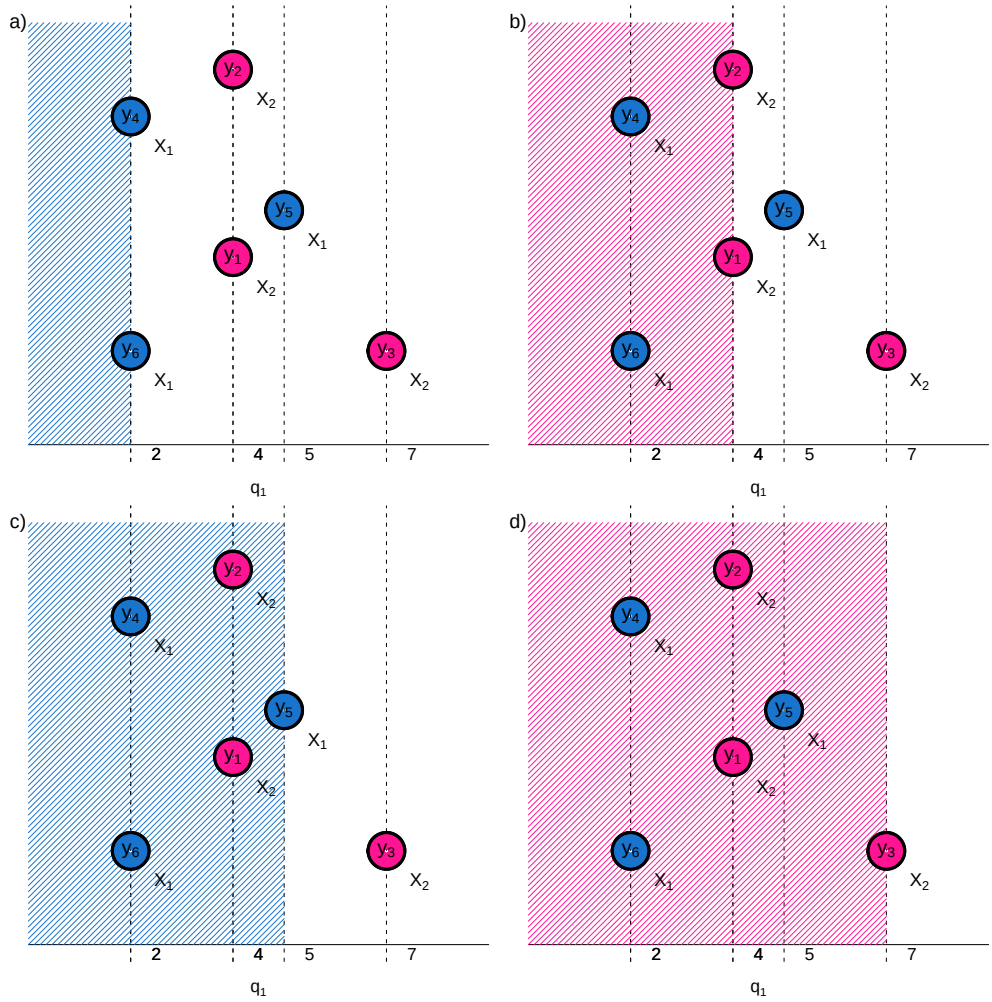


Figure 3.2: Dominance cones $D^-_{q_1}$ in exemplary set of objects described by means of one condition gain-type criterion q_1.

We get also the following negative cones illustrated in Figure 3.4:

$$
\begin{aligned}
D^-_{q_1,q_2}(y_2) &= \{y_1, y_2, y_4, y_6\},\\
D^-_{q_1,q_2}(y_5) &= \{y_1, y_5, y_6\},\\
D^-_{q_1,q_2}(y_4) &= \{y_4, y_6\},\\
D^-_{q_1,q_2}(y_1) &= \{y_1, y_6\},\\
D^-_{q_1,q_2}(y_3) &= \{y_3, y_6\},\\
D^-_{q_1,q_2}(y_6) &= \{y_6\},
\end{aligned}
$$

In this case, shown in Figure 3.4, inconsistent objects are y_1 and y_5. The observed

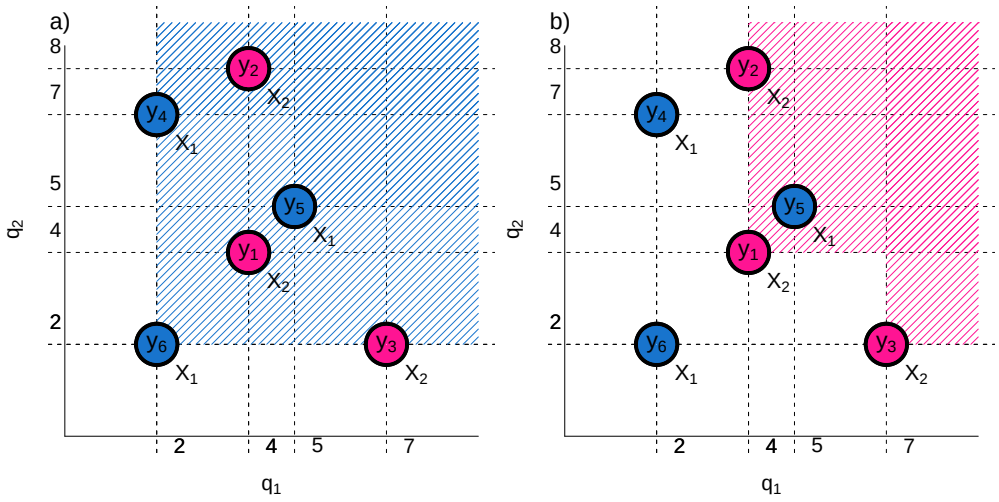


Figure 3.3: Dominance cones $D^{+}_{q_1,q_2}$ in exemplary set of objects described by means of one condition criteria q_1 and q_2.

inconsistency is caused by object y_5 dominating object y_1 which violates the dominance principle.

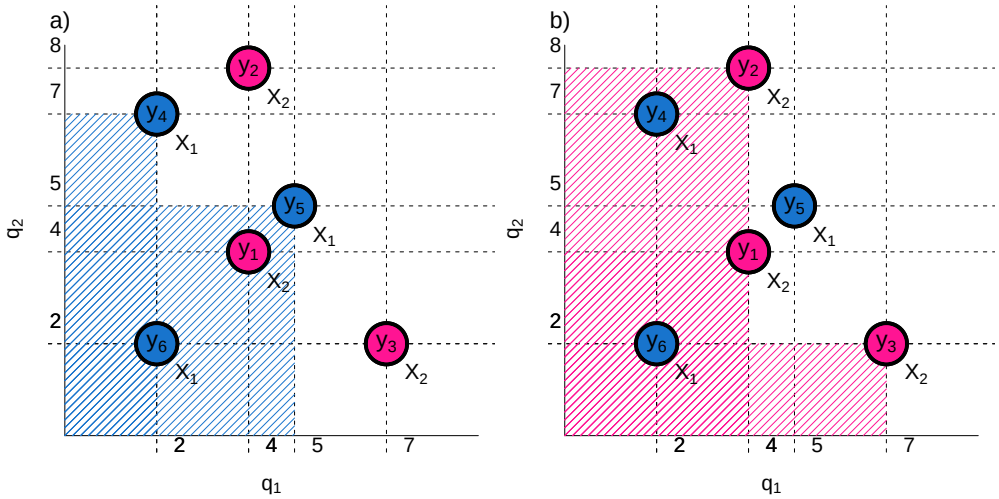


Figure 3.4: Dominance cones $D^{-}_{q_1,q_2}$ in exemplary set of objects described by means of one condition criteria q_1 and q_2.

DRSA takes into account monotonic relationships between evaluations of objects on particular criteria and assignment of these objects into decision classes which are also ordered. 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.

3.3 Consistency principle and consistency measures

Analogously to IRSA, the inconsistencies detected in data can be handled by precisiation (i.e., by extension of the set of attributes A). However, to treat data when precisiation is not possible, measures of consistency are defined. These measures quantify the level of inconsistency in granules of knowledge, so that this information can be taken into account in further reasoning.

Following (Greco et al., 1995, 1998b, 1999b; Słowiński et al., 2005), we reformulate definitions of monotonic approaches presented in section 2.3, replacing indiscernibility relation by dominance relation. We also present measures specific for VC-DRSA.

μ measure

In this case, gain-type rough membership measures are defined for $P \subseteq C$, $X^{\geq}, X^{\leq} \subseteq U$, $y \in U$, as

$$\mu^P_{X^{\geq}}(y) = \frac{\left|D^+_P(y) \cap X^{\geq}\right|}{\left|D^+_P(y)\right|}, \quad \mu^P_{X^{\leq}}(y) = \frac{\left|D^-_P(y) \cap X^{\leq}\right|}{\left|D^-_P(y)\right|}, \tag{3.1}$$

where $X^{\geq}$, $X^{\leq}$ denote upward and downward unions of decision classes, respectively. Values of rough membership $\mu^P_{X^{\geq}}(y)$ and $\mu^P_{X^{\leq}}(y)$ can be interpreted as estimates of conditional probability $Pr(x \in X^{\geq} | xD_P y)$ and $Pr(x \in X^{\leq} | yD_P x)$, respectively.

μ' measure

We presented a modification of rough membership measures that has desirable properties in VC-DRSA (Błaszczyński et al., 2006). The resulting gain-type consistency measures are defined for $P \subseteq C$, $X^{\geq}, X^{\leq} \subseteq U$, $y \in U$, as

$$\mu'^P_{X^{\geq}}(y) = \max_{x \in D^-_P(y)} \frac{\left|D^+_P(x) \cap X^{\geq}\right|}{\left|D^+_P(x)\right|}, \quad \mu'^P_{X^{\leq}}(y) = \max_{x \in D^+_P(y)} \frac{\left|D^-_P(x) \cap X^{\leq}\right|}{\left|D^-_P(x)\right|}. \tag{3.2}$$

Values of rough membership $\mu^P_{X^{\geq}}(y)$ and $\mu^P_{X^{\leq}}(y)$ can be interpreted as maximal estimates of probability $Pr(x \in X^{\geq} | xD_P y)$ in dominance cone $D^-_P(y)$ and probability $Pr(x \in X^{\leq} | yD_P x)$ in dominance cone $D^+_P(y)$, respectively.

Bayes factor

Formulation of the gain-type Bayes factors for $P \subseteq C$, $X^{\geq}, X^{\leq} \subseteq U$, $y \in U$, is as follows:

$$B^P_{X^{\geq}}(y) = \frac{|D^+_P(y) \cap X^{\geq}||\neg X^{\geq}|}{|D^+_P(y) \cap \neg X^{\geq}||X^{\geq}|}, \quad B^P_{X^{\leq}}(y) = \frac{|D^-_P(y) \cap X^{\leq}||\neg X^{\leq}|}{|D^-_P(y) \cap \neg X^{\leq}||X^{\leq}|}. \tag{3.3}$$

The Bayes factor can be seen as a ratio of two conditional probabilities $Pr(x \in D^+_P(y) | x \in X^{\geq})$ and $Pr(x \in D^-_P(y) | x \in \neg X^{\geq})$ or $Pr(x \in D^-_P(y) | x \in X^{\leq})$ and $Pr(x \in D^+_P(y) | x \in \neg X^{\leq})$, respectively.

β precision

Another gain-type consistency measures called β precisions are introduced for Variable-Precision Dominance-based Rough Set Analysis (VP-DRSA) in (Inuiguchi and Yoshioka, 2006). These measures are defined for $P \subseteq C$, $X^{\geq}, X^{\leq} \subseteq U$, $y \in U$, as follows:

$$\beta^P_{X^{\geq}}(y) = \frac{|D^-_P(y) \cap X^{\geq}|}{|D^-_P(y) \cap X^{\geq}| + |D^+_P(y) \cap \neg X^{\geq}|}, \tag{3.4}$$

$$\beta^P_{X^{\leq}}(y) = \frac{|D^+_P(y) \cap X^{\leq}|}{|D^+_P(y) \cap X^{\leq}| + |D^-_P(y) \cap \neg X^{\leq}|}. \tag{3.5}$$

Precisions $\beta^P_{X^{\geq}}(y)$ (or $\beta^P_{X^{\leq}}(y)$) are defined as a ratio of the number of objects that belonging to dominance cone $D^-_P(y)$ (or $D^+_P(y)$) and to union of classes $X^{\geq}$ (or $X^{\leq}$) to the number of these objects increased by the number of objects from the opposite dominance cone $D^+_P(y)$ (or $D^-_P(y)$) that belong to the the opposite union $\neg X^{\geq}$ (or $\neg X^{\leq}$).

ϵ measure

We introduce cost-type consistency measures $\epsilon^P_{X^{\geq}}(y)$ and $\epsilon^P_{X^{\leq}}(y)$, for $P \subseteq C$, $X^{\geq}$, $X^{\leq} \subseteq U$, $y \in U$, that are defined as

$$\epsilon^P_{X^{\geq}}(y) = \frac{|D^+_P(y) \cap \neg X^{\geq}|}{|\neg X^{\geq}|}, \quad \epsilon^P_{X^{\leq}}(y) = \frac{|D^-_P(y) \cap \neg X^{\leq}|}{|\neg X^{\leq}|}. \tag{3.6}$$

Consistency measure $\epsilon^P_{X^{\geq}}(y)$ (or $\epsilon^P_{X^{\leq}}(y)$) is defined as a ratio of the number of objects that belong to dominance cone $D^+_P(y)$ ($D^-_P(y)$) and to $\neg X^{\geq}$ ($\neg X^{\leq}$) to the number of objects belonging to $\neg X^{\geq}$ ($\neg X^{\leq}$). It can be interpreted as an estimate of conditional probability $Pr(x \in D^+_P(y))|x \in \neg X^{\geq})$ (or $Pr(x \in D^-_P(y))|x \in \neg X^{\leq})$), that any object $x \in U$ belongs to the considered dominance cone based on y given that it does not belong to the considered union. In other words, this is the number of objects in the dominance cone of object y that do not belong to the considered union of classes, divided by the number of all those objects that do not belong to the considered union of classes. Measures $\epsilon^P_{X^{\geq}}(y)$ and $\epsilon^P_{X^{\leq}}(y)$ can be interpreted as catch-all likelihoods (Fitelson, 2007). For all $x \in U$, probability $Pr(x \in D^+_P(y))|x \in \neg X^{\geq})$ can be rewritten as $\frac{Pr(x \in D^+_P(y) \wedge x \in \neg X^{\geq})}{Pr(x \in \neg X^{\geq})}$. Logically, implication $x \in D^+_P(y) \rightarrow x \in X^{\geq}$ can be rewritten as $\neg(x \in D^+_P(y) \wedge x \in \neg X^{\geq})$. Thus, the intuition of calculating measure $\epsilon^P_{X^{\geq}}(y)$ is that we can see how far the implication, i.e., rule, stating that any x from dominance cone based on y belongs to $X^{\geq}$ is not supported by objects from data table. Analogous interpretation can be formulated for $\epsilon^P_{X^{\leq}}(y)$.

ϵ^* measure

Definition 3.6 can be extended to cost-type consistency measures $\epsilon^{*P}_{X^{\geq}}(y)$ and $\epsilon^{*P}_{X^{\leq}}(y)$, for $P \subseteq C$, $X^{\geq}, X^{\leq} \subseteq U$, $y \in U$, are defined as

$$\epsilon^{*P}_{X^{\geq}}(y) = \max_{X^{\geq}} \epsilon^P_{X^{\geq}}(y), \tag{3.7}$$

$$\epsilon^{*P}_{X^{\leq}}(y) = \max_{X^{\leq}} \epsilon^P_{X^{\leq}}(y). \tag{3.8}$$

Measure $\epsilon^{*P}_{X^{\geq}}(y)$ (or $\epsilon^{*P}_{X^{\leq}}(y)$) is defined as a maximal value of measure $\epsilon^{P}_{X^{\geq}}(y)$ ($\epsilon^{P}_{X^{\leq}}(y)$) over all unions of decision classes which contain considered union $X^{\geq}$ ($X^{\leq}$).

Another type of cost-type consistency measures are $\epsilon'^{P}_{X^{\geq}}(y)$ and $\epsilon'^{P}_{X^{\leq}}(y)$. For $P \subseteq C$, ϵ' measure $X^{\geq}$, $X^{\leq} \subseteq U$, $y \in U$, which are defined as

$$\epsilon'^{P}_{X^{\geq}}(y) = \frac{|D^{+}_{P}(y) \cap \neg X^{\geq}|}{|X^{\geq}|}, \quad \epsilon'^{P}_{X^{\leq}}(y) = \frac{|D^{-}_{P}(y) \cap \neg X^{\leq}|}{|X^{\leq}|}. \tag{3.9}$$

Consistency measure $\epsilon'^{P}_{X^{\geq}}(y)$ (or $\epsilon'^{P}_{X^{\leq}}(y)$) is defined as a ratio of the number of objects that belong both to dominance cone $D^{+}_{P}(y)$ ($D^{-}_{P}(y)$) and $\neg X^{\geq}$ ($\neg X^{\leq}$), to the number of objects belonging to union $X^{\geq}$ ($X^{\leq}$). In other words, this measure divides the number of objects in the dominance cone of object y that do not belong to considered union of classes $X^{\geq}$ (or $X^{\leq}$) by the cardinality of that union of classes. Measure $\epsilon'^{P}_{X^{\geq}_{i}}(y)$ (or $\epsilon'^{P}_{X^{\leq}_{i}}(y)$) has different interpretation from consistency measures $\epsilon^{P}_{X^{\geq}_{i}}(y)$ ($\epsilon^{P}_{X^{\leq}_{i}}(y)$) and $\epsilon^{*P}_{X^{\geq}_{i}}(y)$ ($\epsilon^{*P}_{X^{\leq}_{i}}(y)$). It lacks likelihood explication that is appropriate for the other two measures. However, the intuition associated with implication implication $x \in D^{+}_{P}(y) \rightarrow x \in X^{\geq}$ remains valid. According to the definition, the number of objects in the dominance cone of considered object y that do not belong to the considered union of classes is divided by the cardinality of the considered union of classes. This may result in low values of consistency measure $\epsilon'^{P}_{X^{\geq}_{i}}(y)$ ($\epsilon'^{P}_{X^{\leq}_{i}}(y)$) for those unions of classes $X^{\geq}_{i}$ ($X^{\leq}_{i}$) that have a high cardinality.

We consider, moreover, the following gain-type consistency measures $\overline{\mu}^{P}_{X^{\geq}}(y)$ and $\overline{\mu}$ measure $\overline{\mu}^{P}_{X^{\leq}}(y)$. They are defined for $P \subseteq C$, $X^{\geq}, X^{\leq} \subseteq U$, $y \in U$, as:

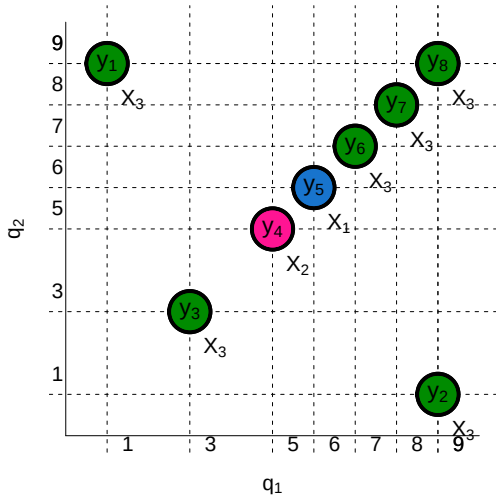
$$\overline{\mu}^{P}_{X^{\geq}}(y) = \max_{\substack{R \subseteq P, \\ z \in D^{-}_{R}(y) \cap X^{\geq}}} \frac{|D^{+}_{R}(z) \cap X^{\geq}|}{|D^{+}_{R}(z)|}, \tag{3.10}$$

$$\overline{\mu}^{P}_{X^{\leq}}(y) = \max_{\substack{R \subseteq P, \\ z \in D^{+}_{R}(y) \cap X^{\leq}}} \frac{|D^{-}_{R}(z) \cap X^{\leq}|}{|D^{-}_{R}(z)|}.$$

Measure $\overline{\mu}^{P}_{X^{\geq}}(y)$ (or $\overline{\mu}^{P}_{X^{\leq}}(y)$) is defined as a maximum rough membership to union $X^{\geq}$ ($X^{\leq}$) over all subsets R of the set of attributes P and over all objects z dominated by y (dominating y) and belonging to $X^{\geq}$ ($X^{\leq}$). Comparing definitions of $\overline{\mu}^{P}_{X^{\geq}}(y)$ (and $\overline{\mu}^{P}_{X^{\leq}}(y)$) with the analogous definition presented for VC-IRSA, one can easily observe that they have a new ingredient - the maximum is calculated not only for all subsets R of P but also for all objects belonging to the intersection of the particular dominance cone of object y and the considered union of decision classes.

Example 3.3.1. *Let us consider the example in Figure 3.5 to show differences between measures that employ μ measure, β precision and ϵ, ϵ' measures. There are eight objects*

from three classes $X_1 = \{y_5\}$, $X_2 = \{y_4\}$ *and* $X_3 = \{y_1, y_2, y_3, y_6, y_7, y_8\}$. *The classes are gain-ordered (i.e.,* X_1 *is the worst class and* X_3 *is the best).*



object	q_1	q_2	decision class
y_1	8	1	X_3
y_2	9	1	X_3
y_3	3	3	X_3
y_4	5	5	X_2
y_5	6	6	X_1
y_6	8	7	X_3
y_7	8	8	X_3
y_8	9	9	X_3

Figure 3.5: Illustration of difference between measures μ, β and ϵ, ϵ' in VC-DRSA.

First, we consider evidence supporting hypothesis that object y_3 *belongs to union* $X_3^{\geq}$. *We get the following values of consistency measures:* $\mu^P_{X_3^{\geq}}(y_3) = \frac{4}{6}$, $\beta^P_{X_3^{\geq}}(y_3) = \frac{1}{3}$ *and* $\epsilon^P_{X_3^{\geq}}(y_3) = \frac{2}{2} = 1$, $\epsilon'^P_{X_3^{\geq}}(y_3) = \frac{2}{6}$. *Value of* μ *is the one that supports the most belonging of object* y_3 *to union* $X_3^{\geq}$. *Value of* ϵ *expresses the highest inconsistency of object* y_3 *belonging to union* $X_3^{\geq}$.

Then, we consider evidence behind object y_4 *belonging to union* $X_2^{\geq}$. *We get the following values:* $\mu^P_{X_2^{\geq}}(y_4) = \frac{4}{5}$, $\beta^P_{X_2^{\geq}}(y_4) = \frac{2}{3}$ *and* $\epsilon^P_{X_2^{\geq}}(y_4) = \frac{1}{1} = 1$, $\epsilon'^P_{X_2^{\geq}}(y_4) = \frac{1}{7}$. *Values of* μ *and* β *support object* y_4 *belonging to union* $X_2^{\geq}$. *Value of* ϵ *expresses the highest inconsistency of object* y_4 *belonging to union* $X_2^{\geq}$.

Finally, we consider evidence behind object y_5 *belonging to union* $X_1^{\leq}$. *We get the following values:* $\mu^P_{X_1^{\leq}}(y_5) = \frac{1}{3}$, $\beta^P_{X_1^{\leq}}(y_5) = \frac{1}{3}$ *and* $\epsilon^P_{X_1^{\leq}}(y_5) = \frac{2}{7}$, $\epsilon'^P_{X_1^{\leq}}(y_5) = \frac{2}{1}$. *Value of* ϵ' *express high inconsistency of object* y_5 *belonging to union* $X_1^{\leq}$.

This example allows for a following explanation of properties of considered measures. Measure μ *for a given union reflects the ration of cardinality of objects from this union to cardinality of all objects in a granule. In this sense this measure is local as it does not take into account objects outside the granule. Precision* β *is also local in this sense, but it express a ratio of cardinality of objects supporting object belonging to the union to cardinality of object that neglect this belonging. On the other hand, measures* ϵ *and* ϵ' *express a global evidence of object belonging to a given union. They relate the cardinality*

of inconsistent objects in a granule to the cardinality of the complement of the union or to the union. Values of ϵ tend to favor small unions while values of ϵ' tend to favor large unions.

3.4 Definition of lower and upper approximations using consistency measures

In Variable Consistency Dominance-based Rough Set Approaches (VC-DRSA), a key point is to find a sufficient evidence for assignment of objects to lower and upper approximations of a particular union of decision classes.

Let $X^{\geq}$ be a set of object belonging to a given upward union of classes and $X^{\leq}$ be a set of objects belonging to downward union of classes. Considering objects from universe U and a subset $P \subseteq C$ of attributes and criteria, from the viewpoint of the evidence of their membership to $X^{\geq}$ and $X^{\leq}$, one can approximate sets $X^{\geq}$ and $X^{\leq}$ by two sets, the *P-lower approximation* and the *P-upper approximation*. The P-lower approximation of $X^{\geq}$ and $X^{\leq}$ is thus composed of objects for which there is enough evidence of membership to the approximated set. The P-upper approximation of $X^{\geq}$ and $X^{\leq}$ is composed of those objects that possibly belong to the approximated set (i.e., includes also those objects for which there is not enough evidence of membership). The *P-boundary* of $X^{\geq}$ and $X^{\leq}$ is the set of objects for which there is not enough evidence of membership to the approximated set.

P-lower approximation

In VC-DRSA, consistency measures (see definition in section 2.3) are used to measure evidence of membership of particular objects to union of classes $X^{\geq}$ or $X^{\leq}$. For $P \subseteq C$, $X^{\geq} \subseteq U$, $X^{\leq} \subseteq U$, $y \in U$, given gain-type consistency measures $f^P_{X^{\geq}}(y)$, $f^P_{X^{\leq}}(y)$ and gain-thresholds $\alpha_{X^{\geq}}$, $\alpha_{X^{\leq}}$, one can consider two variants of the definition of P-lower approximation of union $X^{\geq}$ and $X^{\leq}$:

$$\begin{aligned} \underline{P}^{\alpha_{X^{\geq}}}(X^{\geq}) &= \{y \in U : f^P_{X^{\geq}}(y) \geq \alpha_{X^{\geq}}\}, \\ \underline{P}^{\alpha_{X^{\leq}}}(X^{\leq}) &= \{y \in U : f^P_{X^{\leq}}(y) \geq \alpha_{X^{\leq}}\} \qquad (3.11) \\ \text{or } \underline{P}^{\alpha_{X^{\geq}}}(X^{\geq}) &= \{y \in X^{\geq} : f^P_{X^{\geq}}(y) \geq \alpha_{X^{\geq}}\}, \\ \underline{P}^{\alpha_{X^{\leq}}}(X^{\leq}) &= \{y \in X^{\leq} : f^P_{X^{\leq}}(y) \geq \alpha_{X^{\leq}}\}. \qquad (3.12) \end{aligned}$$

Analogically, given cost-type consistency measures $g^P_{X^{\geq}}(y)$, $g^P_{X^{\leq}}(y)$ and gain-thresholds

$\beta_{X^{\geq}}$, $\beta_{X^{\leq}}$, the two variants are:

$$\begin{aligned} \underline{P}^{\beta_{X^{\geq}}}(X^{\geq}) &= \{y \in U : g^P_{X^{\geq}}(y) \leq \beta_{X^{\geq}}\}, \\ \underline{P}^{\beta_{X^{\leq}}}(X^{\leq}) &= \{y \in U : g^P_{X^{\leq}}(y) \leq \beta_{X^{\leq}}\} \qquad (3.13) \\ \text{or } \underline{P}^{\beta_{X^{\geq}}}(X^{\geq}) &= \{y \in X^{\geq} : g^P_{X^{\geq}}(y) \leq \beta_{X^{\geq}}\}, \\ \underline{P}^{\beta_{X^{\leq}}}(X^{\leq}) &= \{y \in X^{\leq} : g^P_{X^{\leq}}(y) \leq \beta_{X^{\leq}}\}. \qquad (3.14) \end{aligned}$$

In the above definitions, gain-thresholds $\alpha_{X^{\geq}} \in [0, A_X]$, $\alpha_{X^{\leq}} \in [0, A_X]$ and cost-thresholds $\beta_{X^{\geq}} \in [0, B_X]$, $\beta_{X^{\leq}} \in [0, B_X]$. These thresholds are parameters depending on the interpretation of the gain-type or cost-type consistency measure, respectively. They play the role of technical parameters influencing the degree of consistency of objects belonging to lower approximation of $X^{\geq}$ and $X^{\leq}$.

Thus, the values of A_X and B_X also depend on the interpretation of the corresponding consistency measure. For example, in case of probabilistic P-lower approximation defined using the rough membership measure, $A_X = 1$ and values of gain-thresholds $\alpha_{X^{\geq}}, \alpha_{X^{\leq}} \in [0, 1]$ can be calculated using method presented in (Greco et al., 2007; Yao, 2007). This method is based on application of the Bayesian decision procedure in transformation of risk into the value of $\alpha_{X^{\geq}}$ or $\alpha_{X^{\leq}}$.

Let us remark a fundamental difference between definitions (3.11) and (3.12) as well as (3.13) and (3.14). This difference concerns the source of objects considered for inclusion in the P-lower approximation of set $X^{\geq}$ or $X^{\leq}$ either from U or from $X^{\geq}$ or $X^{\leq}$ itself. This feature will be more thoroughly discussed in section 3.5.

The above definitions of P-lower approximations relax the original non-parametric definitions. Precisely, the non-parametric definition for DRSA, and unions of classes $X^{\geq}$, $X^{\leq}$, it is as follows:

$$\begin{aligned} \underline{P}(X^{\geq}) &= \{y \in U : D^+_P(y) \subseteq X^{\geq}\} = \{y \in X^{\geq} : D^+_P(y) \subseteq X^{\geq}\}, \\ \underline{P}(X^{\leq}) &= \{y \in U : D^-_P(y) \subseteq X^{\leq}\} = \{y \in X^{\leq} : D^-_P(y) \subseteq X^{\leq}\}. \end{aligned}$$

An obvious condition of this relaxation is:

$$\underline{P}(X^{\geq}) \subseteq \underline{P}^{\alpha_{X^{\geq}}}(X^{\geq}), \quad \underline{P}(X^{\leq}) \subseteq \underline{P}^{\alpha_{X^{\leq}}}(X^{\leq}), \qquad (3.15)$$

$$\underline{P}(X^{\geq}) \subseteq \underline{P}^{\beta_{X^{\geq}}}(X^{\geq}), \quad \underline{P}(X^{\leq}) \subseteq \underline{P}^{\beta_{X^{\leq}}}(X^{\leq}). \qquad (3.16)$$

The definition of P-upper approximation and of P-boundary of set $X^{\geq}$ and $X^{\leq}$ make use of the complementarity property of rough approximations.

For $P \subseteq C, X^{\geq}, X^{\leq} \subseteq U$, P-upper approximation of sets $X^{\geq}$ and $X^{\leq}$ is defined as

$$\overline{P}^{\alpha\frac{\geq}{X}}(X^{\geq}) = U - \underline{P}^{\alpha\frac{\geq}{X}}(\neg X^{\geq}), \quad \overline{P}^{\alpha\frac{\leq}{X}}(X^{\leq}) = U - \underline{P}^{\alpha\frac{\leq}{X}}(\neg X^{\leq}), \tag{3.17}$$

$$\overline{P}^{\beta\frac{\geq}{X}}(X^{\geq}) = U - \underline{P}^{\beta\frac{\geq}{X}}(\neg X^{\geq}), \quad \overline{P}^{\beta\frac{\leq}{X}}(X^{\leq}) = U - \underline{P}^{\beta\frac{\leq}{X}}(\neg X^{\leq}), \tag{3.18}$$

where $\neg X^{\geq} = U - X^{\geq}$ and $\neg X^{\leq} = U - X^{\leq}$.

P-boundary of $X^{\geq}$ and $X^{\leq}$ is defined as

$$Bn_P^{\alpha\frac{\geq}{X}}(X^{\geq}) = \overline{P}^{\alpha\frac{\geq}{X}}(X^{\geq}) - \underline{P}^{\alpha\frac{\geq}{X}}(X^{\geq}), \quad Bn_P^{\alpha\frac{\leq}{X}}(X^{\leq}) = \overline{P}^{\alpha\frac{\leq}{X}}(X^{\leq}) - \underline{P}^{\alpha\frac{\leq}{X}}(X^{\leq}), \tag{3.19}$$

$$Bn_P^{\beta\frac{\geq}{X}}(X^{\geq}) = \overline{P}^{\beta\frac{\geq}{X}}(X^{\geq}) - \underline{P}^{\beta\frac{\geq}{X}}(X^{\geq}), \quad Bn_P^{\beta\frac{\leq}{X}}(X^{\leq}) = \overline{P}^{\beta\frac{\leq}{X}}(X^{\leq}) - \underline{P}^{\beta\frac{\leq}{X}}(X^{\leq}). \tag{3.20}$$

3.5 Monotonicity of VC-DRSA lower approximations

Lower approximations of unions of classes can be defined in VC-DRSA according to formulas (3.11) and (3.13) or (3.12) and (3.14). To ensure monotonicity of lower approximation it is reasonable to use definitions (3.12) and (3.14). Monotonicity of lower approximation requires that when an object is once included to lower approximation it must remain in it after specific transformations of the data set (i.e., change of set of attributes and/or objects). Let us consider the following example to show why we do not consider definitions (3.11) and (3.13) for monotonic VC-DRSA.

Example 3.5.1. *We have four objects, three of them: y_1, y_2, y_3 belonging to class X_2 and one object y_4 belonging to a worse class X_1. The objects are described by set of criteria $P = \{q_1\}$. As, it is illustrated in Figure 3.6, objects y_1, y_2, y_3 are dominated by object y_4. Let us assume that all object y_1, y_2, y_3 and y_4 have sufficient consistency of belonging to union of classes $X_2^{\geq}$. Thus, all of these objects are included in P-lower approximation of union $X_2^{\geq}$, which is defined according to (3.11) or (3.13). We denote this fact as: $\underline{P}(X_2^{\geq}) = \{y_1, y_2, y_3, y_4\}$.*

Now, let us consider that the set of criteria is extended by criterion q_2, and thus $R = \{q_1, q_2\}$, $R \supset P$. This results in the example illustrated in Figure 3.7.

Objects y_1, y_2, y_3 are incomparable with object y_4 on criteria R (i.e., on q_1 and q_2). Nevertheless, to preserve monotonicity of P-lower approximation of union $X_2^{\geq}$ defined according to (3.11) or (3.13) it would be necessary that object y_4 remains in R-lower approximation of $X_2^{\geq}$. So, it should be $\underline{R}(X_2^{\geq}) = \{y_1, y_2, y_3, y_4\}$.

Note that it is impossible to include object y_4 to a lower approximation defined according to (3.12) or (3.14).

object	q_1	class
y_1	4	X_2
y_2	5	X_2
y_3	7	X_2
y_4	8	X_1

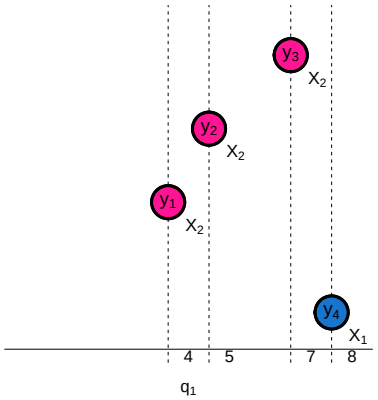


Figure 3.6: Illustration of non-monotonicity of definitions (3.11) and (3.13) on criterion q_1. Exemplary set of objects described by means of set P of one gain-type condition criterion q_1.

object	q_1	q_2	class
y_1	4	4	X_2
y_2	5	6	X_2
y_3	7	8	X_2
y_4	8	1	X_1

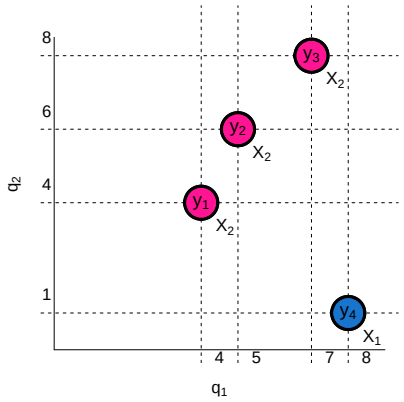


Figure 3.7: Illustration of non-monotonicity of definitions (3.11) and (3.13) on criteria q_1 and q_2. Exemplary set of objects described by means of set R of two gain-type condition criteria q_1 and q_2.

One can observe that properties of rough approximations defined above depend on properties of consistency measures $f_X^P(y)$ and $g_X^P(y)$. Thus, it is possible to formulate some properties with respect to these measures, which ensure desirable properties of rough approximations. To ensure that monotonicity properties of consistency measures are reflected by monotonicity of probabilistic lower approximation it is required to use them in definitions (3.12) and (3.14). In this sense, we postulate to use measures $f_{X^{\geq}}^P(y)$, $f_{X^{\leq}}^P(y)$ and $g_{X^{\geq}}^P(y)$, $g_{X^{\leq}}^P(y)$ that fulfill the following properties of monotonicity (henceforth called *monotonicity properties*).

The following two properties are analogous to properties (m1) and (m2) defined in section 2.5 for VC-IRSA:

(m1) Monotonicity with respect to (w.r.t.) set of attributes $P \subseteq C$. Formally, for all $P \subseteq P' \subseteq C$, $X^{\geq}, X^{\leq} \subseteq U$, $y \in U$, a gain-type measure $f^P_X(y)$ is monotonically non-decreasing w.r.t. P, if and only if (iff)

$$f^P_{X^{\geq}}(y) \leq f^{P'}_{X^{\geq}}(y), \quad f^P_{X^{\leq}}(y) \leq f^{P'}_{X^{\leq}}(y), \tag{3.21}$$

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

$$g^P_{X^{\geq}}(y) \geq g^{P'}_{X^{\geq}}(y), \quad g^P_{X^{\leq}}(y) \geq g^{P'}_{X^{\leq}}(y). \tag{3.22}$$

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

$$f^P_{X^{\geq}}(y) \leq f^P_{X'^{\geq}}(y), \quad f^P_{X^{\leq}}(y) \leq f^P_{X'^{\leq}}(y), \tag{3.23}$$

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

$$g^P_{X^{\geq}}(y) \geq g^P_{X'^{\geq}}(y), \quad g^P_{X^{\leq}}(y) \geq g^P_{X'^{\leq}}(y). \tag{3.24}$$

Moreover, for DRSA, it is reasonable to require that measures $f^P_{X_i^{\geq}}(y)$ (or $f^P_{X_i^{\leq}}(y)$) and $g^P_{X_i^{\geq}}(y)$ (or $g^P_{X_i^{\leq}}(y)$) fulfill the following monotonicity properties:

(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 $f^P_{X_i^{\geq}}(y)$ and $f^P_{X_k^{\leq}}(y)$ are monotonically non-decreasing w.r.t. $X_i^{\geq}$ and $X_k^{\leq}$, respectively, iff

$$f^P_{X_i^{\geq}}(y) \leq f^P_{X_j^{\geq}}(y), \quad f^P_{X_k^{\leq}}(y) \leq f^P_{X_l^{\leq}}(y). \tag{3.25}$$

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

$$g^P_{X_i^{\geq}}(y) \geq g^P_{X_j^{\geq}}(y), \quad g^P_{X_k^{\leq}}(y) \geq g^P_{X_l^{\leq}}(y). \tag{3.26}$$

(m4) Monotonicity w.r.t. P-dominance relation, $P \subseteq C$. 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 $f^P_{X_i^*}(y)$ is monotonically non-decreasing w.r.t. P-dominance relation, iff

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow f^P_{X_i^*}(y_1) \geq f^P_{X_i^*}(y_2), \tag{3.27}$$

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

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow g^P_{X_i^*}(y_1) \leq g^P_{X_i^*}(y_2). \tag{3.28}$$

Monotonicity properties (m1) and (m2) are related to the basic properties of rough sets. Monotonicity properties (m3) and (m4) are specific to DRSA. A rough set approach is called monotonic when the consistency measure used to define its lower approximation fulfills relevant monotonicity properties. For IRSA, relevant properties are (m1) and (m2), while for DRSA, relevant properties are (m1), (m2), (m3) and (m4).

Property (m1) is particularly important. Property (m1) of measures $f^P_{X^{\geq}}(y)$, $f^P_{X^{\leq}}(y)$ and $g^P_{X^{\geq}}(y)$, $g^P_{X^{\leq}}(y)$ ensures monotonicity of P-lower approximation w.r.t. set of attributes $P \subseteq C$, defined according to (3.12) and (3.14), respectively. This property imposes that additional information about objects from U can only give more evidence for the observed assignment of objects to classes. In this case, additional information means, of course, more detailed description of considered objects by an extended set of attributes. Property (m1) is also concordant with the observation that additional attributes can only decrease comparability in the set of objects. When less objects are comparable, then also less inconsistent assignments into classes is observed.

Property (m2) of measures $f^P_{X^{\geq}}(y)$ and $g^P_{X^{\geq}}(y)$ or $f^P_{X^{\leq}}(y)$ and $g^P_{X^{\leq}}(y)$ ensures monotonicity of P-lower approximation w.r.t. set of objects $X^{\geq}, X^{\leq} \subseteq U$. Property (m2) states that when we consider two sets of objects $X'^{\geq} \supset X^{\geq}$ and $X'^{\leq} \supset X^{\leq}$, the evidence for membership to $X'^{\geq}$ and $X'^{\leq}$ for objects from $X^{\geq}$ and $X^{\leq}$should not be worse than the evidence for their membership to $X^{\geq}$ and $X^{\leq}$. In other words, extension of class X_i or union of classes $X_i^{\geq}$ ($X_i^{\leq}$) with new objects, should not negatively affect the evidence for membership of the objects to the extended class or union of classes.

In DRSA, property (m3) of measures $f^P_{X_i^{\geq}}(y)$ (or $f^P_{X_i^{\leq}}(y)$) and $g^P_{X_i^{\geq}}(y)$ (or $g^P_{X_i^{\leq}}(y)$) ensures monotonicity of P-lower approximation w.r.t. union $X_i^{\geq} \subseteq U$ (or $X_i^{\leq} \subseteq U$). This property states that value of a gain-type consistency measure for a union that is a superset should not decrease, while value of a cost-type consistency measure should not increase. For example, for object y which belongs to upward unions $X_i^{\geq}$ and $X_j^{\geq}$, where $X_i^{\geq} \subseteq X_j^{\geq} \subseteq U$, value of gain-type consistency measure $f^P_{X_j^{\geq}}(y)$ should not be worse than the value of this measure calculated for union $X_i^{\geq}$.

The importance of property (m4) in Variable Consistency DRSA (VC-DRSA) was already discussed in (Błaszczyński et al., 2006), however, under the name of *monotonicity of membership to lower approximation.* Monotonicity w.r.t. P-dominance relation,

$P \subseteq C$, is a very desirable property for a measure used in the definition of P-lower approximation of union X_i^*, where $*$ is either $\geq$ or $\leq$. In case of definitions based on formula (3.12), where it is checked if $f^P_{X_i^*}(y) \geq \alpha_{X_i^*}$, a consistency measure defined for $X_i^{\geq}$ should satisfy (3.27), while a consistency measure defined for $X_i^{\leq}$ should satisfy (3.28). For definitions based on formula (3.14), where it is checked if $g^P_{X_i^*}(y) \leq \beta_{X_i^*}$, a consistency measure defined for $X_i^{\geq}$ should satisfy (3.28), while a consistency measure defined for $X_i^{\leq}$ should satisfy (3.27). This ensures continuity of lower approximations - as soon as some object $y \in X_i^{\geq}$ is included into P-lower approximation of union $X_i^{\geq}$, every object $z \in X_i^{\geq}$, which P-dominates y, will also be included into this approximation. Analogically, if some object $y \in X_i^{\leq}$ is included into P-lower approximation of union $X_i^{\leq}$, then every object $z \in X_i^{\leq}$, which is P-dominated by y, will also belong to the considered approximation.

3.5.1 Consistency measure μ

According to (3.1), gain-type rough membership measures are defined for $P \subseteq C$, $X^{\geq}, X^{\leq} \subseteq U$, $y \in U$, as

$$\mu^P_{X^{\geq}}(y) = \frac{\left|D^+_P(y) \cap X^{\geq}\right|}{\left|D^+_P(y)\right|}, \quad \mu^P_{X^{\leq}}(y) = \frac{\left|D^-_P(y) \cap X^{\leq}\right|}{\left|D^-_P(y)\right|},$$

Rough membership μ used within VC-DRSA in definition (3.12). It is expected to have properties (m1), (m2), (m3) and (m4). It can be shown that measure $\mu^P_{X^{\geq}}(y)$ (and $\mu^P_{X^{\leq}}(y)$) has properties (m2) and (m3). Unfortunately, measure $\mu^P_{X^{\geq}}(y)$ (or $\mu^P_{X^{\leq}}(y)$) does not have property (m1) nor (m4).

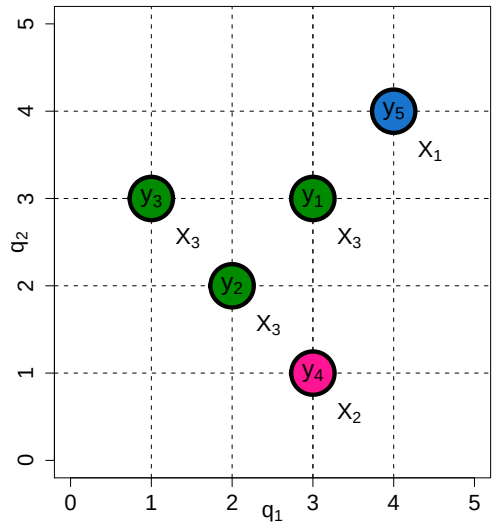
Theorem 3.5.1. *Measures $\mu^P_{X_i^{\geq}}(y)$ and $\mu^P_{X_i^{\leq}}(y)$ do not have property (m1), i.e., for all $P \subseteq P' \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$, $\mu^P_{X_i^{\geq}}(y)$ and $\mu^P_{X_i^{\leq}}(y)$ are not (m1) monotonically non-decreasing w.r.t. sets of attributes P and P'.*

Proof. 3.5.1. The proof will be presented by study of situation presented in Figure 3.8. Measures $\mu^P_{X_i^{\geq}}(y)$ and $\mu^P_{X_i^{\leq}}(y)$ have property (m1) iff for all $P \subseteq P' \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$:

$$\mu^P_{X_i^{\geq}}(y) \leq \mu^{P'}_{X_i^{\geq}}(y), \quad \mu^P_{X_i^{\leq}}(y) \leq \mu^{P'}_{X_i^{\leq}}(y).$$

We can notice that $\mu^{\{q_2\}}_{X_3^{\geq}}(y_2) = \frac{3}{4}$, while $\mu^{\{q_1,q_2\}}_{X_3^{\geq}}(y_2) = \frac{2}{3}$. □

Theorem 3.5.2. *Measures $\mu^P_{X_i^{\geq}}(y)$ and $\mu^P_{X_i^{\leq}}(y)$ do not have property (m4), i.e., for all $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$, measures $\mu^P_{X_i^{\geq}}(y)$ and $\mu^P_{X_i^{\leq}}(y)$ are not (m4) monotonically non-decreasing w.r.t. P-dominance relation.*



object	q_1	q_2	decision class
y_1	3	3	X_3
y_2	2	2	X_3
y_3	1	3	X_3
y_4	3	1	X_2
y_5	4	4	X_1

Figure 3.8: Illustration of measure μ not having property (m4). Exemplary set of objects described by means of set P of gain-type criteria q_1 and q_2.

Proof. 3.5.2. Measures $\mu^P_{X_i^{\geq}}(y)$ and $\mu^P_{X_i^{\leq}}(y)$ have property (m4) iff for all $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U,\ y \in U$:

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \mu^P_{X_i^{\geq}}(y_1) \geq \mu^P_{X_i^{\geq}}(y_2),$$
$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \mu^P_{X_i^{\leq}}(y_1) \leq \mu^P_{X_i^{\leq}}(y_2).$$

If we analyze situation presented in Figure 3.8, we can notice $\mu^P_{X_3^{\geq}}(y_1) = \frac{1}{2}$ and $\mu^P_{X_3^{\geq}}(y_2) = \frac{2}{3}$. □

However, as it is shown in (Błaszczyński et al., 2006), the lack of property (m4) can be overcome by modification of definition of measure μ resulting in measure μ'.

3.5.2 Consistency measure μ'

According to (3.2), formulation of the gain-type Consistency measure μ' for $P \subseteq C$, $X^{\geq}, X^{\leq} \subseteq U$, $y \in U$, is as follows:

$$\mu'^P_{X^{\geq}}(y) = \max_{x \in D_P^-(y)} \frac{\left|D_P^+(x) \cap X^{\geq}\right|}{\left|D_P^+(x)\right|}, \quad \mu'^P_{X^{\leq}}(y) = \max_{x \in D_P^+(y)} \frac{\left|D_P^-(x) \cap X^{\leq}\right|}{\left|D_P^-(x)\right|}. \tag{3.29}$$

Consistency measure μ' used within VC-DRSA in definition (3.12). It is expected to have properties (m1), (m2), (m3) and (m4). As it was noted before, this measure is proposed as modification of rough membership measure μ having property (m4).

It can be shown that measure $\mu'^{P}_{X^{\geq}}(y)$ (and $\mu'^{P}_{X^{\leq}}(y)$)) has properties (m2) and (m3). Unfortunately, this measure does not have property (m1). It can be useful in some applications not requiring property (m1).

Theorem 3.5.3. *Measures $\mu'^{P}_{X_i^{\geq}}(y)$ and $\mu'^{P}_{X_i^{\leq}}(y)$ have property (m4), i.e., for all $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$:*

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \mu'^{P}_{X_i^{\geq}}(y_1) \geq \mu'^{P}_{X_i^{\geq}}(y_2),$$
$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \mu'^{P}_{X_i^{\leq}}(y_1) \leq \mu'^{P}_{X_i^{\leq}}(y_2).$$

Proof. 3.5.3. It results directly from definition of measure μ'. □

Monotonic, in sense of having properties (m2), (m3) and (m4), P-lower approximation of union of classes $X_i^{\geq}$, $X_i^{\leq}$ defined according to (2.7) takes the form:

$$\underline{P}^{\alpha'_{X_i^{\geq}}}(X_i^{\geq}) = \{y \in X_i^{\geq} : \mu'^{P}_{X_i^{\geq}}(y) \geq \alpha'_{X_i^{\geq}}\}, \tag{3.30}$$

$$\underline{P}^{\alpha'_{X_i^{\leq}}}(X_i^{\leq}) = \{y \in X_i^{\leq} : \mu'^{P}_{X_i^{\leq}}(y) \geq \alpha'_{X_i^{\leq}}\}, \tag{3.31}$$

where gain-thresholds $\alpha'_{X_i^{\geq}}, \alpha'_{X_i^{\leq}} \in [0,1]$ reflects the lowest degree of consistency acceptable to include object y in the P-lower approximation of union of classes $X_i^{\geq}$, $X_i^{\leq}$, respectively.

Theorem 3.5.4. *Lower approximations defined according to (3.30) and (3.31) satisfy condition (3.15):*

$$\underline{P}(X_i^{\geq}) \subseteq \underline{P}^{\alpha'_{X_i^{\geq}}}(X_i^{\geq}),$$
$$\underline{P}(X_i^{\leq}) \subseteq \underline{P}^{\alpha'_{X_i^{\leq}}}(X_i^{\leq}).$$

Proof. 3.5.4. For each object $y \in X_i^{\geq}$, $D_P^+(y) \subseteq X_i^{\geq}$ iff $\mu'^{P}_{X_i^{\geq}}(y) = 1$. For each object $y \in X_i^{\leq}$, $D_P^-(y) \subseteq X_i^{\leq}$ iff $\mu'^{P}_{X_i^{\leq}}(y) = 1$. □

3.5.3 Bayes Factor

According to (3.3), formulation of the gain-type Bayes factors for $P \subseteq C$, $X^{\geq}, X^{\leq} \subseteq U$, $y \in U$, is as follows:

$$B^P_{X^{\geq}}(y) = \frac{|D_P^+(y) \cap X^{\geq}||\neg X^{\geq}|}{|D_P^+(y) \cap \neg X^{\geq}||X^{\geq}|}, \quad B^P_{X^{\leq}}(y) = \frac{|D_P^-(y) \cap X^{\leq}||\neg X^{\leq}|}{|D_P^-(y) \cap \neg X^{\leq}||X^{\leq}|}.$$

Unfortunately, measure $B^P_{X^{\geq}}(y)$ (or $B^P_{X^{\leq}}(y)$) has none of the considered monotonicity properties.

Theorem 3.5.5. *Measures $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ do not have property (m1), i.e., for all $P \subseteq P' \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$, $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ are not (m1) monotonically non-decreasing w.r.t. sets of attributes P and P'.*

Proof. 3.5.5. Measures $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ have property (m1) iff for all $P \subseteq P' \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U,\ y \in U$:

$$B^P_{X_i^{\geq}}(y) \leq B^{P'}_{X_i^{\geq}}(y), \quad B^P_{X_i^{\leq}}(y) \leq B^{P'}_{X_i^{\leq}}(y).$$

If we analyze situation presented in Figure 3.8, we can notice that$B^{\{a_2\}}_{X_3^{\geq}}(y_2) = 2$, while $B^{\{a_1,a_2\}}_{X_3^{\geq}}(y_2) = \frac{4}{3}$. □

Theorem 3.5.6. *Measures $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ do not have property (m2), i.e., for all $P \subseteq C$, $X_i^{\geq} \subseteq U$, $X_i^{'\geq} = X_i^{\geq} \cup X_i^{\Delta\geq}$, $U' = U \cup X_i^{\Delta\geq}$, $X_i^{\Delta\geq} \cap U = \emptyset$, $X_i^{\leq} \subseteq U$, $X_i^{'\leq} = X_i^{\leq} \cup X_i^{\Delta\leq}$, $U' = U \cup X_i^{\Delta\leq}$, $X_i^{\Delta\leq} \cap U = \emptyset$, $y \in U$, they are not (m2) monotonically non-decreasing w.r.t sets of objects $X^{\geq}$ and $X^{\leq}$ when they are augmented by new objects.*

Proof. 3.5.6. Measures $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ have property (m2) iff for all $P \subseteq C$, $X_i^{\geq} \subseteq U$, $X_i^{'\geq} = X_i^{\geq} \cup X_i^{\Delta\geq}$, $U' = U \cup X_i^{\Delta\geq}$, $X_i^{\Delta\geq} \cap U = \emptyset$, $X_i^{\leq} \subseteq U$, $X_i^{'\leq} = X_i^{\leq} \cup X_i^{\Delta\leq}$, $U' = U \cup X_i^{\Delta\leq}$, $X_i^{\Delta\leq} \cap U = \emptyset$, $y \in U$:

$$B^P_{X_i^{\geq}}(y) \leq B^P_{X_i^{'\geq}}(y),$$
$$B^P_{X_i^{\leq}}(y) \leq B^P_{X_i^{'\leq}}(y).$$

Let us consider situation presented in Figure 3.8. In order to show that measure $B^P_{X_i^{\geq}}(y)$ does not have property (m2), let us assume that object y_3 is not originally present in the considered data set and is added as a new object. We can observe that $B^P_{X_3^{\geq}}(y_2) = 2 > B^P_{X_3^{'\geq}}(y_2) = \frac{4}{3}$, for $X_3^{\geq} = \{y_1, y_2\}$ and $X_3^{'\geq} = \{y_1, y_2, y_3\}$ □

Theorem 3.5.7. *Measures $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ do not have property (m3), i.e., for all $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$, $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ are not (m3) monotonically non-decreasing w.r.t. unions $X_i^{\geq} \subseteq U$ and $X_i^{\leq} \subseteq U$.*

Proof. 3.5.7. Measures $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ have property (m3) iff for all $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U,\ y \in U$, $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ are not (m3) monotonically non-decreasing

w.r.t. unions $X_i^{\geq} \subseteq U$ and $X_i^{\leq} \subseteq U$:

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow B^P_{X_i^{\geq}}(y_1) \geq B^P_{X_i^{\geq}}(y_2),$$
$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow B^P_{X_i^{\leq}}(y_1) \leq B^P_{X_i^{\leq}}(y_2).$$

Let us now calculate Bayes factors for situation in Figure 3.8. Let us consider object y_2 and unions of classes $X_2^{\geq}$, $X_3^{\geq}$. We have $B^P_{X_3^{\geq}}(y_2) = \frac{4}{3} > B^P_{X_2^{\geq}}(y_2) = \frac{1}{2}$. □

Theorem 3.5.8. *Measures $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ do not have property (m4), i.e., for all $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$, measures $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ are not (m4) monotonically non-decreasing w.r.t. P-dominance relation.*

Proof. 3.5.8. Measures $B^P_{X_i^{\geq}}(y)$ and $B^P_{X_i^{\leq}}(y)$ have property (m4) iff for all $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U,\ y \in U$:

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow B^P_{X_i^{\geq}}(y_1) \geq B^P_{X_i^{\geq}}(y_2),$$
$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow B^P_{X_i^{\leq}}(y_1) \leq B^P_{X_i^{\leq}}(y_2).$$

We analyze situation presented in Figure 3.8, let us notice that $B^P_{X_3^{\geq}}(y_2) = \frac{4}{3} > B^P_{X_3^{\geq}}(y_1) = \frac{2}{3}$.
□

3.5.4 β precision measure

According to (3.4), gain-type consistency measures called β precisions were introduced for Variable-Precision Dominance-based Rough Set Analysis (VP-DRSA) in (Inuiguchi and Yoshioka, 2006). These measures are defined for $P \subseteq C$, $X^{\geq}, X^{\leq} \subseteq U$, $y \in U$, as follows:

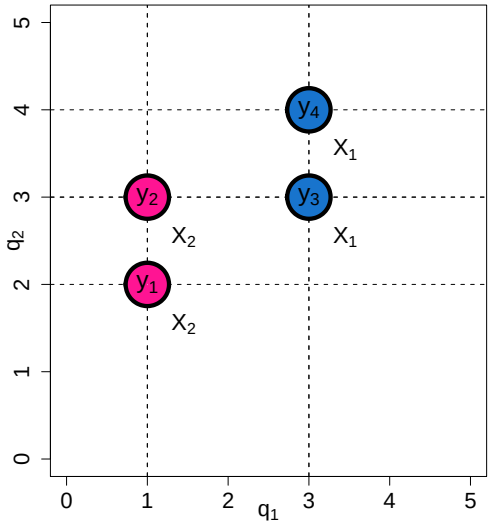
$$\beta^P_{X^{\geq}}(y) = \frac{|D_P^-(y) \cap X^{\geq}|}{|D_P^-(y) \cap X^{\geq}| + |D_P^+(y) \cap \neg X^{\geq}|},$$
$$\beta^P_{X^{\leq}}(y) = \frac{|D_P^+(y) \cap X^{\leq}|}{|D_P^+(y) \cap X^{\leq}| + |D_P^-(y) \cap \neg X^{\leq}|}.$$

β precisions are used within VC-DRSA in definition (3.12). It does not have property (m1), while it appears to have properties (m2), (m3) and (m4).

Theorem 3.5.9. *Measures $\beta^P_{X_i^{\geq}}(y)$ and $\beta^P_{X_i^{\leq}}(y)$ do not have property (m1), i.e., for all $P \subseteq P' \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$, $\beta^P_{X_i^{\geq}}(y)$ and $\beta^P_{X_i^{\leq}}(y)$ are not (m1) monotonically non-decreasing w.r.t. sets of attributes P and P'.*

Proof. 3.5.9. The proof will be presented by study of situation presented in Figure 3.9. Measures $\beta^P_{X_i^{\geq}}(y)$ and $\beta^P_{X_i^{\leq}}(y)$ have property (m1) iff for all $P \subseteq P' \subseteq C,\ X_i^{\geq}, X_i^{\leq} \subseteq U,$ $y \in U$:

$$\beta^P_{X_i^{\geq}}(y) \leq \beta^{P'}_{X_i^{\geq}}(y), \quad \beta^P_{X_i^{\leq}}(y) \leq \beta^{P'}_{X_i^{\leq}}(y).$$



object	q_1	q_2	decision class
y_1	1	2	X_2
y_2	1	3	X_2
y_3	3	3	X_1
y_4	3	4	X_1

Figure 3.9: Illustration of measure β not having property (m1). Exemplary set of objects described by means of set P of gain-type criteria q_1 and q_2.

We can notice that $\beta^{\{q_1\}}_{X_2^{\geq}}(y_1) = \frac{1}{2}$, while $\beta^{\{q_1,q_2\}}_{X_2^{\geq}}(y_1) = \frac{1}{3}$. □

3.5.5 Consistency measure ϵ

According to (3.6), cost-type consistency measures $\epsilon^P_{X_i^{\geq}}(y)$ and $\epsilon^P_{X_i^{\leq}}(y)$, for $P \subseteq C$, $X_i^{\geq}$, $X_i^{\leq}$, $X_{i-1}^{\leq}$, $X_{i+1}^{\geq} \subseteq U$, $y \in U$, are defined as

$$\epsilon^P_{X_i^{\geq}}(y) = \frac{|D_P^+(y) \cap X_{i-1}^{\leq}|}{|X_{i-1}^{\leq}|}, \quad \epsilon^P_{X_i^{\leq}}(y) = \frac{|D_P^-(y) \cap X_{i+1}^{\geq}|}{|X_{i+1}^{\geq}|}.$$

Theorem 3.5.10. *Measures $\epsilon^P_{X_i^{\geq}}(y)$ and $\epsilon^P_{X_i^{\leq}}(y)$ have property (m1), i.e., for all $P \subseteq P' \subseteq C,\ X_i^{\geq}, X_i^{\leq} \subseteq U,\ y \in U$:*

$$\epsilon^P_{X_i^{\geq}}(y) \geq \epsilon^{P'}_{X_i^{\geq}}(y), \quad \epsilon^P_{X_i^{\leq}}(y) \geq \epsilon^{P'}_{X_i^{\leq}}(y).$$

Proof. 3.5.10. From the definition of dominance cones $D_P^+(y)$ and $D_{P'}^+(y)$, $P \subseteq P' \subseteq C$, $y \in U$,

$$D_P^+(y) \supseteq D_{P'}^+(y)$$

for $X_i^{\geq}, X_{i-1}^{\leq} \subseteq U$ being both independent of sets of considered attributes P and P'. This implies:

$$\frac{|D_P^+(y) \cap X_{i-1}^{\leq}|}{|X_{i-1}^{\leq}|} \geq \frac{|D_{P'}^+(y) \cap X_{i-1}^{\leq}|}{|X_{i-1}^{\leq}|} \Leftrightarrow \epsilon^P_{X_i^{\geq}}(y) \geq \epsilon^{P'}_{X_i^{\geq}}(y).$$

The proof for downward union $X_i^{\leq}$ is analogical, but starts from the observation that for negative dominance cones $D_P^-(y)$ and $D_{P'}^-(y)$, $P \subseteq P' \subseteq C$, $y \in U$,

$$D_P^-(y) \supseteq D_{P'}^-(y).$$

□

Theorem 3.5.11. *Measure $\epsilon^P_{X_i^{\geq}}(y)$ has property (m2), i.e., for all $P \subseteq C$, $X_i^{\geq} \subseteq U$, $X_i^{'\geq} = X_i^{\geq} \cup X_i^{\Delta\geq}$, $U' = U \cup X_i^{\Delta\geq}$, $X_i^{\Delta\geq} \cap U = \emptyset$, $y \in U$:*

$$\epsilon^P_{X_i^{\geq}}(y) = \epsilon^P_{X_i^{'\geq}}(y).$$

Proof. 3.5.11. New objects are introduced to union of classes $X_i^{\geq} \subseteq U$. Thus, for all sets of objects $X_i^{\geq} \subseteq U$, $X_i^{'\geq} \subseteq U'$, where $X_i^{'\geq} = X_i^{\geq} \cup X_i^{\Delta\geq}$, $U' = U \cup X_i^{\Delta\geq}$, $X_i^{\Delta\geq} \cap U = \emptyset$,

$$X_{i-1}^{\leq} = X_{i-1}^{'\leq}.$$

For all $P \subseteq C$, $y \in U$, this implies:

$$\frac{|D_P^+(y) \cap X_{i-1}^{\leq}|}{|X_{i-1}^{\leq}|} = \frac{|D_P^{'+}(y) \cap X_{i-1}^{'\leq}|}{|X_{i-1}^{'\leq}|} \Leftrightarrow \epsilon^P_{X_i^{\geq}}(y) = \epsilon^P_{X_i^{'\geq}}(y),$$

where $D_P^{'+}(y)$ denotes P-positive dominance cone of object y when considering universe U'. □

Theorem 3.5.12. *Measure $\epsilon^P_{X_i^{\leq}}(y)$ has property (m2), i.e., for all $P \subseteq C$, $X_i^{\leq} \subseteq U$, $X_i^{'\leq} = X_i^{\leq} \cup X_i^{\Delta\leq}$, $U' = U \cup X_i^{\Delta\leq}$, $X_i^{\Delta\leq} \cap U = \emptyset$, $y \in U$:*

$$\epsilon^P_{X_i^{\leq}}(y) = \epsilon^P_{X_i^{'\leq}}(y).$$

Proof. 3.5.12. Analogous to proof 3.5.11 which is carried out for sets of objects $X_i^{\geq}$ and $X_i^{'\geq}$. In this case, sets of objects $X_{i+1}^{\geq}$ and $X_{i+1}^{'\geq}$ are considered instead of sets $X_{i-1}^{\leq}$ and $X_{i-1}^{'\leq}$, respectively. □

Theorem 3.5.13. *Measure $\epsilon^P_{X_i^{\geq}}(y)$ and $\epsilon^P_{X_i^{\leq}}(y)$ do not have property (m3), i.e., for all $P \subseteq C$, $X_i^{\geq} \subseteq X_j^{\geq} \subseteq U$, $j \leq i$, $y \in U$, measure $\epsilon^P_{X_i^{\geq}}(y)$ is not monotonically non-increasing w.r.t. sets of objects $X_i^{\geq}$, and for all $P \subseteq C$, $X_i^{\leq} \subseteq X_j^{\leq} \subseteq U$, $j \geq i$, $y \in U$, measure $\epsilon^P_{X_i^{\leq}}(y)$ is not monotonically non-increasing w.r.t. sets of objects $X_i^{\leq}$.*

Proof. 3.5.13. The lack of property (m3) of $\epsilon^P_{X_i^{\geq}}(y)$ and $\epsilon^P_{X_i^{\leq}}(y)$ can be illustrated by the following example. We have $P = \{a_1\}$, $X_1 = \{y_1\}$, $X_2 = \{y_2\}$, $X_3 = \{y_3\}$, where $f(y_1, a_1) = 3$, $f(y_2, a_1) = 1$, $f(y_3, a_1) = 2$. Moreover, let us assume that attribute a_1 is gain-type and decision classes are ordered such that class X_3 is better than X_2, which is better than X_1. We have, $\epsilon^P_{X_3^{\geq}}(y_3) = \frac{1}{2} < \epsilon^P_{X_2^{\geq}}(y_3) = 1$. The same can be shown for downward unions. □

In order to ensure property (m3), we introduce two possible modifications of measures $\epsilon^P_{X_i^{\geq}}(y)$ and $\epsilon^P_{X_i^{\leq}}(y)$.

Theorem 3.5.14. *Measures $\epsilon^P_{X_i^{\geq}}(y)$ and $\epsilon^P_{X_i^{\leq}}(y)$ have property (m4), i.e., for all $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$:*

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \epsilon^P_{X_i^{\geq}}(y_1) \leq \epsilon^P_{X_i^{\geq}}(y_2),$$
$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \epsilon^P_{X_i^{\leq}}(y_1) \geq \epsilon^P_{X_i^{\leq}}(y_2).$$

Proof. 3.5.14. Let us consider $y_1, y_2 \in U$ such that $y_1 D_P y_2$, $P \subseteq C$. From the definition of dominance cone $D_P^+(y)$, $y \in U$,

$$D_P^+(y_1) \subseteq D_P^+(y_2).$$

For $X_i^{\geq}, X_{i-1}^{\leq} \subseteq U$, this implies:

$$D_P^+(y_1) \cap X_{i-1}^{\leq} \subseteq D_P^+(y_2) \cap X_{i-1}^{\leq} \Rightarrow |D_P^+(y_1) \cap X_{i-1}^{\leq}| \leq |D_P^+(y_2) \cap X_{i-1}^{\leq}| \Rightarrow$$
$$\Rightarrow \frac{|D_P^+(y_1) \cap X_{i-1}^{\leq}|}{|X_{i-1}^{\leq}|} \leq \frac{|D_P^+(y_2) \cap X_{i-1}^{\leq}|}{|X_{i-1}^{\leq}|} \Leftrightarrow \epsilon^P_{X_i^{\geq}}(y_1) \leq \epsilon^P_{X_i^{\geq}}(y_2).$$

The proof for downward union $X_i^{\leq}$ is analogical, but starts from the observation that for negative dominance cone $D_P^-(y)$, $y \in U$,

$$D_P^-(y_1) \supseteq D_P^-(y_2).$$

□

3.5.6 Consistency measure ϵ^*

According to (3.7), cost-type consistency measures $\epsilon^{*P}_{X_i^{\geq}}(y)$ and $\epsilon^{*P}_{X_i^{\leq}}(y)$, for $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$, are defined as

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

Theorem 3.5.15. *Measures $\epsilon^{*P}_{X_i^{\geq}}(y)$ and $\epsilon^{*P}_{X_i^{\leq}}(y)$ have property (m1), i.e., for all $P \subseteq P' \subseteq C,\ X_i^{\geq}, X_i^{\leq} \subseteq U,\ y \in U$:*

$$\epsilon^{*P}_{X_i^{\geq}}(y) \geq \epsilon^{*P'}_{X_i^{\geq}}(y), \quad \epsilon^{*P}_{X_i^{\leq}}(y) \geq \epsilon^{*P'}_{X_i^{\leq}}(y).$$

Proof. 3.5.15. As it was already proved in proof 3.5.10, for all $P \subseteq P' \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$,

$$\epsilon^{P}_{X_i^{\geq}}(y) \geq \epsilon^{P'}_{X_i^{\geq}}(y)$$

and

$$\epsilon^{P}_{X_i^{\leq}}(y) \geq \epsilon^{P'}_{X_i^{\leq}}(y).$$

Thus,

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

and

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

□

Theorem 3.5.16. *Measure $\epsilon^{*P}_{X_i^{\geq}}(y)$ has property (m2), i.e., for all $P \subseteq C$, $X_i^{\geq} \subseteq U$, $X_i^{'\geq} = X_i^{\geq} \cup X_i^{\Delta\geq}$, $U' = U \cup X_i^{\Delta\geq}$, $X_i^{\Delta\geq} \cap U = \emptyset$, $y \in U$:*

$$\epsilon^{*P}_{X_i^{\geq}}(y) = \epsilon^{*P}_{X_i^{'\geq}}(y).$$

Proof. 3.5.16. New objects are introduced to union of classes $X_i^{\geq} \subseteq U$. Thus, for all sets of objects $X_i^{\geq} \subseteq U$, $X_i^{'\geq} \subseteq U'$, where $X_i^{'\geq} = X_i^{\geq} \cup X_i^{\Delta\geq}$, $U' = U \cup X_i^{\Delta\geq}$, $X_i^{\Delta\geq} \cap U = \emptyset$,

$$X_{i-1}^{\leq} = X_{i-1}^{'\leq}.$$

For all $P \subseteq C$, $y \in U$, this implies:

$$\epsilon^{*P}_{X_i^{\geq}}(y) = \max_{j \leq i} \frac{|D_P^+(y) \cap X_{j-1}^{\leq}|}{|X_{j-1}^{\leq}|} = \max_{j \leq i} \frac{|D_P^{'+}(y) \cap X_{j-1}^{'\leq}|}{|X_{j-1}^{'\leq}|} = \epsilon^{*P}_{X_i^{'\geq}}(y),$$

where $D_P^{'+}(y)$ denotes P-positive dominance cone of object y when considering universe U'. □

Theorem 3.5.17. *Measure $\epsilon^{*P}_{X_i^{\leq}}(y)$ has property (m2), i.e., for all $P \subseteq C$, $X_i^{\leq} \subseteq U$, $X_i^{'\leq} = X_i^{\leq} \cup X_i^{\Delta\leq}$, $U' = U \cup X_i^{\Delta\leq}$, $X_i^{\Delta\leq} \cap U = \emptyset$, $y \in U$:*

$$\epsilon^{*P}_{X_i^{\leq}}(y) = \epsilon^{*P}_{X_i^{'\leq}}(y).$$

Proof. 3.5.17. Analogous to proof 3.5.16 which is carried out for sets of objects $X_i^{\geq}$ and $X_i'^{\geq}$. In this case, sets of objects $X_{i+1}^{\geq}$ and $X_{i+1}'^{\geq}$ are considered instead of sets $X_{i-1}^{\leq}$ and $X_{i-1}'^{\leq}$, respectively. □

Theorem 3.5.18. *Measure $\epsilon^{*P}_{X_i^{\geq}}(y)$ has property (m3), i.e., for all $P \subseteq C$, $X_i^{\geq} \subseteq X_j^{\geq} \subseteq U$, $j \leq i$, $y \in U$:*

$$\epsilon^{*P}_{X_i^{\geq}}(y) \geq \epsilon^{*P}_{X_j^{\geq}}(y).$$

Proof. 3.5.18. Let us consider $P \subseteq C$, $X_i^{\geq} \subseteq X_j^{\geq} \subseteq U$, $j \leq i$, $y \in U$. Since $j \leq i$,

$$\epsilon^{*P}_{X_i^{\geq}}(y) = \max_{k \leq i} \frac{|D_P^+(y) \cap X_{k-1}^{\leq}|}{|X_{k-1}^{\leq}|} \geq \max_{k \leq j} \frac{|D_P^+(y) \cap X_{k-1}^{\leq}|}{|X_{k-1}^{\leq}|} = \epsilon^{*P}_{X_j^{\geq}}(y).$$

□

Theorem 3.5.19. *Measure $\epsilon^{*P}_{X_i^{\leq}}(y)$ has property (m3), i.e., for all $P \subseteq C$, $X_i^{\leq} \subseteq X_j^{\leq} \subseteq U$, $j \geq i$, $y \in U$:*

$$\epsilon^{*P}_{X_i^{\leq}}(y) \geq \epsilon^{*P}_{X_j^{\leq}}(y).$$

Proof. 3.5.19. Analogous to proof 3.5.18. Let us consider $P \subseteq C$, $X_i^{\leq} \subseteq X_j^{\leq} \subseteq U$, $j \geq i$, $y \in U$. Since $j \geq i$,

$$\epsilon^{*P}_{X_i^{\leq}}(y) = \max_{k \geq i} \frac{|D_P^-(y) \cap X_{k+1}^{\geq}|}{|X_{k+1}^{\geq}|} \geq \max_{k \geq j} \frac{|D_P^-(y) \cap X_{k+1}^{\geq}|}{|X_{k+1}^{\geq}|} = \epsilon^{*P}_{X_j^{\leq}}(y).$$

□

Theorem 3.5.20. *Measures $\epsilon^{*P}_{X_i^{\geq}}(y)$ and $\epsilon^{*P}_{X_i^{\leq}}(y)$ have property (m4), i.e., for all $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$:*

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \epsilon^{*P}_{X_i^{\geq}}(y_1) \leq \epsilon^{*P}_{X_i^{\geq}}(y_2),$$
$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \epsilon^{*P}_{X_i^{\leq}}(y_1) \geq \epsilon^{*P}_{X_i^{\leq}}(y_2).$$

Proof. 3.5.20. Let us consider $y_1, y_2 \in U$ such that $y_1 D_P y_2$, $P \subseteq C$. From the definition of dominance cone $D_P^+(y)$, $y \in U$,

$$D_P^+(y_1) \subseteq D_P^+(y_2).$$

For $X_i^{\geq}, X_{i-1}^{\leq} \subseteq U$, this implies:

$$D_P^+(y_1) \cap X_{i-1}^{\leq} \subseteq D_P^+(y_2) \cap X_{i-1}^{\leq} \Rightarrow \frac{|D_P^+(y_1) \cap X_{i-1}^{\leq}|}{|X_{i-1}^{\leq}|} \leq \frac{|D_P^+(y_2) \cap X_{i-1}^{\leq}|}{|X_{i-1}^{\leq}|} \Rightarrow$$
$$\Rightarrow \max_{k \leq i} \frac{|D_P^+(y_1) \cap X_{k-1}^{\leq}|}{|X_{k-1}^{\leq}|} \leq \max_{k \leq i} \frac{|D_P^+(y_2) \cap X_{k-1}^{\leq}|}{|X_{k-1}^{\leq}|} \Leftrightarrow \epsilon^{*P}_{X_i^{\geq}}(y_1) \leq \epsilon^{*P}_{X_i^{\geq}}(y_2).$$

The proof for downward union $X_i^{\leq}$ is analogical, but starts from the observation that for negative dominance cone $D_P^-(y)$, $y \in U$,

$$D_P^-(y_1) \supseteq D_P^-(y_2). \qquad \square$$

$\square$

Monotonic P-lower approximation of union of classes $X_i^{\geq}$, $X_i^{\leq}$ defined according to (2.9) takes the form:

$$\underline{P}^{\beta^*_{X_i^{\geq}}}(X_i^{\geq}) = \{y \in X_i^{\geq} : \epsilon^{*P}_{X_i^{\geq}}(y) \leq \beta^*_{X_i^{\geq}}\}, \tag{3.32}$$

$$\underline{P}^{\beta^*_{X_i^{\leq}}}(X_i^{\leq}) = \{y \in X_i^{\leq} : \epsilon^{*P}_{X_i^{\leq}}(y) \leq \beta^*_{X_i^{\leq}}\}, \tag{3.33}$$

where cost-threshold $\beta^*_{X_i^{\geq}}, \beta^*_{X_i^{\leq}} \in [0,1]$ reflects the highest degree of consistency acceptable to include object y in the P-lower approximation of union of classes $X_i^{\geq}$, $X_i^{\leq}$, respectively.

Theorem 3.5.21. *Lower approximations defined according to (3.32) and (3.33) satisfy condition (3.16):*

$$\underline{P}(X_i^{\geq}) \subseteq \underline{P}^{\beta^*_{X_i^{\geq}}}(X_i^{\geq}),$$
$$\underline{P}(X_i^{\leq}) \subseteq \underline{P}^{\beta^*_{X_i^{\leq}}}(X_i^{\leq}).$$

Proof. 3.5.21. For each object $y \in X_i^{\geq}$, $D_P^+(y) \subseteq X_i^{\geq}$ iff $\epsilon^{*P}_{X_i^{\geq}}(y) = 0$. For each object $y \in X_i^{\leq}$, $D_P^-(y) \subseteq X_i^{\leq}$ iff $\epsilon^{*P}_{X_i^{\leq}}(y) = 0$. $\square$

3.5.7 Consistency measure ϵ'

Another way to overcome the lack of property (m3) is to consider cost-type consistency measures $\epsilon'^{P}_{X_i^{\geq}}(y)$ and $\epsilon'^{P}_{X_i^{\leq}}(y)$. For $P \subseteq C$, $X_i^{\geq}$, $X_i^{\leq}$, $X_{i-1}^{\leq}$, $X_{i+1}^{\geq} \subseteq U$, $y \in U$, they are defined, according to (3.9), as

$$\epsilon'^{P}_{X_i^{\geq}}(y) = \frac{|D_P^+(y) \cap X_{i-1}^{\leq}|}{|X_i^{\geq}|}, \quad \epsilon'^{P}_{X_i^{\leq}}(y) = \frac{|D_P^-(y) \cap X_{i+1}^{\geq}|}{|X_i^{\leq}|}.$$

Theorem 3.5.22. *Measures $\epsilon'^{P}_{X_i^{\geq}}(y)$ and $\epsilon'^{P}_{X_i^{\leq}}(y)$ have property (m1), i.e., for all $P \subseteq P' \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$:*

$$\epsilon'^{P}_{X_i^{\geq}}(y) \geq \epsilon'^{P'}_{X_i^{\geq}}(y), \quad \epsilon'^{P}_{X_i^{\leq}}(y) \geq \epsilon'^{P'}_{X_i^{\leq}}(y).$$

Proof. 3.5.22. Analogous to proof 3.5.10 for measures $\epsilon^P_{X_i^{\geq}}(y)$ and $\epsilon^P_{X_i^{\leq}}(y)$ - only the common denominators in fractions are changed from $|X_{i-1}^{\leq}|$ and $|X_{i+1}^{\geq}|$ to $|X_i^{\geq}|$ and $|X_i^{\leq}|$, respectively. □

Theorem 3.5.23. *Measure $\epsilon'^P_{X_i^{\geq}}(y)$ has property (m2), i.e., for all $P \subseteq C$, $X_i^{\geq} \subseteq U$, $X_i'^{\geq} = X_i^{\geq} \cup X_i^{\Delta\geq}$, $U' = U \cup X_i^{\Delta\geq}$, $X_i^{\Delta\geq} \cap U = \emptyset$, $y \in U$:*

$$\epsilon'^P_{X_i^{\geq}}(y) > \epsilon'^P_{X_i'^{\geq}}(y).$$

Proof. 3.5.23. New objects are introduced to union of classes $X_i^{\geq} \subseteq U$. Thus, for all sets of objects $X_i^{\geq} \subseteq U$, $X_i'^{\geq} = X_i^{\geq} \cup X_i^{\Delta\geq}$, where $X_i^{\Delta\geq} \cap U = \emptyset$,

$$X_{i-1}^{\leq} = X_{i-1}'^{\leq}, \quad |X_i^{\geq}| < |X_i'^{\geq}|.$$

This implies that for all $P \subseteq C$, $y \in U$:

$$\frac{|D_P^+(y) \cap X_{i-1}^{\leq}|}{|X_i^{\geq}|} > \frac{|D_P'^+(y) \cap X_{i-1}'^{\leq}|}{|X_i'^{\geq}|} \Leftrightarrow \epsilon'^P_{X_i^{\geq}}(y) > \epsilon'^P_{X_i'^{\geq}}(y),$$

where $D_P'^+(y)$ denotes P-positive dominance cone of object y when considering universe $U \cup X_i^{\Delta\geq}$. □

Theorem 3.5.24. *Measure $\epsilon'^P_{X_i^{\leq}}(y)$ has property (m2), i.e., for all $P \subseteq C$, $X_i^{\leq} \subseteq U$, $X_i'^{\leq} = X_i^{\leq} \cup X_i^{\Delta\leq}$, $U' = U \cup X_i^{\Delta\leq}$, $X_i^{\Delta\leq} \cap U = \emptyset$, $y \in U$:*

$$\epsilon'^P_{X_i^{\leq}}(y) > \epsilon'^P_{X_i'^{\leq}}(y).$$

Proof. 3.5.24. Analogous to proof 3.5.23, carried out for sets of objects $X_i^{\geq}$, $X_i'^{\geq}$. Here, sets of objects $X_{i+1}^{\geq}$, $X_{i+1}'^{\geq}$ and cardinalities of sets $|X_i^{\leq}|$, $|X_i'^{\leq}|$ are taken into account instead of sets $X_{i-1}^{\leq}$, $X_{i-1}'^{\leq}$ and cardinalities $|X_i^{\geq}|$, $|X_i'^{\geq}|$, respectively. □

Theorem 3.5.25. *Measure $\epsilon'^P_{X_i^{\geq}}(y)$ has property (m3), i.e., for all $P \subseteq C$, $X_i^{\geq} \subseteq X_j^{\geq} \subseteq U$, $j \leq i$, $y \in U$:*

$$\epsilon'^P_{X_i^{\geq}}(y) \geq \epsilon'^P_{X_j^{\geq}}(y).$$

Proof. 3.5.25. Let us consider $P \subseteq C$, $X_i^{\geq} \subseteq X_j^{\geq} \subseteq U$, $j \leq i$, $y \in U$. Since $j \leq i$,

$$X_i^{\geq} \subseteq X_j^{\geq} \quad \text{and} \quad X_{i-1}^{\leq} \supseteq X_{j-1}^{\leq}.$$

This implies:

$$\frac{|D_P^+(y) \cap X_{i-1}^{\leq}|}{|X_i^{\geq}|} \geq \frac{|D_P^+(y) \cap X_{j-1}^{\leq}|}{|X_j^{\geq}|} \quad \Leftrightarrow \quad \epsilon'^P_{X_i^{\geq}}(y) \geq \epsilon'^P_{X_j^{\geq}}(y).$$

□

Theorem 3.5.26. *Measure* $\epsilon'^{P}_{X_i^{\leq}}(y)$ *has property (m3), i.e., for all* $P \subseteq C$, $X_i^{\leq} \subseteq X_j^{\leq} \subseteq U$, $j \geq i$, $y \in U$*:*

$$\epsilon'^{P}_{X_i^{\leq}}(y) \geq \epsilon'^{P}_{X_j^{\leq}}(y).$$

Proof. 3.5.26. Analogous to proof 3.5.25. Let us consider $P \subseteq C$, $X_i^{\leq} \subseteq X_j^{\leq} \subseteq U$, $j \geq i$, $y \in U$. Since $j \geq i$,

$$X_i^{\leq} \subseteq X_j^{\leq} \quad \text{and} \quad X_{i+1}^{\geq} \supseteq X_{j+1}^{\geq}.$$

This implies:

$$\frac{|D_P^-(y) \cap X_{i+1}^{\geq}|}{|X_i^{\leq}|} \geq \frac{|D_P^-(y) \cap X_{j+1}^{\geq}|}{|X_j^{\leq}|} \quad \Leftrightarrow \quad \epsilon'^{P}_{X_i^{\leq}}(y) \geq \epsilon'^{P}_{X_j^{\leq}}(y).$$

□

Theorem 3.5.27. *Measures* $\epsilon'^{P}_{X_i^{\geq}}(y)$ *and* $\epsilon'^{P}_{X_i^{\leq}}(y)$ *have property (m4), i.e., for all* $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$*:*

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \epsilon'^{P}_{X_i^{\geq}}(y_1) \leq \epsilon'^{P}_{X_i^{\geq}}(y_2),$$
$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \epsilon'^{P}_{X_i^{\leq}}(y_1) \geq \epsilon'^{P}_{X_i^{\leq}}(y_2).$$

Proof. 3.5.27. Analogous to proof 3.5.14 for measures $\epsilon^{P}_{X_i^{\geq}}(y)$ and $\epsilon^{P}_{X_i^{\leq}}(y)$ - only the common denominators in fractions are changed from $|X_{i-1}^{\leq}|$ and $|X_{i+1}^{\geq}|$ to $|X_i^{\geq}|$ and $|X_i^{\leq}|$, respectively. □

Monotonic P-lower approximation of union of classes $X_i^{\geq}$, $X_i^{\leq}$ defined according to (2.9) takes the form:

$$\underline{P}^{\beta'_{X_i^{\geq}}}(X_i^{\geq}) = \{y \in X_i^{\geq} : \epsilon'^{P}_{X_i^{\geq}}(y) \leq \beta'_{X_i^{\geq}}\}, \tag{3.34}$$

$$\underline{P}^{\beta'_{X_i^{\leq}}}(X_i^{\leq}) = \{y \in X_i^{\leq} : \epsilon'^{P}_{X_i^{\leq}}(y) \leq \beta'_{X_i^{\leq}}\}, \tag{3.35}$$

where cost-threshold $\beta'_{X_i^{\geq}} \in \left[0, \frac{|X_{i-1}^{\leq}|}{|X_i^{\geq}|}\right]$, $\beta'_{X_i^{\leq}} \in \left[0, \frac{|X_{i+1}^{\geq}|}{|X_i^{\leq}|}\right]$ reflects the highest degree of consistency acceptable to include object y in the P-lower approximation of union of classes $X_i^{\geq}$, $X_i^{\leq}$, respectively.

Theorem 3.5.28. *Lower approximations defined according to (3.34) and (3.35) satisfy condition (2.11):*

$$\underline{P}(X_i^{\geq}) \subseteq \underline{P}^{\beta'_{X_i^{\geq}}}(X_i^{\geq}),$$
$$\underline{P}(X_i^{\leq}) \subseteq \underline{P}^{\beta'_{X_i^{\leq}}}(X_i^{\leq}).$$

3.5.28. For each object $y \in X_i^{\geq}$, $D_P^+(y) \subseteq X_i^{\geq}$ iff $\epsilon'^{P}_{X_i^{\geq}}(y) = 0$. For each object $y \in X_i^{\leq}$, $D_P^-(y) \subseteq X_i^{\leq}$ iff $\epsilon'^{P}_{X_i^{\leq}}(y) = 0$. □

3.5.8 Consistency measure $\overline{\mu}$

For $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$, we also consider the following gain-type consistency measures, defined according to (3.10), as:

$$\overline{\mu}^P_{X_i^{\geq}}(y) = \max_{\substack{R \subseteq P, \\ z \in D_R^-(y) \cap X_i^{\geq}}} \frac{|D_R^+(z) \cap X_i^{\geq}|}{|D_R^+(z)|},$$

$$\overline{\mu}^P_{X_i^{\leq}}(y) = \max_{\substack{R \subseteq P, \\ z \in D_R^+(y) \cap X_i^{\leq}}} \frac{|D_R^-(z) \cap X_i^{\leq}|}{|D_R^-(z)|}.$$

Theorem 3.5.29. *Measures $\overline{\mu}^P_{X_i^{\geq}}(y)$ and $\overline{\mu}^P_{X_i^{\leq}}(y)$ have property (m1), i.e., for all $P \subseteq P' \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$:*

$$\overline{\mu}^P_{X_i^{\geq}}(y) \leq \overline{\mu}^{P'}_{X_i^{\geq}}(y),$$

$$\overline{\mu}^P_{X_i^{\leq}}(y) \leq \overline{\mu}^{P'}_{X_i^{\leq}}(y).$$

Proof. 3.5.29. For all $P \subseteq P' \subseteq C$, $X_i^{\geq} \subseteq U$, $y \in U$,

$$\overline{\mu}^P_{X_i^{\geq}}(y) = \max_{\substack{R \subseteq P, \\ z \in D_R^-(y) \cap X_i^{\geq}}} \frac{|D_R^+(z) \cap X_i^{\geq}|}{|D_R^+(z)|} \leq \max_{\substack{R \subseteq P', \\ z \in D_R^-(y) \cap X_i^{\geq}}} \frac{|D_R^+(z) \cap X_i^{\geq}|}{|D_R^+(z)|} = \overline{\mu}^{P'}_{X_i^{\geq}}(y).$$

The same can be proved for measure $\overline{\mu}^P_{X_i^{\leq}}(y)$. □

Theorem 3.5.30. *Measure $\overline{\mu}^P_{X_i^{\geq}}(y)$ has property (m2), i.e., for all $P \subseteq C$, $X_i^{\geq} \subseteq U$, $X_i'^{\geq} = X_i^{\geq} \cup X_i^{\Delta \geq}$, $U' = U \cup X_i^{\Delta \geq}$, $X_i^{\Delta \geq} \cap U = \emptyset$, $y \in U$:*

$$\overline{\mu}^P_{X_i^{\geq}}(y) \leq \overline{\mu}^P_{X_i'^{\geq}}(y).$$

Proof. 3.5.30. Analogous to proof 2.5.11. Let us consider $P \subseteq C$, $X_i^{\geq} \subseteq U$, $X_i'^{\geq} = X_i^{\geq} \cup X_i^{\Delta \geq}$, $X_i^{\Delta \geq} \cap U = \emptyset$, $y \in U$. Since all new objects are added to union of classes $X_i^{\geq}$, both numerator and denominator of fraction

$$\frac{\left|D_P^+(y) \cap X_i^{\geq}\right|}{\left|D_P^+(y)\right|} = \mu^P_{X_i^{\geq}}(y)$$

can increase only with the same number $k \geq 0$, equal to difference $|D_P^{'+}(y)| - |D_P^+(y)|$:

$$\frac{\left|D_P^+(y) \cap X_i^{\geq}\right| + k}{\left|D_P^+(y)\right| + k} = \frac{\left|D_P^{'+}(y) \cap X_i^{'\geq}\right|}{\left|D_P^{'+}(y)\right|} = \mu^P_{X_i^{'\geq}}(y),$$

where $D_P^{'+}(y)$ denotes the set of objects dominating object y when considering set of attributes P and universe $U \cup X_i^{\Delta\geq}$. Using the same reasoning as in proof 2.5.11, we can show that

$$\mu^P_{X_i^{\geq}}(y) \leq \mu^P_{X_i^{'\geq}}(y). \tag{3.36}$$

Thus,

$$\overline{\mu}^P_{X_i^{\geq}}(y) = \max_{\substack{R\subseteq P,\\ z\in D_R^-(y)\cap X_i^{\geq}}} \mu^R_{X_i^{\geq}}(z) \leq \max_{\substack{R\subseteq P,\\ z\in D_R^{'-}(y)\cap X_i^{'\geq}}} \mu^R_{X_i^{'\geq}}(z) = \overline{\mu}^P_{X_i^{'\geq}}(y).$$

□

Theorem 3.5.31. *Measure $\overline{\mu}^P_{X_i^{\leq}}(y)$ has property (m2), i.e., for all $P \subseteq C$, $X_i^{\leq} \subseteq U$, $X_i^{'\leq} = X_i^{\leq} \cup X_i^{\Delta\leq}$, $U' = U \cup X_i^{\Delta\leq}$, $X_i^{\Delta\leq} \cap U = \emptyset$, $y \in U$:*

$$\overline{\mu}^P_{X_i^{\leq}}(y) \leq \overline{\mu}^P_{X_i^{'\leq}}(y).$$

Proof. 3.5.31. Analogous to proof 3.5.30 - only the upward unions are changed to downward unions and positive dominance cones are changed to negative dominance cones, respectively. □

Theorem 3.5.32. *Measure $\overline{\mu}^P_{X_i^{\geq}}(y)$ has property (m3), i.e., for all $P \subseteq C$, $X_i^{\geq} \subseteq X_j^{\geq} \subseteq U$, $j \leq i$, $y \in U$:*

$$\overline{\mu}^P_{X_i^{\geq}}(y) \leq \overline{\mu}^P_{X_j^{\geq}}(y).$$

Proof. 3.5.32. Let us consider $P \subseteq C$, $X_i^{\geq} \subseteq X_j^{\geq} \subseteq U$, $j \leq i$, $y \in U$. Since $X_i^{\geq} \subseteq X_j^{\geq}$,

$$\overline{\mu}^P_{X_i^{\geq}}(y) = \max_{\substack{R\subseteq P,\\ z\in D_R^-(y)\cap X_i^{\geq}}} \frac{|D_R^+(z) \cap X_i^{\geq}|}{|D_R^+(z)|} \leq \max_{\substack{R\subseteq P,\\ z\in D_R^-(y)\cap X_j^{\geq}}} \frac{|D_R^+(z) \cap X_j^{\geq}|}{|D_R^+(z)|} = \overline{\mu}^P_{X_j^{\geq}}(y).$$

□

Theorem 3.5.33. *Measure $\overline{\mu}^P_{X_i^{\leq}}(y)$ has property (m3), i.e, for all $P \subseteq C$, $X_i^{\leq} \subseteq X_j^{\leq} \subseteq U$, $j \geq i$, $y \in U$:*

$$\overline{\mu}^P_{X_i^{\leq}}(y) \leq \overline{\mu}^P_{X_j^{\leq}}(y).$$

Proof. 3.5.33. Analogous to proof 3.5.32. Unions of classes $X_i^{\leq} \subseteq X_j^{\leq} \subseteq U$ are considered. □

Theorem 3.5.34. *Measures $\overline{\mu}^P_{X_i^{\geq}}(y)$ and $\overline{\mu}^P_{X_i^{\leq}}(y)$ have property (m4), i.e., for all $P \subseteq C$, $X_i^{\geq}, X_i^{\leq} \subseteq U$, $y \in U$:*

$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \overline{\mu}^P_{X_i^{\geq}}(y_1) \geq \overline{\mu}^P_{X_i^{\geq}}(y_2),$$
$$\forall y_1, y_2 \in U : y_1 D_P y_2 \Rightarrow \overline{\mu}^P_{X_i^{\leq}}(y_1) \leq \overline{\mu}^P_{X_i^{\leq}}(y_2).$$

Proof. 3.5.34. Let us consider $y_1, y_2 \in U$ such that $y_1 D_P y_2$, $P \subseteq C$. From the definitions of dominance cones $D_P^+(y)$ and $D_P^-(y)$, $y \in U$,

$$D_P^+(y_1) \subseteq D_P^+(y_2) \text{ and } D_P^-(y_1) \supseteq D_P^-(y_2).$$

For $X_i^{\geq}, X_i^{\leq} \subseteq U$, this implies:

$$\begin{aligned}
&\forall R \subseteq P : D_R^-(y_1) \supseteq D_R^-(y_2) \Rightarrow \forall R \subseteq P : D_R^-(y_1) \cap X_i^{\geq} \supseteq D_R^-(y_2) \cap X_i^{\geq} && \Rightarrow \\
&\Rightarrow \{(R,z) : R \subseteq P, z \in D_R^-(y_1) \cap X_i^{\geq}\} \supseteq \{(R,z) : R \subseteq P, z \in D_R^-(y_2) \cap X_i^{\geq}\} && \Rightarrow \\
&\Rightarrow \max_{\substack{R \subseteq P, \\ z \in D_R^-(y_1) \cap X_i^{\geq}}} \frac{|D_R^+(z) \cap X_i^{\geq}|}{|D_R^+(z)|} \geq \max_{\substack{R \subseteq P, \\ z \in D_R^-(y_2) \cap X_i^{\geq}}} \frac{|D_R^+(z) \cap X_i^{\geq}|}{|D_R^+(z)|} && \Leftrightarrow \\
&\Leftrightarrow \overline{\mu}^P_{X_i^{\geq}}(y_1) \geq \overline{\mu}^P_{X_i^{\geq}}(y_2),
\end{aligned}$$

$$\begin{aligned}
&\forall R \subseteq P : D_R^+(y_1) \subseteq D_R^+(y_2) \Rightarrow \forall R \subseteq P : D_R^+(y_1) \cap X_i^{\leq} \subseteq D_R^+(y_2) \cap X_i^{\leq} && \Rightarrow \\
&\Rightarrow \{(R,z) : R \subseteq P, z \in D_R^+(y_1) \cap X_i^{\leq}\} \subseteq \{(R,z) : R \subseteq P, z \in D_R^+(y_2) \cap X_i^{\leq}\} && \Rightarrow \\
&\Rightarrow \max_{\substack{R \subseteq P, \\ z \in D_R^+(y_1) \cap X_i^{\leq}}} \frac{|D_R^-(z) \cap X_i^{\leq}|}{|D_R^-(z)|} \leq \max_{\substack{R \subseteq P, \\ z \in D_R^+(y_2) \cap X_i^{\leq}}} \frac{|D_R^-(z) \cap X_i^{\leq}|}{|D_R^-(z)|} && \Leftrightarrow \\
&\Leftrightarrow \overline{\mu}^P_{X_i^{\leq}}(y_1) \leq \overline{\mu}^P_{X_i^{\leq}}(y_2).
\end{aligned}$$

□

Monotonic P-lower approximation of union of classes $X_i^{\geq}$, $X_i^{\leq}$ defined according to (2.7) takes the form:

$$\underline{P}^{\overline{\alpha}_{X_i^{\geq}}}(X_i^{\geq}) = \{y \in X_i^{\geq} : \overline{\mu}^P_{X_i^{\geq}}(y) \geq \overline{\alpha}_{X_i^{\geq}}\}, \tag{3.37}$$

$$\underline{P}^{\overline{\alpha}_{X_i^{\leq}}}(X_i^{\leq}) = \{y \in X_i^{\leq} : \overline{\mu}^P_{X_i^{\leq}}(y) \geq \overline{\alpha}_{X_i^{\leq}}\}, \tag{3.38}$$

where gain-threshold $\overline{\alpha}_{X_i^{\geq}}, \overline{\alpha}_{X_i^{\leq}} \in [0,1]$ reflects the lowest degree of consistency acceptable to include object y in the P-lower approximation of union of classes $X_i^{\geq}$, $X_i^{\leq}$, respectively.

Theorem 3.5.35. *Lower approximations defined according to (3.37) and (3.38) satisfy condition (3.15):*

$$\underline{P}(X_i^{\geq}) \subseteq \underline{P}^{\overline{\alpha}_{X_i^{\geq}}}(X_i^{\geq}),$$
$$\underline{P}(X_i^{\leq}) \subseteq \underline{P}^{\overline{\alpha}_{X_i^{\leq}}}(X_i^{\leq}).$$

Proof. 3.5.35. For each object $y \in X_i^{\geq}$, $D_P^+(y) \subseteq X_i^{\geq}$ iff $\overline{\mu}^P_{X_i^{\geq}}(y) = 1$. For each object $y \in X_i^{\leq}$, $D_P^-(y) \subseteq X_i^{\leq}$ iff $\overline{\mu}^P_{X_i^{\leq}}(y) = 1$. □

3.6 Properties of rough approximations from the viewpoint of rule induction

As we already showed in section 2.6, P-lower approximations defined as (2.7) and (2.9) are not sufficient to define sets of objects covered by rules in VC-IRSA. For this reason, we used the concept of P-positive region of approximated set. This situation also holds for P-lower approximations defined as (3.12) and (3.14) for VC-DRSA. Let us explain this point in detail.

A decision rule that assigns to a given upward union of classes $X_i^{\geq}$, covers object y and objects P-dominating object y, i.e., if it covers object y it also covers all objects from granule $D_P^+(y)$. Analogously, a decision rule that assigns to a given downward union of classes $X_i^{\leq}$, covers object y and objects that are P-dominated by y, i.e., if it covers object y it also covers all objects from granule $D_P^-(y)$. When we create a rule covering object y that belong to P-lower approximation of $X_i^{\geq}$ and $D_P^+(y)$ happens to be composed of objects that do not belong to $X_i^{\geq}$ there may be no possibility to cover y while not covering objects from $D_P^+(y)$ that do not belong to $X_i^{\geq}$. For object y that belong to union of classes $X_i^{\leq}$ and granule $D_p^-(y)$ the situation may be the same. For this reason, we define P-positive, P-negative and P-boundary regions of unions of classes $X_i^{\geq}$ and $X_i^{\leq}$ in P-evaluation space, i.e., in $V_P = \prod_{j:a_j \in P} V_{a_j}$.

P-positive region

For $P \subseteq C, X_i^{\geq}, X_i^{\leq} \subseteq U$, where $y \in U$ and $\alpha_{X_i} \in [0, A_X]$, $\beta_{X_i} \in [0, B_X]$, P-positive regions of a unions of classes $X_i^{\geq}$ and $X_i^{\leq}$ are defined as:

$$POS_P^{\alpha_{X_i^{\geq}}}(X_i^{\geq}) = \bigcup_{y \in \underline{P}^{\alpha_{X_i^{\geq}}}(X_i^{\geq})} D_P^+(y), \quad POS_P^{\alpha_{X_i^{\leq}}}(X_i^{\leq}) = \bigcup_{y \in \underline{P}^{\alpha_{X_i^{\leq}}}(X_i^{\leq})} D_P^-(y), \quad (3.39)$$

$$POS_P^{\beta_{X_i^{\geq}}}(X_i^{\geq}) = \bigcup_{y \in \underline{P}^{\beta_{X_i^{\geq}}}(X_i^{\geq})} D_P^+(y), \quad POS_P^{\beta_{X_i^{\leq}}}(X_i^{\leq}) = \bigcup_{y \in \underline{P}^{\beta_{X_i^{\leq}}}(X_i^{\leq})} D_P^-(y), \quad (3.40)$$

where $\underline{P}^{\alpha_{X_i^{\geq}}}(X_i^{\geq})$ and $\underline{P}^{\alpha_{X_i^{\leq}}}(X_i^{\leq})$ are defined according to (3.12) and $\underline{P}^{\beta_{X_i^{\geq}}}(X_i^{\geq})$ and $\underline{P}^{\beta_{X_i^{\leq}}}(X_i^{\leq})$ are defined according to (3.14). From (3.39 and 3.40), positive regions are composed of all objects y from P-lower approximation of $X_i^{\geq}$ or $X_i^{\leq}$ and objects that belong to dominance cone $D_P^+(y)$ or $D_P^-(y)$ (i.e., all objects from respective dominance cone starting in y). This can be denoted as property of P-positive regions:

$$
\begin{aligned}
&POS_P^{\alpha_{X_i^{\geq}}}(X_i^{\geq}) = \\
&= \left\{y \in X_i^{\geq} : f^P_{X_i^{\geq}}(y) \geq \alpha_{X_i^{\geq}}\right\} \cup \left\{y \in D_P^+(x) : x \in \underline{P}^{\alpha_{X_i^{\geq}}}(X_i^{\geq}) \wedge f^P_{X_i^{\geq}}(y) \geq \alpha_{X_i^{\geq}}\right\} = \\
&= \underline{P}^{\alpha_{X_i^{\geq}}}(X_i^{\geq}) \cup \left\{y \in D_P^+(x) : x \in \underline{P}^{\alpha_{X_i^{\geq}}}(X_i^{\geq}) \wedge f^P_{X_i^{\geq}}(y) \geq \alpha_{X_i^{\geq}}\right\}, \\
&POS_P^{\alpha_{X_i^{\leq}}}(X_i^{\leq}) = \\
&= \left\{y \in X_i^{\leq} : f^P_{X_i^{\leq}}(y) \geq \alpha_{X_i^{\leq}}\right\} \cup \left\{y \in D_P^-(x) : x \in \underline{P}^{\alpha_{X_i^{\leq}}}(X_i^{\leq}) \wedge f^P_{X_i^{\leq}}(y) \geq \alpha_{X_i^{\leq}}\right\} = \\
&= \underline{P}^{\alpha_{X_i^{\leq}}}(X_i^{\leq}) \cup \left\{y \in D_P^-(x) : x \in \underline{P}^{\alpha_{X_i^{\leq}}}(X_i^{\leq}) \wedge f^P_{X_i^{\leq}}(y) \geq \alpha_{X_i^{\leq}}\right\},
\end{aligned} \tag{3.41}
$$

$$
\begin{aligned}
&POS_P^{\beta_{X_i^{\geq}}}(X_i^{\geq}) = \\
&= \left\{y \in X_i^{\geq} : g^P_{X_i^{\geq}}(y) \leq \beta_{X_i^{\geq}}\right\} \cup \left\{y \in D_P^+(x) : x \in \underline{P}^{\beta_{X_i^{\geq}}}(X_i^{\geq}) \wedge g^P_{X_i^{\geq}}(y) \leq \beta_{X_i^{\geq}}\right\} = \\
&= \underline{P}^{\beta_{X_i^{\geq}}}(X_i^{\geq}) \cup \left\{y \in D_P^+(x) : x \in \underline{P}^{\beta_{X_i^{\geq}}}(X_i^{\geq}) \wedge g^P_{X_i^{\geq}}(y) \leq \beta_{X_i^{\geq}}\right\}, \\
&POS_P^{\beta_{X_i^{\leq}}}(X_i^{\leq}) = \\
&= \left\{y \in X_i^{\leq} : g^P_{X_i^{\leq}}(y) \leq \beta_{X_i^{\leq}}\right\} \cup \left\{y \in D_P^-(x) : x \in \underline{P}^{\beta_{X_i^{\leq}}}(X_i^{\leq}) \wedge g^P_{X_i^{\leq}}(y) \leq \beta_{X_i^{\leq}}\right\} = \\
&= \underline{P}^{\beta_{X_i^{\leq}}}(X_i^{\leq}) \cup \left\{y \in D_P^-(x) : x \in \underline{P}^{\beta_{X_i^{\leq}}}(X_i^{\leq}) \wedge g^P_{X_i^{\leq}}(y) \leq \beta_{X_i^{\leq}}\right\},
\end{aligned} \tag{3.42}
$$

Lemma 3.6.1. *P-positive regions defined according to (3.39) and (3.40) differ in general from P-lower approximations defined according to (3.11) and (3.13).*

Let us observe that according to definitions (3.11) and (3.39), using property (3.41):

$$
\begin{aligned}
&\underline{P}^{\alpha_{X_i^{\geq}}}(X_i^{\geq}) = \left\{y \in U : f^P_{X_i^{\geq}}(y) \geq \alpha_{X_i^{\geq}}\right\} = \\
&= \left\{y \in X_i^{\geq} : f^P_{X_i^{\geq}}(y) \geq \alpha_{X_i^{\geq}}\right\} \cup \left\{y \in X_{i-1}^{\leq} : f^P_{X_i^{\geq}}(y) \geq \alpha_{X_i^{\geq}}\right\}, \text{ while} \\
&POS_P^{\alpha_{X_i^{\geq}}}(X_i^{\geq}) = \bigcup_{y \in \underline{P}^{\alpha_{X_i^{\geq}}}(X_i^{\geq})} D_P^+(y) = \\
&\left\{y \in X_i^{\geq} : f^P_{X_i^{\geq}}(y) \geq \alpha_{X_i^{\geq}}\right\} \cup \left\{y \in D_P^+(x) : x \in \underline{P}^{\alpha_{X_i^{\geq}}}(X_i^{\geq}) \wedge f^P_{X_i^{\geq}}(y) \geq \alpha_{X_i^{\geq}}\right\}.
\end{aligned}
$$

P-lower approximation defined according to (3.11) contains all objects satisfying condition on consistency of belonging to a given upward union of classes $X_i^{\geq}$. P-positive region contains only these objects that satisfy the condition and are dominating objects that belong to the P-lower approximation of $X_i^{\geq}$. Analogous propriety can be shown for the P-lower approximation and P-positive region of a downward union of classes $X_i^{\leq}$. The same can be also shown for definitions (3.13) and (3.40).

Moreover, from the same reason, if one would consider a P-positive region of $X_i^{\geq}$ that is composed of objects dominating objects that belong to P-lower approximation of $X_i^{\geq}$ defined by (3.11), (3.13), it would differ from the P-positive region defined according to (3.39), (3.40).

P-negative region

We define P-negative and P-boundary regions of approximated sets, for $P \subseteq C, X_i^{\geq}, X_i^{\leq} \subseteq U$, and $\alpha_{X_i} \in [0,1]$, $\beta_{X_i} \in [0,1]$, as the following:

$$\begin{aligned} NEG_P^{\alpha_{X_i^{\geq}}}(X_i^{\geq}) &= POS_P^{\alpha_{X_i^{\geq}}}(X_{i-1}^{\leq}) - POS_P^{\alpha_{X_i^{\geq}}}(X_i^{\geq}), \\ NEG_P^{\alpha_{X_i^{\leq}}}(X_i^{\leq}) &= POS_P^{\alpha_{X_i^{\leq}}}(X_{i+1}^{\geq}) - POS_P^{\alpha_{X_i^{\leq}}}(X_i^{\leq}), \end{aligned} \tag{3.43}$$

$$\begin{aligned} NEG_P^{\beta_{X_i^{\geq}}}(X_i^{\geq}) &= POS_P^{\beta_{X_i^{\geq}}}(X_{i-1}^{\leq}) - POS_P^{\beta_{X_i^{\geq}}}(X_i^{\geq}), \\ NEG_P^{\beta_{X_i^{\leq}}}(X_i^{\leq}) &= POS_P^{\beta_{X_i^{\leq}}}(X_{i+1}^{\geq}) - POS_P^{\beta_{X_i^{\leq}}}(X_i^{\leq}). \end{aligned} \tag{3.44}$$

P-boundary region

$$\begin{aligned} BND_P^{\alpha_{X_i^{\geq}}}(X_i^{\geq}) &= (U - POS_P^{\alpha_{X_i^{\geq}}}(X_i^{\geq})) - NEG_P^{\alpha_{X_i^{\geq}}}(X_i^{\geq}), \\ BND_P^{\alpha_{X_i^{\leq}}}(X_i^{\leq}) &= (U - POS_P^{\alpha_{X_i^{\leq}}}(X_i^{\leq})) - NEG_P^{\alpha_{X_i^{\leq}}}(X_i^{\leq}), \end{aligned} \tag{3.45}$$

$$\begin{aligned} BND_P^{\beta_{X_i^{\geq}}}(X_i^{\geq}) &= (U - POS_P^{\beta_{X_i^{\geq}}}(X_i^{\geq})) - NEG_P^{\beta_{X_i^{\geq}}}(X_i^{\geq}), \\ BND_P^{\beta_{X_i^{\leq}}}(X_i^{\leq}) &= (U - POS_P^{\beta_{X_i^{\leq}}}(X_i^{\leq})) - NEG_P^{\beta_{X_i^{\leq}}}(X_i^{\leq}). \end{aligned} \tag{3.46}$$

Analogously as in case of VC-IRSA, once decision rules are learned, they can be applied by a classifier (see chapter 5) to suggest assignment of objects to classes. The rules are learned from P-positive regions of the unions of decision classes. This type of structuring of the data involves an a priori restriction of the set of objects, on which the classifier is learned. The rough set analysis enables estimation of the attainable predictive accuracy before learning of a classifier occurs. A classifier learned on P-positive regions of unions of decision classes *may* correctly assign object $y \in X_i$ to class X_i if y belongs to the P-positive region of $X_i^{\geq}$ or $X_i^{\leq}$.

The following two measures estimate the predictive accuracy that may be attained by the classifier. The first, λ measure, estimates the ratio of objects in the data table that may be learned by the classifier:

λ measure

$$\lambda_P^{\boldsymbol{\alpha}_X} = \frac{\left|X_1 \cap POS_P^{\alpha_{X_1^{\leq}}}(X_1^{\leq})\right|}{|U|} + \tag{3.47}$$
$$+ \frac{\bigcup_{i=2}^{n-1}\left|X_i \cap (POS_P^{\alpha_{X_i^{\geq}}}(X_i^{\geq}) \cup POS_P^{\alpha_{X_i^{\leq}}}(X_i^{\leq}))\right|}{|U|} + \frac{\left|X_n \cap POS_P^{\alpha_{X_n^{\geq}}}(X_n^{\geq})\right|}{|U|},$$

$$\lambda_P^{\boldsymbol{\beta}_X} = \frac{\left|X_1 \cap POS_P^{\beta_{X_1^{\leq}}}(X_1^{\leq})\right|}{|U|} + \tag{3.48}$$
$$+ \frac{\bigcup_{i=2}^{n-1}\left|X_i \cap (POS_P^{\beta_{X_i^{\geq}}}(X_i^{\geq}) \cup POS_P^{\beta_{X_i^{\leq}}}(X_i^{\leq}))\right|}{|U|} + \frac{\left|X_n \cap POS_P^{\beta_{X_n^{\geq}}}(X_n^{\geq})\right|}{|U|},$$

where n is the number of the decision classes.

The second, δ measure, estimates the average minimal absolute difference between index of the class to which an object may be assigned and index of the class to which the object belongs. For $i : y_j \in X_i$, it is defined as:

δ measure

$$\delta_P^{\boldsymbol{\alpha}_X} = \frac{1}{|U|}\sum_{j=1}^{|U|} \min_{k\,:\,y_j \in POS_P^{\alpha_{X_k^{\geq}}}(X_k^{\geq}) \vee y_j \in POS_P^{\alpha_{X_k^{\leq}}}(X_k^{\leq})} |i-k|, \tag{3.49}$$

$$\delta_P^{\boldsymbol{\beta}_X} = \frac{1}{|U|}\sum_{j=1}^{|U|} \min_{k\,:\,y_j \in POS_P^{\beta_{X_k^{\geq}}}(X_k^{\geq}) \vee y_j \in POS_P^{\beta_{X_k^{\leq}}}(X_k^{\leq})} |i-k|. \tag{3.50}$$

Both these measures can be used to characterize the data set on which the classifier is learned.

3.7 Summary

In this chapter, we considered consistency measures for VC-DRSA. Their properties are summarized in Table 3.2. Remark that $\epsilon^P_{X_i^{\geq}}(y)$ and $\epsilon^P_{X_i^{\leq}}(y)$ are missing desirable property (m3). Therefore, two possible modifications of these measures, denoted by $\epsilon^{*P}_{X_i^{\geq}}(y)$, $\epsilon^{*P}_{X_i^{\leq}}(y)$ and $\epsilon^{'P}_{X_i^{\geq}}(y)$, $\epsilon^{'P}_{X_i^{\leq}}(y)$, were further investigated.

We defined monotonic lower approximations for those of consistency measures. These lower approximations have all considered monotonicity properties. Further, the monotonic lower approximations were used to define positive, negative and boundary regions which, as it was presented, are more desirable basis for the induction of the decision

rules. Moreover, we defined measures that estimate the predictive accuracy attainable to a classifier learned on positive regions.

As a conclusion, we can recommend using consistency measure ϵ^* or $\epsilon^{'}$. These measures have all required monotonicity properties and are much less computationally intensive than monotonic measures $\overline{\mu}$.

Table 3.2: Monotonicity of consistency measures considered for VC-DRSA.

consistency measure	(m1)	(m2)	(m3)	(m4)
$\mu^P_{X_i^{\geq}}(y)$, $\mu^P_{X_i^{\leq}}(y)$	no	yes	yes	no
$\mu^{'P}_{X_i^{\geq}}(y)$, $\mu^{'P}_{X_i^{\leq}}(y)$	no	yes	yes	yes
$B^P_{X_i^{\geq}}(y)$, $B^P_{X_i^{\leq}}(y)$	no	no	no	no
$\beta^P_{X_i^{\geq}}(y)$, $\beta^P_{X_i^{\leq}}(y)$	no	yes	yes	yes
$\epsilon^P_{X_i^{\geq}}(y)$, $\epsilon^P_{X_i^{\leq}}(y)$	yes	yes	no	yes
$\epsilon^{*P}_{X_i^{\geq}}(y)$, $\epsilon^{*P}_{X_i^{\leq}}(y)$	yes	yes	yes	yes
$\epsilon^{'P}_{X_i^{\geq}}(y)$, $\epsilon^{'P}_{X_i^{\leq}}(y)$	yes	yes	yes	yes
$\overline{\mu}^P_{X_i^{\geq}}(y)$, $\overline{\mu}^P_{X_i^{\leq}}(y)$	yes	yes	yes	yes

CHAPTER 4

Rule Models

4.1 Introduction

In VC-IRSA and VC-DRSA, induction of decision rules is subsequent to computation of probabilistic rough approximations. In computation of rough approximations of a set, objects are divided into lower and upper approximations based on their consistency calculated with respect to (w.r.t.) this set (see sections 2.4 and 3.5). Since it is impossible to induce decision rules directly from rough approximations (see sections 2.6 and 3.6), they are induced from positive regions. The purpose of computation of rough approximations and positive regions is to identify sufficiently consistent objects. This process can be viewed as a kind of preprocessing of data. Objects identified as sufficiently consistent are a good basis for induction of decision rules. The purpose of induction of decision rules is to discover strong relationships between description of these objects and their membership to a set. If the rules are intended to be used in classification, then the goal of the induction procedure is to find a preferably small set of rules with high predictive accuracy.

Induction of ordinal decision rules for VC-DRSA is a more general problem than induction of decision rules for VC-IRSA. In VC-IRSA, elementary conditions of decision rules have the form: $attribute = value$. In VC-DRSA, the elementary conditions have a more general form: $attribute \geq value$ or $attribute \leq value$. Moreover, in VC-IRSA decision rules assign to decision classes while in VC-DRSA they assign to unions of decision classes. In this chapter, the rule induction algorithms are presented from VC-DRSA perspective. Nevertheless, they can be easily adopted to VC-IRSA.

First, we define the syntax and semantics of decision rules. Then, we follow with specification of characteristics of decision rules induced in variable consistency rough set approaches. We present VC-DomLEM, which is an algorithm to induce decision rules by sequential covering (Han and Kamber, 2006), also called separate and conquer (Fürnkranz, 1999). VC-DomLEM is applied general learning framework of bagging (Breiman, 1996). The resulting algorithm, which uses information about object consistency is called variable consistency bagging (VC-bagging). It constructs an ensemble of decision rules classifiers.

4.2 The syntax and semantics of decision rules

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

$$\textit{if } \Phi \textit{ then } \Psi,$$

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 set 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 P-positive region of the considered set is computed.

In order to avoid repetition of the same definitions and properties for VC-IRSA and VC-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. Further, let us denote by Θ a consistency measure used to compute the lower approximation of any X. Let us denote by θ_X the consistency threshold on measure Θ w.r.t. set X. Then, for a given $P \subseteq C$ and $X \subset U$, we can denote probabilistic lower approximation of X by $\underline{P}^{\theta_X}(X)$. The set of objects belonging to $\underline{P}^{\theta_X}(X)$ is the basis for induction of a set of decision rules $R_X^{\hat{\theta}_X}$, i.e., rules assigning objects to set X. The elementary conditions (selectors) in decision rules belonging to $R_X^{\hat{\theta}_X}$ are taking values from objects belonging to lower approximation $\underline{P}^{\theta_X}(X)$. 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 suggest 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 consistency measures used in the definition of probabilistic P-lower approximation. Different rule consistency measures are discussed in section 4.3. The value of threshold $\hat{\theta}_X$ depends on the value of threshold θ_X, which is also shown in section 4.3.

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

$$\begin{aligned} &\textit{if } q_{i_1}(y) \succeq r_{i_1} \wedge \ldots \wedge q_{i_p}(y) \succeq r_{i_p} \wedge g_{i_{p+1}}(y) = r_{i_{p+1}} \wedge \ldots \wedge g_{i_z}(y) = r_{i_z} \\ &\qquad\qquad \textit{then } y \in X^{\geq}, \end{aligned} \tag{4.1}$$

$$\begin{aligned} &\textit{if } q_{i_1}(y) \preceq r_{i_1} \wedge \ldots \wedge q_{i_p}(y) \preceq r_{i_p} \wedge g_{i_{p+1}}(y) = r_{i_{p+1}} \wedge \ldots \wedge g_{i_z}(y) = r_{i_z} \\ &\qquad\qquad \textit{then } y \in X^{\leq}, \end{aligned} \tag{4.2}$$

where $q_j, j \in \{i_1, i_2, \ldots, i_p\}$ denotes criterion and $g_j, j \in \{i_{p+1}, i_{p+2}, \ldots, i_z\}$ denotes regular attribute. Moreover, $r_j \in V_j$, $j = \{i_1, i_2, \ldots, i_p, i_{p+1}, i_{p+2}, \ldots, i_z\}$ are values from the domain of criterion q_j or regular attribute g_j. We use symbols $\succeq$ and $\preceq$ to indicate weak preference w.r.t. single criterion and inverse weak preference, respectively. If $q_j \in Q$ is a gain (cost) criterion, then elementary condition $q_j(y) \succeq r_j$ means that the value on criterion $q_j(y)$ is not smaller (not greater) than value r_j. Elementary conditions for regular attributes are of type $g_j(y) = r_j$.

Decision rule $r_X^{\hat{\theta}_X}$ covers objects that fulfill its condition part and suggest their assignment to set X. The condition part of $r_X^{\hat{\theta}_X}$ rule can be denoted by $\Phi_{r_X^{\hat{\theta}_X}}$ while its decision part can be denoted by $\Psi_{r_X^{\hat{\theta}_X}}$ (Greco et al., 2008c). 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: basic rule measures

$$\text{support of } r_X^{\hat{\theta}_X} : supp(r_X^{\hat{\theta}_X}) = \left| \|\Phi_{r_X^{\hat{\theta}_X}}\| \cap \|\Psi_{r_X^{\hat{\theta}_X}}\| \right|, \tag{4.3}$$

$$\text{strength of } r_X^{\hat{\theta}_X} : \sigma(r_X^{\hat{\theta}_X}) = \frac{supp(r_X^{\hat{\theta}_X})}{|U|}, \tag{4.4}$$

$$\text{certainty of } r_X^{\hat{\theta}_X} : cer(r_X^{\hat{\theta}_X}) = \frac{supp(r_X^{\hat{\theta}_X})}{\left|\|\Phi_{r_X^{\hat{\theta}_X}}\|\right|}, \tag{4.5}$$

$$\text{coverage of } r_X^{\hat{\theta}_X} : cov(r_X^{\hat{\theta}_X}) = \frac{supp(r_X^{\hat{\theta}_X})}{\left|\|\Psi_{r_X^{\hat{\theta}_X}}\|\right|}, \tag{4.6}$$

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.

4.3 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 (Clark and Niblett, 1989; Clark and Boswell, 1991; Cohen, 1995; Weiss and Indurkhya, 2000). 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 (Iba et al., 1988).

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:

rule consistency measures

$$\epsilon\text{-consistency of } r_X^{\hat{\theta}_X} : \epsilon(r_X^{\hat{\theta}_X}) = \frac{\left|\|\Phi_{r_X^{\hat{\theta}_X}}\| \cap \neg \underline{P}^{\theta_X}(X)\right|}{|\neg \underline{P}^{\theta_X}(X)|}, \tag{4.7}$$

$$\epsilon'\text{-consistency of } r_X^{\hat{\theta}_X} : \epsilon'(r_X^{\hat{\theta}_X}) = \frac{\left|\|\Phi_{r_X^{\hat{\theta}_X}}\| \cap \neg \underline{P}^{\theta_X}(X)\right|}{|\underline{P}^{\theta_X}(X)|}, \tag{4.8}$$

$$\mu\text{-consistency of } r_X^{\hat{\theta}_X} : \mu(r_X^{\hat{\theta}_X}) = \frac{\left|\|\Phi_{r_X^{\hat{\theta}_X}}\| \cap \underline{P}^{\theta_X}(X)\right|}{\left|\|\Phi_{r_X^{\hat{\theta}_X}}\|\right|}, \tag{4.9}$$

where $\hat{\theta}_X = \frac{|\neg X|}{|\neg \underline{P}^{\theta_X}(X)|}\theta_X$ in definition (4.7), $\hat{\theta}_X = \frac{|X|}{|\underline{P}^{\theta_X}(X)|}\theta_X$ in definition (4.8), and $\hat{\theta}_X = \theta_X$ in definition (4.9).

ϵ-consistency measure is related to cost-type consistency measure ϵ defined as (2.3) and as (3.6). ϵ'-consistency measure is related to cost-type consistency measure ϵ' defined as (2.4) and as (3.9). μ-consistency measure is related to gain-type rough membership measure μ used in definitions (2.1) and as (3.1). It can be shown that each of the defined above rule consistency measures derives monotonicity properties from the corresponding object consistency measure. We do not apply other measures concerned in chapters 2 and 3. Consistency measure $\overline{\mu}$ (see section 2.5.5 and section 3.5.8) has too high computational complexity.

As it will be shown in section 4.4, 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 4.4, it is possible, with some additional steps, to induce rules satisfying constraints on μ-consistency from positive regions computed using 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 consistency measures ϵ, ϵ' and μ. In the definitions of rule consistency measures, $\underline{P}^{\theta_X}(X)$ is used instead of X. 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 (see sections 2.5 and 3.5), 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 4.4, 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.

discriminant decision rule

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 thesis, rule is discriminant if it covers only objects belonging to positive region $POS_P^{\theta_X}(X)$ from its complement.

minimal decision rule

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 4.2, $r_X^{\hat{\theta}_X}$ 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 $\|\Psi_{r_Y^{\hat{\theta}_Y}}\| \subseteq \|\Psi_{r_X^{\hat{\theta}_X}}\|$.

complete set of rules

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 $r_X^{\hat{\theta}_X} \in R_X^{\hat{\theta}_X}$.

non-redundant decision rule

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 (Michalski, 1993; Michalski and Kaufman, 1998), as well as in FOIL (Quinlan, 1990; Quinlan and Cameron-Jones, 1993), 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, in particular, LEM2 (Grzymała-Busse, 1992; Grzymała-Busse and Lakshmanan, 1996; Grzymała-Busse and Wang, 1997; Grzymała-Busse, 1997; Grzymała-Busse and Zou, 1998; Grzymała-Busse and Stefanowski, 2001), and DomLEM (Greco et al., 2000a, 2001b). The requirement of completeness is however softened in case of pruned sets of rules induced by IREP (Fürnkranz and Widmer, 1994), RIPPER (Cohen, 1995) or SLIPPER (Cohen and Singer, 1999). In other cases, like Lightweight Rule Induction (LRI) (Weiss and Indurkhya, 2000), 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 (Rückert and Kramer, 2006), where it is assumed that the number of induced rules is parametrized. 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 (Rückert and Kramer, 2006).

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 (Friedman, 2006; Friedman and Popescu, 2008; Greco et al., 2008a). Not only a precise response of a classifier but also interpretable justification of presented suggestion is considered to be important.

4.4 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, is inspired by LEM2 algorithm (Grzymała-Busse, 1992) and its adaptation to ordinal data called DomLEM (Greco et al., 2000a). VC-DomLEM induces rules for classification problems addressed in VC-IRSA and ordinal classification problems considered in VC-DRSA (Błaszczyński et al., 2009c; Błaszczyński et al., accepted for publication 2010). It can be also easily adapted to induce certain, possible and approximate rules in IRSA, as well as certain and possible rules in DRSA. This algorithm heuristically searches for rules whose consistency measures (4.7), (4.8) or (4.9) satisfy a specified threshold value. The applied heuristic strategy is called sequential covering (Han and Kamber, 2006) or separate and conquer (Michalski, 1969; Pagallo and Haussler, 1990; Fürnkranz, 1999). 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, LEM2, IREP, RIPPER and DomLEM.

VC-DomLEM induces a minimal set $\mathbf{R}$ of minimal decision rules. This algorithm is composed of two parts. The first part is presented as Algorithm 1, while the second one is presented as Algorithm 2. In the following, we describe both parts, referring to numbered lines of the algorithms.

In Algorithm 1, set of rules $R_X^{\hat{\theta}_X}$ is induced for each set of objects X by method

VC-SequentialCovering^{mix}, which is presented as Algorithm 2. *VC-SequentialCovering*^{mix} is inducting rules using elementary conditions constructed on attributes from set $P \subseteq C$ (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) implies induction of rules covering the positive region only. Option 3) implies induction of rules that may cover any object in the data set. Such rules, in general, may be composed of fewer elementary conditions than those induced according to option 1). Option 2) is intermediate between option 1) and option 3) – it does not allow rules to cover 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 $P \subseteq C$ of attributes,
rule consistency measure $\hat{\Theta}$,
set $\{\hat{\theta}_X : X \in \mathbf{X}\}$ of rule consistency measure thresholds,
object covering option s.

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

1 $\mathbf{R} := \emptyset$;
2 **foreach** *element* $X \in \mathbf{X}$ **do**
3 $\quad AO_P^{\theta_X}(X) :=$ `AllowedObjects` (X, P, θ_X, s);
4 $\quad R_X^{\hat{\theta}_X} :=$ *VC-SequentialCovering*^{mix} $(\underline{P}^{\theta_X}(X),\ AO_P^{\theta_X}(X),\ P,\ \hat{\Theta},\ \hat{\theta}_X)$;
5 $\quad \mathbf{R} := \mathbf{R} \cup R_X^{\hat{\theta}_X}$;
6 $\quad$ `RemoveNonMinimalRules` $(\mathbf{R})$;

In Algorithm 2, rules for a given set X are induced by *VC-SequentialCovering*^{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 (4.7), (4.8) or (4.9) 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 10. 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 are not covered by the constructed 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 10). 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 11). Objects that are covered by the rule are removed from set B, which is the base for building candidate elementary conditions (line 12).

Constructed set of rules $R_X^{\hat{\theta}_X}$ is checked for redundancy in line 13. 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 worst (i.e., 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 $R_X^{\hat{\theta}_X}$.

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

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

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

```
1  B := P^{θ_X}(X);
2  R_X^{θ̂_X} := ∅;
3  while B ≠ ∅ do
4      r_X^{θ̂_X} := ∅;
5      EC := ElementaryConditions (B, P);
6      while (Θ̂(r_X^{θ̂_X}) does not satisfy θ̂_X) or (‖Φ_{r_X^{θ̂_X}}‖ ⊈ AO_P^{θ_X}(X)) do
7          ec := BestElementaryCondition (EC, r_X^{θ̂_X}, Θ̂, P^{θ_X}(X));
8          r_X^{θ̂_X} := r_X^{θ̂_X} ∪ ec;
9          EC := ElementaryConditions (B ∩ supp(r_X^{θ̂_X}), P);
10     RemoveRedundantElementaryConditions (r_X^{θ̂_X}, Θ̂, θ̂_X, AO_P^{θ_X}(X));
11     R_X^{θ̂_X} := R_X^{θ̂_X} ∪ r_X^{θ̂_X};
12     B := B \ supp(r_X^{θ̂_X});
13 RemoveRedundantRules (R_X^{θ̂_X}, Θ̂, P^{θ_X}(X));
```

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

Monotonicity properties of rule consistency measures: ϵ-consistency (4.7) and ϵ'-consistency (4.8), allow to increase efficiency of rule induction in *VC-SequentialCovering*^{mix} algorithm. These properties are derived from corresponding consistency measures ϵ (see definitions (2.3) and (3.6)) and ϵ' (see definitions (2.4) and (3.9)).

There are two scenarios defined for *VC-SequentialCovering*^{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,

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:

a) the best (i.e., the smallest) value of 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)

b) the best (i.e., the greatest) value of $\left|\|\Phi_{r_X^{\hat{\theta}_X} \cup ec}\| \cap \underline{P}^{\theta_X}(X)\right|$.

Theorem 4.4.1. *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 the first scenario or $\hat{\theta}_X = \frac{|X|}{|\underline{P}^{\epsilon'_X}(X)|}\epsilon'_X$ in the second scenario.*

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 $\hat{\theta}_X$. □

According to theorems 2.5.4, 3.5.10 and theorems 2.5.7, 3.5.22 both ϵ and ϵ' share 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 elementary conditions of the rule. New elementary condition constructed on an attribute already present in the rule decreases the quality of the rule, measured by its consistency and the number of covered objects from the P-lower approximation of X, as shown by the following theorem.

Theorem 4.4.2. *For VC-SequentialCovering^{mix}, subject to one of the scenarios: α) or β) and applying γ) condition quality measures, 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 $a_i \in R$, 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 $a_i \in R$ 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 at that time 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 has been 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. □

Theorem 4.4.2 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) (according to theorems 3.5.14 and 3.5.27). This allows us to further increase the efficiency of the rule induction algorithm. We can sort elementary conditions on each criterion $q \in Q$, where $Q \subseteq C$, 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 consistency measures ϵ and ϵ'.

The remaining processing after the sorting is simple because we search for elementary conditions with the best value of 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 (Weiss and Indurkhya, 2000), 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 (3.7), $\epsilon^{*P}_{X_i^{\geq}}(y) \geq \epsilon^{P}_{X_i^{\geq}}(y)$, $\forall y \in U, X_i^{\geq} \subseteq U, P \subseteq C$. 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 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.

4.4.2 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, resulting from lack of monotonicity property (m4) of rough membership measure μ (see theorem 3.5.2). Notice that μ-consistency measure is also missing property (m1) (see theorems 2.5.1 and to 3.5.1), 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 consistency measure μ' defined as (3.30) or (3.31). The problem that can be faced by VC-DomLEM during induction of rules is presented in the following Example 4.4.1 and Figure 4.1.

Example 4.4.1. *Applying in equation (3.12) consistency measure μ' defined as (3.30), 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 (3.39), $POS_P^{0.75}(X_2^{\geq}) = \{y_1, y_2, y_3, y_6\}$.

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 (4.9). 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 $VC\text{-}SequentialCovering^{mix}$ 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 a criterion 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 to reduce of the set of objects considered when creating elementary conditions by using edge regions of unions of classes $X_i^{\geq}$ and $X_i^{\leq}$.

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: *P-edge region*

$$EDGE_P^{\theta_{X_i^{\geq}}}(X_i^{\geq}) = \{y \in \underline{P}^{\theta_{X_i^{\geq}}}(X_i^{\geq}) : z \in D_P^-(y) \cap \underline{P}^{\theta_{X_i^{\geq}}}(X_i^{\geq}) \Rightarrow z \in D_P^+(y)\}, \quad (4.10)$$

$$EDGE_P^{\theta_{X_i^{\leq}}}(X_i^{\leq}) = \{y \in \underline{P}^{\theta_{X_i^{\leq}}}(X_i^{\leq}) : z \in D_P^+(y) \cap \underline{P}^{\theta_{X_i^{\leq}}}(X_i^{\leq}) \Rightarrow z \in D_P^-(y)\}. \quad (4.11)$$

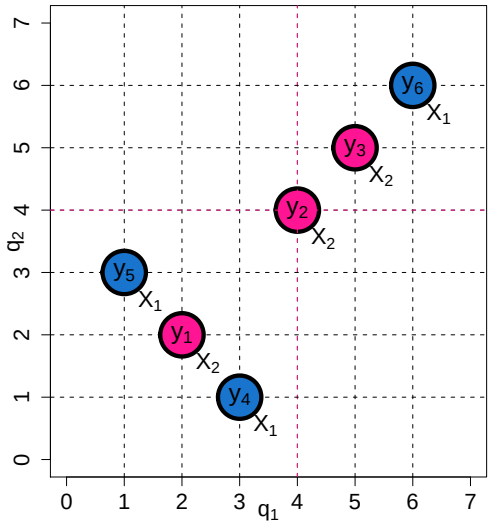


Figure 4.1: Illustration of VC-DomLEM problems with induction of rules satisfying μ-consistency condition, caused by lack of property (m4).

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 $VC\text{-}SequentialCovering^{mix}$ 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 $VC\text{-}SequentialCovering^{mix}$ 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 4.4.3. *For VC-SequentialCovering^{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. □

In order to adjust VC-DomLEM algorithm for μ-consistency measure, we need to modify line 1 of method *VC-SequentialCovering^{mix}*. This modification consists in substituting the P-edge region of set X for the P-lower approximation $\underline{X}$ of this set. In this way, the set of objects for which elementary conditions are constructed is reduced.

Presented modification of VC-DomLEM algorithm implies additional computational cost because P-edge regions must be calculated. On the other hand, an edge region is smaller than the corresponding lower approximation, thus the number of potential elementary conditions to be checked by VC-DomLEM is also smaller. Moreover, as we have shown in Example 4.4.1, without this modification it might be impossible to induce rules having value of μ-consistency measure not lower than specified threshold $\hat{\theta}_X$. If the reduction of the set of objects is considerable, significantly smaller space of elementary conditions is searched.

4.4.3 Induction of random rules satisfying ϵ-consistency and ϵ'-consistency condition

VC-DomLEM, when it is inducing rules satisfying ϵ-consistency and ϵ'-consistency condition, can be used in a setting inspired by Random Forests (Breiman, 2001). In this setting, in *VC-SequentialCovering^{mix}* method, elementary conditions constructed on

the basis of a random subsets of attributes $P \subseteq C$ of fixed size are considered when selecting the best elementary condition that is added to the rule in line 7. The cardinality of set P is parametrized. The rest of the induction algorithm is the same as described in section 4.4.1.

This version of VC-DomLEM is intended to be used in ensembles of classifiers. It is expected to increase the diversity of such ensembles due to randomization of the set of attributes on which elementary conditions are constructed.

4.5 Induction of ensembles of decision rule classifiers in VC-bagging

We propose a generalization of the bagging scheme, called *variable consistency bagging* (VC-bagging) (Błaszczyński et al., accepted for publication 2009, 2009b), where the sampling of objects is controlled by consistency measures. This extension of bagging shares some of its motivations with VC-IRSA and VC-DRSA but can be easily used with almost any learning algorithm. The only requirement is unstability of the algorithm, which is also postulated by standard bagging scheme (see section 4.5.1).

Let us remark that the main idea of the standard version of *bagging* method (Breiman, 1996) is that several classifiers, called component or base classifiers, are induced by the same learning algorithm over several different distributions of input objects, which are bootstrap samples obtained by *uniformly sampling* with replacement. Bagging method has been extended in a number of ways in attempt to improve the predictive accuracy of the constructed ensemble. These extensions focused mainly on increasing diversity of component classifiers. *Random forest* (Breiman, 2001) is a well known example of such extension. It uses feature subset randomized decision tree component classifiers (Breiman, 2001). Other extensions of bagging profit from random selection of features. In some cases, several random subspaces of features are selected along with the idea presented as the *random subspaces* method (Ho, 1998). In other cases, the random selection of features is combined with standard bootstrap sampling. Examples of such ensembles of classifiers were considered by different researchers (Patrice et al., 2000; Stefanowski and Kaczmarek, 2004; Panov and Dzeroski, 2007).

By introducing VC-bagging, we also have motivation to increase diversity by changing sampling phase. However, we take into account the postulate saying that base classifiers used in bagging are expected to have sufficiently high predictive accuracy apart from being diversified (Breiman, 2001). In our opinion, this requirement is par-

ticularly important for processing input data containing inconsistent objects. Usually, these inconsistent objects are source of difficulties that may lead to overfitting of the base classifies and decrease their classification performance. Although bagging is known to be less sensitive to overfitting than boosting, we intend to show that it is possible to change input sampling in such a way that component classifiers could be less influenced by inconsistent objects.

Our key concept is to change the standard bootstrap sampling, where each object is assigned the same probability of being sampled, into more focused variable consistency bootstrap sampling, where *consistent objects* are more likely to be selected than inconsistent ones. To identify consistent objects we can use the same consistency measures that were chosen to define probabilistic lower approximations. The intuition here is that decreasing a chance for selecting inconsistent objects should lead to creating more accurate and still sufficiently diversified base classifiers in the bagging scheme. Moreover, we also consider consistency of objects with respect to partial description by the attributes. This results in the consistency being measured in granules on knowledge constructed on random subsets of the set of attributes that describe the problem. When these inconsistent objects are identified on subsets of attributes the intuition is the same but it is expressed with respect to objects that can be basis for construction of consistent patterns.

We would like to stress that we want to identify consistent objects and change their sampling probability in a pre-processing phase before learning base classifiers in a similar manner as probabilistic lower approximations are computed. In this way, we modify the standard bootstrap sampling with uniform probability distribution into more focused distribution where consistent objects are more likely to be selected than inconsistent ones. The goal is to learn component classifiers on more perturbed distributions characterised by higher rates of consistent objects. This is a different approach comparing to an iterative identification of incorrectly classified objects while constructing boosting integrated classifiers (Freund et al., 1997; Schapire and Singer, 1999; Friedman and Popescu, 2008). Boosting consists in subsequent extending of an ensemble of classifiers by adding component classifiers focused on objects incorrectly classified so far. In our approach, we evaluate consistency of objects and we change their sampling probability in a *pre-processing phase* before learning of component classifiers. This is different than evaluating objects' classification in boosting. Additionally, in the way typical for bagging, consistency of objects is calculated independently for each of bootstrap samples.

4.5.1 Bagging scheme

First, let us present the *Bagging* (an acronym from **B**ootstrap **agg**regat**ing**) that was introduced by Breiman (Breiman, 1996). The idea of bagging is quite simple: it combines base classifiers generated by the same learning algorithm from different bootstrap samples of the input training set. The outputs of these classifiers are aggregated by an equal weight voting to make a final classification decision.

bootstrap sample

The diversity results from using different training samples. Each *bootstrap sample* is obtained by sampling objects uniformly with replacement. Each sample contains $n \leq |U|$ objects (usually it has the same size as the original set), however, some objects do not appear in it, while others may appear more than once. The same probability $1/n$ of being sampled is assigned to each object. The probability of an object being selected at least once is $1 - (1 - 1/n)^n$. For a large n, this is about $1 - 1/e$. Each bootstrap sample contains, on the average, 63.2% unique objects from the training set (Breiman, 1996). Thus, on average, approximately 36.8% of objects from the original training set are not present in a given bootstrap sample. We may suspect that some bootstrap samples may contain less misleading training objects than the complete original training set. Consequently more accurate classifiers could be generated and aggregating them may improve classification performance.

The bagging has one parameter m, which is the number of repetitions, i.e, the number of component classifiers that is created. For more details see, e.g., (Breiman, 1996; Kuncheva, 2004). Let us also remark that the bagging is a kind of parallelization in training and classification phases, i.e., there is no transfer of additional information between components unlike it happens in the boosting which iteratively builds a new classifier using information about performance of the previously generated base classifiers.

Bagging is a learning framework in which almost any learning algorithm can be used. Many experimental results show a significant improvement of the classification accuracy, in particular, using decision tree classifiers. An improvement is also observed when using rule classifiers. However, the choice of a base classifier is not indifferent. According to Breiman (Breiman, 1996), what makes a base classifier suitable is its *unstability*. A base classifier is unstable, when small changes in the training set do cause major changes in the classifier. For instance, the decision tree and rule classifiers are unstable, while k-Nearest Neighbor classifiers are not. For more theoretical discussion on the justification of "why bagging works" please refer to (Breiman, 1996; Kuncheva, 2004).

4.5.2 Variable consistency sampling

The goal of variable consistency sampling is to increase predictive accuracy of bagged classifiers by using additional information that reflects the treatment of inconsistency of objects, i.e., variation of consistency of objects, which could be easily applied to the training set. The resulting bagged classifiers are trained on bootstrap samples slightly shifted towards more consistent objects.

In general, the above idea is partly related to some earlier proposals of changing probability distributions while constructing bagging inspired ensembles. In particular, Breiman refers to methods that can improve accuracy of unstable classifiers by perturbing and combining (P&C). The key concept of the P&C method is to generate multiple versions of the classifier by perturbing the training set and then to combine these multiple versions into a single classifier. Breiman proposed some P&C methods for bagged classifiers. Among them are *Arcing Classifiers* (Breiman, 1998), *Pasting Small Votes* (Breiman, 1999) and *Random Forests* (Breiman, 2001). These methods perturb the training data by sampling objects sequentially with replacement, where at each step the probability of selecting a given object is modified by its importance. This importance is estimated at each step by the accuracy of a new base classifier. Importance sampling is known to provide better results than the standard bagging scheme.

The reader familiar with ensemble classifiers can notice other solutions for taking into account accuracy of base classifiers in the process of learning of the ensemble. In boosting more focus is given on objects difficult to be classified by iteratively extended set of base classifiers. We argue that these ensembles are based on a different principle of stepwise adding classifiers and using accuracy from the previous step of learning while changing weights of objects.

Variable consistency sampling could be seen as similar to the described above importance sampling because it is also based on modification of probability distribution. However, there is a striking difference between these two approaches, which is grounded in the fact that consistency of the training objects is evaluated in the pre-processing stage before learning of the base classifiers. Moreover, consistency of objects is intended to be a simpler measure than importance in terms of computational expense resulting from its calculation. Our expectation is that drawing bootstrap samples from a distribution that reflects their consistency will not decrease the diversity of the samples.

Our other observation is that estimating the role of objects in the pre-processing of training data is more similar to previous works on edited k-nearest neighbor classifiers, where the most useful training objects for correct classification are kept, while noisy or

borderline objects are removed, see, e.g., (Wilson and Martinez, 2000). For instance, the IBL3 algorithm (Aha and Kibler, 1989, 1991), which keeps the most useful objects for correct classification and removes noisy or borderline examples, is more accurate than IBL2 version, which focuses on difficult examples from border between classes. Similar performance of a variant of nearest neighbor cleaning rule in a specific approach to pre-processing of imbalanced data was observed in (Stefanowski and Wilk, 2007).

The VC-bagging learning algorithm presented as Algorithm 3 is almost the same as the standard bagging scheme. The difference lies in variable consistency sampling, which is a modified procedure of bootstrap sampling on random subsets of attributes P of specified size $p = |P|$, line 3. This procedure is using consistency of object calculated on random subsets of attributes to construct more consistent bootstrap samples. The cardinality p of random subsets of attributes $P \subseteq C$ is limited by the size of set of condition attributes describing problem. The interpretation of this parameter is that it controls the size of patterns that are identified by the consistency measures in the sampling procedure. The rest of the bagging scheme remains unchanged. It is worth noting that, random subsets of attributes are used only to calculate consistency of objects. Objects with complete description are drawn into bootstrap samples and then used during learning of component classifiers.

Algorithm 3: VC-bagging scheme

Input : LS training set; TS testing set; LA learning algorithm;
Θ_X^P consistency measure;
p number of attributes used in consistency sampling;
m number of bootstrap samples;
Output: C^* final classifier

1 *Learning phase*;
2 **for** $i := 1$ *to* m **do**
3 $S_i :=$ `bootstrap sample of objects, which are drawn by consistency sampling from` LS `with measure` Θ_X^P `calculated on randomly selected set of attributes` P`, such that` $|P| = p$ {sample objects with replacement according to measure Θ_X^P } ;
4 $C_i :=$ LA (S_i) {generate a base classifier} ;
5 *Classification phase*;
6 **foreach** y *in* TS **do**
7 $C^*(y) :=$ `combination of the responses of` $C_i(y)$`, where` $i = 1, \ldots, m$ {the suggestion of the classifier for object y is a combination of suggestions of component classifiers C_i} ;

Consistency measures defined for VC-IRSA in chapter 2, and for VC-DRSA in chap-

ter 3, can be applied to evaluate consistency of object y that is then used in variable consistency sampling. To apply both gain-type consistency measure $f_X^P(y)$ and cost-type consistency measure $g_X^P(y)$ in variable consistency sampling we need to transform them to measure $\Theta_X^P(y)$ defined for a given object y, set of attributes P and set of objects X, as

$$\Theta_X^P(y) = f_X^P(y) \quad \text{or} \quad \Theta_X^P(y) = 1 - g_X^P(y). \tag{4.12}$$

This transformation is valid since consistency measures defined in chapters 2 and 3 take values from $[0, 1]$. One exception is measure ϵ' defined as (2.4) and as (3.9). It needs special treatment in the above transformation since it takes values from $\left[0, \frac{|\neg X^{\geq}|}{|X^{\geq}|}\right]$. Thus, ϵ' needs to be normalized using an upper limit of its domain.

In the variable consistency sampling, objects that are inconsistent on the selected random subset P have decreased probability of being sampled. The value of consistency measure calculated for object y is used to tune the probability of y being sampled to a bootstrap sample, e.g., by calculating a product of $\Theta_X^P(y)$ and $1/|U|$. A consistent object y has $\Theta_X^P(y) = 1$, while inconsistent object y has $0 \leq \Theta_X^P(y) < 1$. Thus, objects that are more consistent (i.e., have higher value of a consistency measure) are more likely to appear in the bootstrap sample. Different object consistency measures may result in different probability of inconsistent object y being sampled. The consistency measures that have property $(m1)$, i.e., that are monotonic with respect to the set of attributes, when are applied in consistency sampling on subsets of attributes P, they allow to identify consistent patterns of at least size p , such that $|P| = p$. The object consistency measures that do not have property $(m1)$ allow to identify consistent patterns of exactly size p.

The responses of component classifiers are combined in line 7 of the algorithm. Different combination rules can be applied to this end, depending on the type of classification method used in component classifiers and the nature of classification problem that is solved. The combination rules for classification methods considered in this work are discussed in section 5.3.

4.6 Summary

In this chapter, we have presented rule induction methods for VC-IRSA and VC-DRSA. We started with proposing a single rule induction algorithm VC-DomLEM. This algorithm may be used in a general framework of consistency sensible ensembles of learning methods, called VC-bagging. Moreover, VC-bagging may also be applied with almost

any unstable learning method that is not necessarily aware of information provided by rough set analysis.

All methods presented here are meant to produce sufficiently accurate and comprehensible classifiers that satisfy constraints imposed on consistency measures. These are the same consistency measures as the ones used to define monotonic probabilistic rough set approaches in chapter 2 and in chapter 3. The predictive abilities of these methods are further investigated in chapter 6.

CHAPTER 5

Rule Classifiers

5.1 Introduction

In this chapter, we present how rule models, which are discussed in chapter 4, can be used to classify objects. Sequential covering methods of rule induction, that we use, do not set any order in produced rule sets. Thus, sets of rules used by classification methods presented in this chapter are unordered. This means that, during classification, each of the rules from the set is matched with each of classified objects. The order in which rules are used in classification has no consequence on classification result. Another approach to classification is implemented by methods that use ordered lists of rules (Clark and Niblett, 1989; Quinlan, 1992).

Two types of classifiers are investigated in this chapter. The first type is a single classifier. Single classifier is using one classification method that applies one set of rules to classify objects. The second type is an ensemble of classifiers. In this case, objects are classified by an ensemble of component classifiers that use the same classification method but that apply different sets of rules to classify objects.

In case of a single classifier, the standard classification method for DRSA is described in (Greco et al., 2002b). In this procedure, an object covered by a set of rules is assigned to a class (or a set of contiguous classes) resulting from intersection of unions of decision classes suggested by the rules. This procedure is described more thoroughly in section 5.2. In (Błaszczyński et al., 2007a), we presented a new procedure for a single classifier in DRSA and in VC-DRSA. We recall and discuss this procedure in section 5.2. It is based on a notion of score coefficient associated with a set of rules covering object and classes to which these rules may assign the object. The score coefficient reflects

relevance between rules and class to which they assign objects. A vector of values of score coefficients calculated for an object with respect to each class can be interpreted as a distribution of relevance between rules that cover classified object and classes.

Finally, we present how to combine responses of component classifiers that use standard and new procedure of classification for DRSA and for VC-DRSA in the ensemble framework. We also address the idea of abstaining in such ensembles of classifiers.

5.2 Classification by a set of decision rules

In this section, a method of classification by a set of decision rules is presented. We introduced this method with VC-DRSA classification in mind. Contrary to the classification method introduced earlier for DRSA (Greco et al., 2002b,a), this method allows to calculate a score coefficient for an object with respect to each of decision classes. The previously proposed method for DRSA will be referred to as standard classification method. The new method is particularly suitable for VC-DRSA but, obviously, it can be applied within DRSA as well (Błaszczyński et al., 2007a).

To start presentation of the new method we distinguish three basic situations that occur while classifying new objects using decision rules. For a given set of rules $\mathbf{R}$, let define a set of decision rules covering an object $y \in U$:

$$cov(y) = \{r \in \mathbf{R} : y \in \|\Phi_r\|\}, \tag{5.1}$$

where $\|\Phi_r\|$ denotes a set of objects satisfying condition part of rule r. Note the difference between $cov(y)$ used for the set of rules covering object y and the coverage of rule defined as (4.6).

In general, only one of the following three situations can occur when matching object y to a set of decision rules $\mathbf{R}$:

1) none of the rules from $\mathbf{R}$ cover object y (i.e., $cov(y) = \emptyset$),

2) exactly one decision rule r covers object y (i.e., $cov(y) = 1$),

3) several rules cover object y (i.e., $cov(y) > 1$).

We consider these three situations below. First, we characterize the source of difficulties while dealing with each situation. Then, we show how these difficulties are overcome by the standard classification method and by the new one.

Situation 1 is clear when we do not consider a partial matching of object y by rules.

- *Standard classification method*: object y is assigned to all considered decision classes. Alternatively, object y may be assigned to the majority class (i.e., it may be assigned to the class with the most objects).

- *New classification method*: as above.

Otherwise, different partial matching strategies (Clark and Niblett, 1989; Clark and Boswell, 1991; Grzymała-Busse, 1994; Grzymała-Busse and Zou, 1998; Słowiński and Stefanowski, 1994; Stefanowski, 1995) could be applied when at least one of rule conditions of some rules from $\mathbf{R}$ is satisfied by the corresponding attributes in the description of object y. The classification method is called abstaining if none of partial matching strategies is applied nor the object is assigned to the majority class.

Situation 2 is also relatively simple.

- *Standard classification method*: the classification is inspired by a prudence principle. For rule $r^{\hat{\theta} X^{\geq}}_{X^{\geq}}$ that matches object y, and assigns it to upward union $X^{\geq}$, the standard classification method assigns object y to the lowest class of the union in the decision part of $r^{\hat{\theta} X^{\geq}}_{X^{\geq}}$. For rule $r^{\hat{\theta} X^{\leq}}_{X^{\leq}}$ that matches object y, and assigns it to downward union $X^{\leq}$, the standard classification method assigns object y to to the highest class of the union in the decision part of $r^{\hat{\theta} X^{\geq}}_{X^{\geq}}$. More formally, if the decision part of rule $r^{\hat{\theta} X^{\geq}}_{X^{\geq}}$ matching object y is "*then* $y \in X^{\geq}_t$", then object y is assigned to class X_t. Analogously, if the decision part of rule $r^{\hat{\theta} X^{\leq}}_{X^{\leq}}$ matching object y is "*then* $y \in X^{\leq}_t$", then object y is assigned to X_t.

- *New classification method*: the classification involves calculation of a score coefficient that reflects relevance between rules and class to which they assign objects. For rule $r^{\hat{\theta} X^{\geq}}_{X^{\geq}}$ matching object y and having decision part "*then* $y \in X^{\geq}_t$", a value of $score_{r^{\hat{\theta} X^{\geq}}_{X^{\geq}}}(X_i, y)$ is calculated for object y and each decision class X_i, such that $i \geq t$:

$$score_{r^{\hat{\theta} X^{\geq}}_{X^{\geq}}}(X_i, y) = \frac{\left| \| \Phi_{r^{\hat{\theta} X^{\geq}}_{X^{\geq}}} \| \cap X_i \right|^2}{\left| \| \Phi_{r^{\hat{\theta} X^{\geq}}_{X^{\geq}}} \| \right| \left| X_i \right|}, \tag{5.2}$$

  where $\| \Phi_{r^{\hat{\theta} X^{\geq}}_{X^{\geq}}} \|$ denotes the set of objects verifying the condition part of rule $r^{\hat{\theta} X^{\geq}}_{X^{\geq}}$, and $\left| \| \Phi_{r^{\hat{\theta} X^{\geq}}_{X^{\geq}}} \| \right|$, $|X_i|$ and $\left| \| \Phi_{r^{\hat{\theta} X^{\geq}}_{X^{\geq}}} \| \cap X_i \right|$ denote cardinalities of the corresponding sets: the set of objects verifying $\Phi_{r^{\hat{\theta} X^{\geq}}_{X^{\geq}}}$, the set of objects belonging to class X_i and the set of objects verifying $\Phi_{r^{\hat{\theta} X^{\geq}}_{X^{\geq}}}$ and belonging to class X_i. Analogously, for rule $r^{\hat{\theta} X^{\leq}}_{X^{\leq}}$ matching y and having decision part "*then* $y \in X^{\leq}_t$", a value of

$score_{r^{\hat{\theta}X^{\leq}}_{X^{\leq}}}(X_i, y)$ is calculated for object y and each decision class X_i, such that $i \leq t$:

$$score_{r^{\hat{\theta}X^{\leq}}_{X^{\leq}}}(X_i, y) = \frac{\left|\|\Phi_{r^{\hat{\theta}X^{\leq}}_{X^{\leq}}}\| \cap X_i\right|^2}{\left|\|\Phi_{r^{\hat{\theta}X^{\leq}}_{X^{\leq}}}\|\right|\left|X_i\right|}. \tag{5.3}$$

The value of above defined score coefficient can be interpreted as a product of credibility cr_r and relative strength rs_r of rule r covering object y with respect to decision class X_i, since:

$$cr_r(X_i, y) = \frac{\left|\|\Phi_r\| \cap X_i\right|}{\left|\|\Phi_r\|\right|}, \tag{5.4}$$

$$rs_r(X_i, y) = \frac{\left|\|\Phi_r\| \cap X_i\right|}{\left|X_i\right|}. \tag{5.5}$$

Thus, $score_{r^{\hat{\theta}X^{\geq}}_{X^{\geq}}}(X_i, y) = cr_{r^{\hat{\theta}X^{\geq}}_{X^{\geq}}}(X_i, y) \times rs_{r^{\hat{\theta}X^{\geq}}_{X^{\geq}}}(X_i, y)$. Analogously,
$score_{r^{\hat{\theta}X^{\leq}}_{X^{\leq}}}(X_i, y) = cr_{r^{\hat{\theta}X^{\leq}}_{X^{\leq}}}(X_i, y) \times rs_{r^{\hat{\theta}X^{\leq}}_{X^{\leq}}}(X_i, y)$.
Moreover, the value of the score coefficient can be interpreted as a measure of relevance between condition part of rule r covering object y and class X_i. Using frequentist estimators of probabilities, one can also express the score coefficient as a product of two conditional probabilities:

$$Pr(X_i\big|\|\Phi_r\|) = \frac{\left|\|\Phi_r\| \cap X_i\right|}{\left|\|\Phi_r\|\right|}, \tag{5.6}$$

$$Pr(\|\Phi_r\|\big|X_i) = \frac{\left|\|\Phi_r\| \cap X_i\right|}{\left|X_i\right|}, \tag{5.7}$$

that is, the larger the product of the two probabilities, the stronger is the relevance between $\|\Phi_r\|$ and X_i.

Finally, object y is assigned to the class X_i for which the value of the score coefficient is the greatest. The value of $score_{r^{\hat{\theta}X^{\geq}}_{X^{\geq}}}(X_i, y) \in [0, 1]$ and the value of $score_{r^{\hat{\theta}X^{\leq}}_{X^{\leq}}}(X_i, y) \in [0, 1]$. Thus, it can be interpreted as a degree of certainty of the assignment of y to X_i.

Situation 3; in this case, set $cov(y)$ of decision rules assigning object y to different unions of decision classes is taken into account. Remark that any object y from the learning data set used to induce rules can support many rules suggesting different unions, even if y creates no ambiguity with other objects from the learning data set. For example, object $y \in X_r$, can support rule $r^{\hat{\theta}X^{\geq}}_{X^{\geq}}$ whose decision part is "*then* $y \in X^{\geq}_t$" with $t \leq p$ or decision rule $r^{\hat{\theta}X^{\leq}}_{X^{\leq}}$ whose decision parts is "*then* $y \in X^{\leq}_s$" with $s \geq p$. Analogously, when we pass from the rule induction to the application of decision rules for classification,

object y covered by a rule suggesting assignment to the union of decision classes $X_t^{\geq}$ may be covered also by rules indicating unions of decision classes $X_s^{\geq}$, where $s < t$, since $X_t^{\geq} \subseteq X_s^{\geq}$. Of course, object y can be covered also by a rule suggesting assignment to the union of decision classes $X_v^{\leq}$: in this case; object y may be covered also by rules indicating unions of decision classes $X_w^{\leq}$, where $w > v$, since $X_v^{\leq} \subseteq X_w^{\leq}$.

- *Standard classification method*: it compiles decisions suggested by rules from $cov(y)$ in two steps. First, an intersection of unions suggested by all decision rules $r_{X^{\geq}}^{\hat{\theta}X^{\geq}}$ (i.e., decision rules having decision part "*then* $y \in X^{\geq}$") covering object y is calculated. The lowest class from this intersection, say X_t, constitutes the first limit of final assignment. Second, an intersection of unions suggested by all by all decision rules $r_{X^{\leq}}^{\hat{\theta}X^{\leq}}$ (i.e., decision rules having decision part "*then* $y \in X^{\leq}$") covering object y is calculated. The highest class from this intersection, say X_s, constitutes the second limit of final assignment. The recommended final assignment of y is the interval of decision classes from X_t to X_s (i.e., at least X_t and at most X_s). If $t = s$, then the assignment is univocal, otherwise, one of two cases may occur:

  1) $t < s$, then object y is assigned to classes X_t, $X_{t+1}, \ldots, X_{s-1}, X_s$, without possibility of refinement because of imprecise information,

  2) $t > s$, then object y is assigned to classes X_s, $X_{s+1}, \ldots, X_{t-1}, X_t$, without possibility of discernment because of contradictory information.

  In case (1), the information is imprecise because the classification regards a family of classes, from X_t to X_s, but there is not enough information for a finer specification. In case (2), the information is contradictory because suggestions from $r_{X^{\geq}}^{\hat{\theta}X^{\geq}}$ decision rules and $r_{X^{\leq}}^{\hat{\theta}X^{\leq}}$ decision rules are conflicting. In fact, $r_{X^{\geq}}^{\hat{\theta}X^{\geq}}$ decision rules suggest at least X_t, while $r_{X^{\leq}}^{\hat{\theta}X^{\leq}}$ decision rules suggest at most X_s, but $t > s$. For example, in a classification problem with three classes, $X_1 =$ "*bad*", $X_2 =$ "*medium*", $X_3 =$ "*good*", this is the case in which the suggestion is that object y is at least good (i.e., good or better) but also at most bad (i.e., bad or worse). Reasonably, in this case the conclusion would be that object y is good, medium or bad.

- *New classification method*: score coefficient $score_{cov(y)}(X_t, y)$ is calculated with respect to each class X_t and set $cov(y)$. Object y is assigned to the class with the highest value of score coefficient.

First, let us distinguish in the set of decision rules $cov(y)$ those rules that are concordant with assignment of y to class X_t. These are decision rules $r_i \in cov(y)$, $i = 1, 2, \ldots, k$, that suggest assignment of y to the union of classes $X_s^{\geq}$ and $X_q^{\leq}$, where $X_t \subseteq X_s^{\geq}$ and $X_t \subseteq X_q^{\leq}$, respectively. For, object $y \in U$, set of rules $cov(y)$ and class X_t let us define the set of rules supporting assignment of object y to class X_t as:

$$cov_{X_t}^{+}(y) = \{r \in cov(y) : X_t \in \Psi_r\}, \tag{5.8}$$

where Ψ_r denotes decision part of rule r. Analogously, let us define the set of rules that are not supporting assignment of object y to class X_t. These are decision rules $r_i \in cov(y)$, $i = k+1, \ldots, h$, that suggest assignment of object y to the union of classes $X_s^{\geq}$ and $X_q^{\leq}$ such that $X_t \cap X_s^{\geq} = \emptyset$ and $X_t \cap X_q^{\leq} = \emptyset$, respectively. For, object $y \in U$, set of rules $cov(y)$ and class X_t let us define the set of rules not supporting assignment of object y to class X_t as:

$$cov_{X_t}^{-}(y) = \{r \in cov(y) : X_t \notin \Psi_r\}. \tag{5.9}$$

We define positive score coefficient with respect assignment of object y to class X_t on the basis of rules belonging to $cov_{X_t}^{+}(y)$, as follows:

$$score_{cov_{X_t}^{+}(y)}(X_t, y) = \frac{\left|(\|\Phi_{r_1}\| \cap X_t) \cup \ldots \cup (\|\Phi_{r_k}\| \cap X_t)\right|^2}{\left(\left|\|\Phi_{r_1}\| \cap \|\Phi_{r_k}\|\right)\right||X_t|}, \tag{5.10}$$

where $\|\Phi_{r_1}\|, \ldots, \|\Phi_{r_k}\|$ are the sets of objects verifying condition parts of rules $r_i \in cov_{X_t}^{+}(y)$, $i = 1, 2, \ldots, k$. Positive score coefficient $score_{cov_{X_t}^{+}(y)}(X_t, y)$ takes into account decision rules which are concordant with assignment of y to class X_t. The interpretation of $score_{cov_{X_t}^{+}(y)}(X_t, y)$ is analogous to the interpretation of the score coefficient defined in situation 2.

We define negative score coefficient with respect to assignment of object y to class X_t on the basis of rules belonging to $cov_{X_t}^{-}(y)$, as follows:

$$score_{cov_{X_t}^{-}(y)}(X_t, y) =$$
$$= \frac{\left|(\|\Phi_{k+1}\| \cap X_{t-1}^{\leq}) \cup \ldots \cup (\|\Phi_l\| \cap X_{t-1}^{\leq}) \cup (\|\Phi_{l+1}\| \cap X_{t+1}^{\geq}) \cup \ldots \cup (\|\Phi_h\| \cap X_{t+1}^{\geq})\right|^2}{\left|\|\Phi_{k+1}\| \cup \ldots \cup \|\Phi_l\| \cup \|\Phi_{l+1}\| \cup \ldots \cup \|\Phi_h\|\right||X_{t-1}^{\leq} \cup X_{t+1}^{\geq}|}, \tag{5.11}$$

where $X_{t-1}^{\leq}$ and $X_{t+1}^{\geq}$ are downward union and upward union of classes that do not include class X_t. In case of $t = 1$, all parts of equation (5.11) that involve $X_{t-1}^{\leq}$ are neglected. Analogously, in case of $t = n$, all parts of equation (5.11) that involve $X_{t+1}^{\geq}$ are neglected.

In definition (5.11) we assume that decision rules in set $cov^{-}_{X_t}(y)$ are ordered so that all r_i for $i = k+1, \ldots, l$ are assigning to subset of $X^{\leq}_{t-1}$, while all r_j for $j = l+1, \ldots, h$ are assigning to subset of $X^{\geq}_{t+1}$.

Negative score coefficient $score_{cov^{-}_{X_t}(y)}(X_t, y)$ can be interpreted as a product of credibility and relative strength of all rules matching object y, and suggesting its assignment to decision classes different than X_t. It can also be interpreted as a measure of relevance between condition parts of rules belonging to $cov^{-}_{X_t}(y)$ and $\neg X_t$.

The recommended final assignment of object y is calculated on the basis of score coefficient that involves both positive and negative score coefficients. For object y and class X_t this score coefficient is defined as:

$$score_{cov(y)}(X_t, y) = score_{cov^{+}_{X_t}(y)}(X_t, y) - score_{cov^{-}_{X_t}(y)}(X_t, y). \tag{5.12}$$

Analogously to situation 2, object y is assigned to the class X_t for which the value of $score_{cov(y)}(X_t, y)$ is the highest. In this situation, score coefficient $score_{cov(y)}(X_t, y)$ can be interpreted as a net balance of arguments in favor and arguments against the conclusion "object y belongs to class X_t".

As follows from the above description, the new classification method takes into account a joint strength of covering rules with respect to each particular class. This strength is calculated considering the rules suggesting an assignment to a given class X_t as arguments in favor of X_t, and all other matching rules as arguments against X_t. The standard classification method is not using information about the strength of matching rules and, instead, recommends an assignment based on intersection of suggested unions of decision classes.

The new classification method may give different results to the standard classification method. This is the consequence of the different type of information taken into account by the two methods. The following Example 5.2.1 illustrates the difference in result of the two classification methods.

Example 5.2.1. *Let us consider two rules $\{r_1, r_2\}$ covering object y (i.e., $cov(y) = \{r_1, r_2\}$). Rule r_1 assigns to union of classes $X^{\geq}_2$ while rule r_2 assigns to union of classes $X^{\geq}_3$.*

- Standard classification method*: the result of compilation of decisions suggested by covering rules is class X_3.*

- New classification method*: score coefficient* $score_{cov(y)}(X_t, y)$ *is calculated with respect to each class* X_t *and the set of rules* $cov(y)$*. First, we identify rules that support assignment to class* X_2 *or* X_3 *and rules that are against assignment to these classes:*

$$cov^{+}_{X_2}(y) = \{r_1\}, \qquad cov^{-}_{X_2}(y) = \{r_2\},$$
$$cov^{+}_{X_3}(y) = \{r_1, r_2\}, \qquad cov^{-}_{X_3}(y) = \emptyset.$$

Then we calculate the score coefficient for class X_2 *and* X_3*:*

$$score_{cov(y)}(X_2, y) = \frac{\left|(\|\Phi_{r_1}\| \cap X_2)\right|^2}{\left|\|\Phi_{r_1}\|\right||X_2|} - \frac{\left|(\|\Phi_{r_2}\| \cap X_3)\right|^2}{\left|\|\Phi_{r_2}\|\right||X_3|},$$

$$score_{cov(y)}(X_3, y) = \frac{\left|(\|\Phi_{r_1}\| \cap X_3) \cup (\|\Phi_{r_2}\| \cap X_3)\right|^2}{\left|\|\Phi_{r_1}\| \cup \|\Phi_{r_2}\|\right||X_3|}.$$

The result of classification depends on value of $score_{cov(y)}(X_2, y)$ *and* $score_{cov(y)}(X_3, y)$ *coefficients. Object* y *may be assigned to class* X_2 *if relevance between rule* r_1 *and class* X_2 *is high and relevance between rule* r_2 *and class* X_3 *is low. It is assigned to class* X_3 *otherwise. The first situation occurs for example when* $|X_2|$ *and* $|X_3|$ *are equal,* $|\|\Phi_{r_1}\||$ *is high,* $|\|\Phi_{r_2}\||$ *is low,* $\left|(\|\Phi_{r_1}\| \cap X_2)\right|$ *is high,* $\left|(\|\Phi_{r_1}\| \cap X_3)\right|$ *and* $\left|(\|\Phi_{r_2}\| \cap X_3)\right|$ *are low.*

5.3 Combination of responses in an ensemble of classifiers

In this section, we show how to combine the results of classification, i.e., assignments of object to class (or classes), produced in an ensemble by component classifiers that use both types of classification methods presented in the previous section 5.2. The reason for combination of results of classifications produced by component classifiers in an ensemble is a potential improvement of predictive accuracy of the ensemble. An important property of ensembles that show this type of improvement over its component classifiers is diversity of the component classifiers (see e.g., (Breiman, 1996; Kittler et al., 1998; Kuncheva et al., 2002; Kuncheva, 2004)). In case of both types of classifiers considered here, we use rule models that are known to give unstable classifiers (i.e., small change in the learning set may lead to significantly different set of rules). Unstable component classifiers result in ensembles that are diversified.

The choice of method that combines the results of classification in the ensembles depends on the type of these classification results and on the classification problem that

is solved. When the classification results always indicate single class, in case of non-ordinal classification problem, majority voting is the method of combining the results in the ensemble (Franke and Mandler, 1992; Breiman, 1996). This choice may be attributed to the fact that mode is the measure of central tendency for non-ordinal nominal scale. Thus, majority voting minimizes the number of misclassifications of the ensemble. On the other hand, in case of ordinal classification problem, median of the results is the natural choice. This choice may be attributed to the fact that median is the measure of central tendency for ordinal scales. Median does not depend on the distance between values of the decision attribute, so the scale of the decision attribute does not matter, only the order is taken into account. It minimizes the difference of ranks of the class to which the classified object belongs and to which it is classified. Note, however, that in some applications of ordinal classification, low misclassification rate may be more important than small errors. In such applications, majority voting may be a better choice than median voting.

In our case, the classification result issued by a component classifier that applies the standard classification method (see section 5.2 for details) is class or set of contiguous classes. This type of classifier does not give any additional information that reflects certainty (or consistency) of the classification result. We can treat such classification results, i.e., the suggestions of assignment, as votes. Votes of all component classifiers are equally important. Each component classifier in ensemble is voting for one class or set of contiguous classes according to the classification result produced by the classification method:

1) If j-th component classifier ($j = 1, \ldots, m$) is voting to assign object y to only one class X_i, we can denote it as $vote_j(X_i, y) = 1$. In this case, for any $k \neq i$, $vote_j(X_k, y) = 0$.

2) If result of classification suggested by j-th component classifier ($j = 1, \ldots, m$) for object y is a set of contiguous classes $X_k, X_{k+1}, \ldots, X_l$, (see situation 3 in section 5.2)) then the vote of this component classifier is divided equally (e.g., $1/(l-k)$) between suggested classes. We can denote it as $vote_j(X_i, y) = \frac{1}{l-k}$, for $i \in [k, l]$. Analogically, $vote_j(X_i, y) = 0$, for $i \notin [k, l]$

In case of results issued by component classifiers applying the new classification method, we can treat the value of score coefficient (5.12) as the value of vote. In this case, each of component classifiers issues a distribution of score coefficients that reflect relevance between the classifying rules and the classes. For j-th component classifier

assigning object y, class X_i, $i = 1, \ldots, n$, and $cov_j(y)$, $j = 1, \ldots, m$, being the set of rules in j-th classifier that covers object y, we can denote it as: $vote_j(X_i, y) = score_{cov_j(y)}(X_i, y)$.

The following aggregation rules can be applied to results of classification represented by votes:

1) **Max vote rule**

$$assign \quad y \rightarrow X_i \quad if \quad X_i = \arg\max_{i=1}^{n} \max_{j=1}^{m} \big(vote_j(X_i, y)\big),$$

2) **Majority vote rule**

$$assign \quad y \rightarrow X_i \quad if \quad X_i = \arg\max_{i=1}^{n} \big(\frac{1}{m} \sum_{j=1}^{m} vote_j(X_i, y)\big),$$

3) **Median vote rule**

$$assign \quad y \rightarrow X_i \quad if \quad X_i = \arg\operatorname{med}_{i=1}^{n} \big(\sum_{j=1}^{m} vote_j(X_i, y)\big).$$

As it is shown in (Kittler et al., 1998), the following relationship holds:

$$\max_{i=1}^{n} \big(\frac{1}{m} \sum_{j=1}^{m} vote_j(X_i, y)\big) \leq \max_{i=1}^{n} \max_{j=1}^{m} \big(vote_j(X_i, y)\big). \tag{5.13}$$

The relationship (5.13) reflects that max vote rule can be approximated by majority vote rule as the lower bound. Difference between application of majority voting and median voting is considered earlier in this section.

A component classifier may be abstaining according to definition of situation 1 in section 5.2. Other researchers considered solutions when a classifier in ensemble may not produce class prediction in case of uncertainty of the objects' classification. However, most of the research in this area concerns refraining from the final decision in case of disagreement between votes of component classifiers, e.g., see a study (Rückert and Kramer, 2004) showing that it may improve the final accuracy. Some researches allow single classifiers to give no answer. For instance, rule ensembles, like SLIPPER (Cohen

and Singer, 1999) are based on a weighted combination of *single rules* (being component classifiers) and a rule is excluded from voting if the new object is not covered by it. According to our best knowledge other researchers have not considered abstaining solutions for ensembles where component classifiers are based on sets of unordered rules induced by sequential covering algorithms which are the most popular techniques for inducing rules (Kononenko and Kukar, 2007). Let us remind that we presented sequential covering rule induction algorithm VC-DomLEM in section 4.4. VC-DomLEM induces sets of rules applied by classification methods presented in this chapter.

A set of unordered rules usually covers a subspace in the problem space which can be seen as an area of its expertise. Thus, in a diversified ensemble of rule classifiers, it is likely that if one of component classifiers abstains from classifying an object, other more experienced classifiers that use other sets of rules may classify the object. They should make it better than the classifier that is forced to suggest assignment of object not belonging to his area of expertise. We have shown in (Błaszczyński et al., 2009e) that abstaining strategy gives more accurate ensembles of rule set classifiers than those ensembles that applied partial matching strategy.

5.4 Summary

In this chapter, we presented two classification methods for DRSA and VC-DRSA. These methods resolve situations that occur when the considered object is covered by none of rules, one rule, or multiple rules from the set of rules.

Moreover, these classification methods were investigated with regard to two types of classifiers, namely single classifier and ensemble of classifiers. We have presented methods to combine results of component classifiers in an ensemble of classifiers.

CHAPTER 6

Computational Experiments

6.1 Experimental Setup

In this chapter, we check the predictive accuracy of the variable consistency rough set approach in ordinal classification with monotonicity constraints. To this end, we compare our methods to other classifiers. In general, it is not always the case that ordinal classifiers that preserve monotonicity constraints perform better in terms of the predictive accuracy than non-ordinal classifiers. This is mainly attributed to the fact that monotonicity constraints, that need to be satisfied, bias the classifier. Taking this into account, we included in comparison some well known non-ordinal classifiers. The results are analyzed with application of nonparametric statistical tests.

Our experiment was conducted on several real data sets. We restricted these data sets to only those which are known to be ordinal and that include monotone relationships between values of decision attribute and some of the other attributes.

The experiment has been divided into two parts: in the first part single classifiers, and in the second, ensembles of classifiers, are compared.

According to our motivation concerning application of the methods to decision aiding, we also consider interpretability and traceability of the compared classification models.

6.1.1 Data sets

We included in the experiment fourteen data sets that were known to be ordinal and that include monotonic relationships between values of the decision attribute and some

of the other attributes. The directions of ordering in the domains of the attributes are part of the domain knowledge about the considered classification problems.

Six data sets that we analyzed come from the UCI repository[1]. These are:

- `balance` scale data set,
- `breast-c`: breast cancer Ljubljana data set,
- `breast-w`: breast cancer Wisconsin data set,
- `car` evaluation data set,
- `cpu` performance data set, in case of which the decision attribute was discretized into four classes, containing equal number of objects,
- `housing` data set concerns housing values in suburbs of Boston; decision attribute is discretized into four classes containing equal number of objects as in (Feelders and Pardoel, 2003).

Two data sets concerning credit rating and credit risk assessment problems were taken from Doumpos and Zopounidis (Doumpos and Pasiouras, 2005; Marinakis et al., 2008):

- `bank-g`: bank of Greece data set,
- `fame`: financial analysis made easy data set that contains Bureau van Dijk's company database.

Two additional data sets concern house pricing problem:

- `denbosch` data set that contains housing values from Dutch city Den Bosch; see (Daniëls and Kamp, 1999) for details,
- `windsor` data set that concerns housing values in Windsor, Canada (Koop, 2000); decision attribute is discretized into four classes containing equal number of objects as in (Feelders and Pardoel, 2003).

Finally, four data sets were taken from Ben-David (Ben-David, 1992, 1995):

[1] see http://www.ics.uci.edu/~mlearn/MLRepository.html

- `ERA`: employee rejection/acceptance data set; aiming at determining the most important qualities of potential employees,
- `ESL`: employee selection data set; applications for industrial jobs,
- `LEV`: lectures evaluation data set; anonymous evaluations of lecturers in MBA courses,
- `SWD`: social workers decisions data set; assessments of qualified social workers with regard to the risk that children face if they stay with their families at home.

Data sets `breast-c` and `breast-w` contained a marginal number of missing values which were substituted by the central values of the respective attribute.

Characteristics of these fourteen data sets are given in Table 6.1. In this table, we also show the values of λ (3.47) and δ (3.49), calculated on the whole data sets. For both measures, we present values for the most restrictive consistency thresholds (i.e., $\boldsymbol{\mu}_X^{'*} = \mathbf{1}$, $\boldsymbol{\epsilon}_X^* = \mathbf{0}$), and values calculated for the consistency thresholds used in the further parts of the experiment. These measurements show the predictive accuracy that is attainable by a rough set classifier on a given data set, which can also be interpreted as the consistency of the data set.

Basing on these measures, we can observe that we have three fully consistent data sets among all presented in the table. These are: `balance`, `cpu`, and `housing`. Then, we can distinguish four data sets that have high consistency: `brest-w`, `car`, `bank-g`, and `fame`. Also not bad in terms of consistency are: `denbosch` and `ESL`. While data sets: `breast-c`, `ERA`, `LEV`, `SWD`, and `windsor` are highly inconsistent. We can also observe that application of VC-DRSA leads to considerable improvement of both measures for inconsistent data sets. This means that VC-DRSA allowed to include fair amount of inconsistent objects into extended lower approximations. The values of λ and δ measures will be further compared to the percentage of correctly classified objects and mean absolute error (MAE) obtained in 10-fold cross validation by the methods taking part in the experiment.

Table 6.1: Characteristics of data sets, values of λ and δ measures for $\boldsymbol{\theta}^*_X = \boldsymbol{\mu}'^*_X = \mathbf{1}$, $\boldsymbol{\theta}^*_X = \boldsymbol{\epsilon}^*_X = \mathbf{0}$, as well as for $\boldsymbol{\mu}'_X$ and $\boldsymbol{\epsilon}_X$ used to obtain results shown in Tables 6.2 and 6.3

Id	Data set	#Obj.	#Attr.	#Class.	$\lambda_P^{\boldsymbol{\theta}^*_X}$	$\delta_P^{\boldsymbol{\theta}^*_X}$	$\boldsymbol{\mu}'_X$	$\lambda_P^{\boldsymbol{\mu}'_X}$	$\delta_P^{\boldsymbol{\mu}'_X}$	$\boldsymbol{\epsilon}_X$	$\lambda_P^{\boldsymbol{\epsilon}_X}$	$\delta_P^{\boldsymbol{\epsilon}_X}$
1	balance	625	4	3	100	0	1	100	0	0	100	0
2	breast-c	286	7	2	23.78	0.7622	0.55	90.21	0.0979	0.45	98.60	0.014
3	breast-w	699	9	2	97.57	0.0243	0.95	100	0	0.001	97.57	0.0243
4	car	1296	6	4	98.61	0.0162	0.85	100	0	0.01	99.46	0.0054
5	cpu	209	6	4	100	0	1	100	0	0	100	0
6	bank-g	1411	16	2	98.02	0.0198	0.99	98.72	0.0128	0.001	98.87	0.0113
7	fame	1328	10	5	98.27	0.0211	0.6	100	0	0.001	99.17	0.0113
8	denbosch	119	8	2	89.92	0.1008	0.9	100	0	0.05	99.16	0.0084
9	ERA	1000	4	9	11.3	2.826	0.75	87.3	0.129	0.03	80.8	0.28
10	ESL	488	4	9	85.04	0.1578	0.95	98.98	0.0102	0.03	100	0
11	housing	506	13	4	100	0	1	100	0	0	100	0
12	LEV	1000	4	5	41.20	0.8010	0.9	88.7	0.113	0.03	97.7	0.023
13	SWD	1000	10	4	48.7	0.68	0.85	80.4	0.196	0.15	100	0
14	windsor	546	10	4	69.6	0.40664	0.9	80.04	0.1996	0.05	97.44	0.0256

6.1.2 Methods

In the fist part of the experiment, we considered single classifiers. In this part of experiment, we compared eight classifiers, among which two are our proposals described in this thesis. These were two variants of VC-DomLEM, implemented in jRS library[2]:

- monotonic VC-DomLEM (i.e., with monotonic ϵ-consistency or ϵ'-consistency measure) with the standard classification method and the new classification method,
- non-monotonic VC-DomLEM (i.e., with non-monotonic μ-consistency measure) with the standard classification method and the new classification method.

Moreover, we used two ordinal classifiers that preserve monotonicity constraints, namely:

- Ordinal Learning Model (OLM) (Ben-David et al., 1989; Ben-David and Jagerman, 1997),
- Ordinal Stochastic Dominance Learner (OSDL) (Cao-Van, 2003) in balanced version, which gives better results on inconsistent data (and which is similar with respect to treatment of inconsistencies to the approach presented in this thesis).

Brief description of these methods can be also found in section 1.2.2. In case of both classifiers we used implementation obtained from WEKA (Hall et al., 2009).

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

As it was mentioned, we decided to compare ordinal classifiers that are required to preserve monotonicity constraints to some well known non-ordinal classifiers, like:

- Naïve Bayes,
- Support Vector Machine (SVM) with linear kernel (Platt, 1998),
- decision rule classifier RIPPER (Cohen, 1995),
- decision tree classifier C4.5 (Quinlan, 1992).

These classifiers were used also in implementation obtained from WEKA (Hall et al., 2009).

We compared single classifiers in terms of their predictive accuracy.

In the second part of the experiment, we compared ensembles of classifiers. In this case, we compared standard bagging with proposed by us variable consistency bagging (VC-bagging), more precisely:

- bagging in standard setting (Breiman, 1996), with the number of component classifiers $m = 20$, and and median vote rule in aggregation,
- VC-bagging with ϵ-consistency measure calculated on random subsets of attributes with 50% cardinality, the number of component classifiers $m = 20$, and median vote rule in aggregation.

Two versions of monotonic VC-DomLEM were used in this comparison as component classifiers in standard bagging and VC-bagging:

- monotonic VC-DomLEM with the standard classification method,
- random monotonic VC-DomLEM with the standard classification method.

In this part of the experiment, we focused not only on comparison of the predictive accuracy between classifiers. We also investigated the sources of improvements of predictive accuracy that is achieved by these ensembles. Moreover, we compared the results of our ensembles with the best results from the literature.

6.2 Results of experiments

statistical comparison

We used a statistical approach to verify significance of differences in predictive accuracy between classifiers in variants which we mentioned in section 6.1.2. First, we applied non-parametric Friedman test to globally compare performance of the classifiers on multiple data sets (Demsar, 2006; Kononenko and Kukar, 2007). The null-hypothesis in this test is that all compared classifiers perform equally well. It was tested using the ranks of each of the classifiers on each of the data sets. The Friedman test is defined as follows:

Friedman test

Let r_i^j be the rank of the j-th classifier of m classifiers on the i-th of n data sets. The smaller the rank the better the classifier. The Friedman test compares the average ranks of classifiers, $R_j = \frac{1}{n}\sum_i r_i^j$. Under the null-hypothesis, which states that average ranks R_j of all compared classifiers are equal, the Friedman statistic

$$\chi_F^2 = \frac{12 \times n}{m \times (m+1)} \left[\sum_j R_j^2 - \frac{m \times (m+1)^2}{4} \right] \tag{6.1}$$

is distributed according to χ_F^2 with $m-1$ degrees of freedom, when n and m are sufficiently big (usually, $n > 10$ and $m > 5$). According to suggestion from (Demsar, 2006), Friedman statistics is undesirably conservative and so it is better to apply Iman and Davenport statistic (which is relying on Friedman χ_F^2)

$$F_F = \frac{(n-1) \times \chi_F^2}{n \times (m-1) - \chi_F^2}, \tag{6.2}$$

which is distributed according to the F-distribution with $m-1$ and $(m-1)(n-1)$ degrees of freedom.

We did not present complete post-hoc analysis (Demsar, 2006) of differences between classifiers. We decided to rely on direct comparison of ranks of each classifier averaged over all data sets and on comparison of pairs of classifiers. We continued our experimental comparison with examination of significance of difference in predictive accuracy for each pair of classifiers. We applied Wilcoxon signed rank test (Demsar, 2006; Kononenko and Kukar, 2007) 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. Wilcoxon test is defined as follows:

Wilcoxon test

Let d_i denote the difference between the performance values of the two classifiers on the i-th out of n data sets. The differences are ranked according to their absolute values. Average ranks are assigned in case of ties between absolute values. Then, let R^+ denote the sum of ranks for the data sets on which the first classifier outperformed the

second. Let R^- denote the sum of ranks for the data set on which it was the opposite. Ranks of $d_i = 0$ are split evenly among the sums (if there is an odd number of them, one is ignored. The value of T is the smaller of the sums, $T = min(R^+, R^-)$. Critical values for T can be checked in tables. For a lager number of data sets, the statistics

$$z = \frac{T - \frac{1}{4} \times n \times (n+1)}{\sqrt{\frac{1}{24} \times n \times (n+1) \times (2n+1)}} \tag{6.3}$$

is distributed approximately normally.

estimation of predictive accuracy

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. We also measured the percentage of correctly classified objects. We present the values of these measures in two separate tables. In both cases, the 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 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). Last row of each table shows the average rank obtained by a given classifier. 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.

6.2.1 Single classifiers

predictive accuracy

The results of the experiment which concerned comparison of predictive accuracy, performed for single classifiers are shown in Tables 6.2 and 6.3. The first table contains values of mean absolute error while the second table contains the percentages of correctly classified objects.

We analyzed the ranks of MAE, which are presented in Table 6.2. We begun with Friedman test, which in this case checks whether the measured average ranks are significantly different from the expected under the null-hypothesis mean average rank equal 4.5. In the test, performed for this comparison, we have $F_F = 5.28$, while critical value for $\alpha = 0.05$ is 2.11. The p-value in this test is lower than 0.0001. Then, we analyzed ranks of percentage of correctly classified objects, which are presented in Table 6.3. In this test, we have $F_F = 4.93$. The p-value in Friedman test is in this case is also lower than 0.0001. The results of Friedman test and observed differences in average ranks allow us to state with high confidence that there is a significant difference between compared classifiers.

Table 6.2: Single classifiers - mean absolute error (MAE) results

Id	monotonic VC-DomLEM	non-monotonic VC-DomLEM	Naïve Bayes	SVM	RIPPER	C4.5	OLM	OSDL
1	0.1621 (2) ±0.001996	0.1659 (3) ±0.002719	**0.1104** (1) ±0.002613	0.1723 (4) ±0.003017	0.2917 (5) ±0.01088	0.3088 (6) ±0.02174	0.6384 (7) ±0.01713	0.7003 (8) ±0.004588
2	**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
3	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
4	**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
5	**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
6	**0.04536** (1) ±0.001531	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
7	**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
8	**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
9	1.307 (2) ±0.002055	1.364 (7) ±0.006018	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
10	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
11	0.3235 (2) ±0.01133	**0.3083** (1) ±0.00559	0.5033 (7) ±0.006521	0.3551 (3) ±0.005187	0.3676 (4) ±0.007395	0.3676 (5) ±0.01556	0.3867 (6) ±0.01050	1.078 (8) ±0.00796
12	0.4813 (6) ±0.004028	0.5187 (7) ±0.002867	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
13	0.454 (4) ±0.004320	0.4857 (7) ±0.005249	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
14	**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	3.82	4.54	4.54	5.29	4.71	6.71	4.36

Thus, we continued with Wilcoxon test on MAE values from Table 6.2. We can observe significant difference (p-values lower than 0.05) between monotonic VC-DomLEM and any other classifier except OSDL (p-value in this case is 0.078). The same is true for the following pairs: non-monotonic VC-DomLEM and OLM, non-monotonic VC-DomLEM and RIPPER, Naïve Bayes and OLM, SVM and OLM, C4.5 and OLM. Then, we applied Wilcoxon test to percentage of correctly classified objects from Table 6.3. These results indicate significant differences between monotonic VC-DomLEM and any other classifier. The same is true for following pairs: non-monotonic VC-DomLEM and OLM, Naïve Bayes and OLM, RIPPER and OLM, C4.5 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 (but only in case of MAE). On the other hand, non-monotonic VC-DomLEM is comparable to other classifiers except OLM. OLM is clearly

Table 6.3: Single classifiers - percentage of correctly classified objects results

Id	monotonic VC-DomLEM	non-monotonic VC-DomLEM	Naïve Bayes	SVM	RIPPER	C4.5	OLM	OSDL
1	86.61 (4) ±0.5891	86.93 (3) ±0.3771	**90.56** (1) ±0.1306	87.47 (2) ±0.1508	81.5 (5) ±0.5439	78.45 (6) ±0.7195	61.28 (7) ±1.287	57.81 (8) ±0.3288
2	**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
3	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
4	**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
5	**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
6	**95.46** (1) ±0.1531	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
7	**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
8	**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
9	26.9 (2) ±0.3742	22.17 (7) ±0.1247	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
10	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
11	**72** (1) ±0.6521	**71.61** (2) ±0.09316	59.03 (7) ±0.3727	69.24 (3) ±0.4061	67.59 (6) ±0.9815	68.12 (4) ±1.037	67.65 (5) ±0.796	27.14 (8) ±0.3359
12	55.63 (6) ±0.3771	52.73 (7) ±0.1700	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
13	56.43 (6) ±0.4643	52.8 (7) ±0.4320	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
14	54.58 (2) ±0.7913	53.05 (4) ±1.349	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
	2.29	4	4.25	4.46	5.36	4.29	6.86	4.5

the worst classifier in our experiment.

We also compared the values from Tables 6.2 and 6.3 to the values of δ and λ presented in Table 6.1. Remember that the first ones are estimated by averaged 10-fold cross validation, while the second ones are estimated on the whole data set. Nevertheless, we can observe some interesting relationships. Thresholds $\delta_P^{\mu'_x}$, $\delta_P^{\epsilon_x}$, $\lambda_P^{\mu'_x}$, and $\lambda_P^{\epsilon_x}$ are never reached during learning. This is not surprising since they are defined as limit values of what can be achieved in learning. The nine data sets that were distinguished by $\lambda_P^{\boldsymbol{\theta}^*_X}$ and $\delta_P^{\boldsymbol{\theta}^*_X}$ as at least not bad in terms of consistency, and thus, easier to learn, are also those on which classifiers obtained good predictive accuracy. Exception to this rule are data sets: `ESL`, `fame` and `housing`. This may be caused by the fact that these data sets are described by many attributes and/or classes. It is thus visible that measures λ and δ allowed to distinguish the data sets which are just hard to learn (`ESL`, `fame`, and `housing`) from those which are inconsistent and hard to learn (`breast-c`, `ERA`, `LEV`, `SWD`, `windsor`). It can be also seen that for the highly inconsistent data sets: `breast-c`, `ERA`, `LEV`, and `SWD`, all classifiers performed better than the values of $\lambda_P^{\boldsymbol{\theta}^*_X}$ and $\delta_P^{\boldsymbol{\theta}^*_X}$. The only exception is percentage of correctly classified objects obtained by OLM for data set `SWD`.

This indicates that the classifiers were able to overcome the inconsistencies present in the highly inconsistent data sets.

Finally, we compared mean execution times of both versions of VC-DomLEM over all runs on the data sets. Induction of rules with monotonic VC-DomLEM was on average over three 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.

6.2.2 Ensembles of classifiers

predictive accuracy

We started this part of experiment with analysis of the predictive accuracy of the ensembles of classifiers. We compared standard version of bagging with variable consistency bagging (VC-bagging) on random subsets of attributes with 50% cardinality. The cardinality of the random subset of attributes was chosen according to the results of our previous experiments with this type of ensembles (Błaszczyński et al., 2009b). Naturally, to obtain better results, the cardinality of the random subset of attributes can be also tuned in experiments. We used ϵ-consistency measure in this type of ensemble because it has preferable properties and it is the same measure that is used by VC-DomLEM component classifiers (which is important for the interpretability of the ensemble, considered in section 6.3). In both cases median vote rule (see section 5.3) was used to aggregate the suggestions of component classifiers, which were monotonic VC-DomLEM and random monotonic VC-DomLEM. Random Monotonic VC-DomLEM is expected to perform worse than monotonic VC-DomLEM as a single classifier. That is why it was not considered in the experiment with single classifiers. On the other hand, application of this type of component classifier should result in more diverse ensembles than those composed of monotonic VC-DomLEM classifiers.

Differently, to the previous part of the experiment, we used the standard classification method in VC-DomLEM component classifiers. This choice was made due to computational complexity of the new classification method. The new classification method involves computation of intersections of sets of object covered by rules and sets of objects belonging to unions of classes. As it is described in chapter 5, the standard classification method may assign objects imprecisely or with contradictions. A diverse ensemble of classifiers that apply the standard classification method which are combined by median vote rule should be able to improve the final assignment in such situations. To show this effect we compare ensembles composed of classifiers that use the standard classification method to single classifiers that use the standard classification method and to single

classifiers that use the new classification method. In both cases, these single classifiers were constructed with the same parameters as the respective component classifiers in the ensembles.

The results of the experiment which concerned comparison of predictive accuracy, performed for ensembles of classifiers are shown in Tables 6.4 and 6.5. The first table contains values of mean absolute error while the second table contains the percentages of correctly classified objects. The columns in these tables corresponds to the succeeding classifiers:

- single monotonic VC-DomLEM classifier with the standard classification method,
- single monotonic VC-DomLEM classifier with the new classification method,
- single random monotonic VC-DomLEM classifier with the new classification method,
- single random monotonic VC-DomLEM classifier with the new classification method,
- standard bagging ensemble composed of monotonic VC-DomLEM component classifiers with the standard classification method,
- VC-bagging ensemble composed of monotonic VC-DomLEM component classifiers with the standard classification method,
- standard bagging ensemble composed of random monotonic VC-DomLEM component classifiers with the standard classification method,
- VC-bagging ensemble composed of random monotonic VC-DomLEM component classifiers with the standard classification method,

We analyzed the ranks of MAE from Table 6.4. In this case Friedman test, checks whether the measured average ranks are significantly different from the expected under the null-hypothesis mean average rank equal 4.5. In the test, performed to compare all classifiers in the table, we have $F_F = 4.23$, while critical value for $\alpha = 0.05$ is 2.11. The p-value in this test is close to 0.0004. Then, we analyzed ranks of percentage of correctly classified objects, which are presented in Table 6.5. In this test, we have $F_F = 3.82$. The p-value in Friedman test is in this case is close to 0.001. The results of Friedman test and observed differences in average ranks allow us to state with high confidence that there is a significant difference between compared classifiers.

Thus, w continued with Wilcoxon test on MAE values from Table 6.4. We can observe significant difference (p-values lower than 0.05) between VC-bagging with monotonic

Table 6.4: Mean absolute error (MAE) in repeated 10-fold cross validation of monotonic VC-DomLEM and random monotonic VC-DomLEM classifiers as single classifiers and as component classifiers in ensembles

Id	single std. class.	single new. class.	single random std. class.	single random new. class.	bagging	VC-bagging	bagging random	VC-bagging random
1	**0.1621** (2.5) ±0.001996	**0.1621** (2.5) ±0.001996	**0.1621** (2.5) ±0.001996	**0.1621** (2.5) ±0.001996	0.2011 (8) ±0.003771	0.1973 (7) ±0.01433	0.1872 (5) ±0.00471	0.1915 (6) ±0.001996
2	**0.2331** (1.5) ±0.003297	**0.2331** (1.5) ±0.003297	0.2401 (3.5) ±0.003297	0.2401 (3.5) ±0.003297	0.2448 (5) ±0.008565	0.2459 (6) ±0.008722	0.2809 (8) ±0.004361	0.2669 (7) ±0.001648
3	0.03815 (6) ±0.0006744	0.03720 (5) ±0.002023	0.0391 (7.5) ±0.002432	0.0391 (7.5) ±0.002432	0.03577 (4) ±0.001168	0.03243 (2) ±0.001349	0.03386 (3) ±0.001784	**0.02909** (1) ±0.002432
4	0.04090 (7.5) ±0.00126	**0.03421** (1.5) ±0.0007275	0.04090 (7.5) ±0.00126	**0.03421** (1.5) ±0.0007275	0.03652 (3) ±0.0007275	0.03832 (6) ±0.002623	0.03832 (5) ±0.002385	0.03781 (4) ±0.003274
5	0.1037 (7) ±0.01846	0.08293 (4) ±0.01479	0.1276 (8) ±0.01846	0.08612 (6) ±0.006767	0.08453 (5) ±0.005968	**0.07656** (1) ±0.003907	**0.07974** (2.5) ±0.002256	**0.07974** (2.5) ±0.008132
6	0.05481 (8) ±0.001456	0.04536 (5) ±0.001531	0.05268 (7) ±0.002191	0.04607 (6) ±0.001531	0.04489 (4) ±0.001205	**0.04158** (2) ±0.001205	0.04181 (3) ±0.001531	**0.04087** (1) ±0.000884
7	0.3803 (8) ±0.001627	0.3406 (6) ±0.001878	0.3785 (7) ±0.003095	0.3348 (5) ±0.007834	0.3230 (4) ±0.006419	**0.32** (3) ±0.007993	**0.32** (2) ±0.004032	**0.319** (1) ±0.003155
8	0.1261 (7) ±0.006861	0.1232 (6) ±0.01048	0.1232 (5) ±0.01048	0.1176 (3) ±0.006861	0.1289 (8) ±0.01048	**0.1092** (2) ±0.006861	0.1204 (4) ±0.01585	**0.1064** (1) ±0.01048
9	1.386 (4.5) ±0.003682	1.386 (4.5) ±0.003682	1.415 (6) ±0.01159	1.296 (3) ±0.01257	**1.263** (1) ±0.004497	1.271 (2) ±0.002625	1.704 (8) ±0.03583	1.656 (7) ±0.01302
10	0.4447 (7) ±0.01045	0.3702 (3) ±0.01352	0.5437 (8) ±0.007545	0.3893 (4) ±0.004427	0.3477 (2) ±0.006762	**0.3374** (1) ±0.004211	0.4201 (5) ±0.007293	0.4406 (6) ±0.008853
11	0.3564 (7) ±0.008887	0.3235 (5) ±0.01133	0.3979 (8) ±0.02075	0.3465 (6) ±0.007276	0.2984 (2) ±0.002795	**0.2793** (1) ±0.00796	0.3175 (4) ±0.006521	0.3142 (3) ±0.01130
12	0.4877 (4) ±0.004497	0.4813 (3) ±0.004028	0.5057 (6) ±0.005185	0.5033 (5) ±0.004989	0.4353 (2) ±0.001700	**0.409** (1) ±0.003742	0.5193 (7) ±0.004028	0.5297 (8) ±0.004643
13	0.462 (5) ±0.003742	0.454 (3) ±0.004320	0.5087 (6) ±0.002867	0.4587 (4) ±0.004497	**0.443** (2) ±0.003742	**0.4297** (1) ±0.002867	0.588 (8) ±0.009201	0.5123 (7) ±0.009104
14	0.5354 (8) ±0.008236	**0.5024** (1) ±0.006226	0.5305 (7) ±0.009137	0.5159 (4) ±0.004569	0.5299 (6) ±0.006743	**0.5043** (2) ±0.006044	0.5214 (5) ±0.01122	0.5116 (3) ±0.006226
	5.93	3.64	6.36	4.36	4	2.64	4.96	4.11

VC-DomLEM and any other classifier except VC-bagging and random monotonic VC-DomLEM (p-value in this case is 0.068). We can see similar dependencies for Wilcoxon test on percentage of correctly classified objects from Table 6.5. These results indicate significant differences between VC-bagging with monotonic VC-DomLEM and any other classifier with exception to single random monotonic VC-DomLEM (p-value in this case is 0.078) and VC-bagging with random monotonic VC-DomLEM (p-value in this case is 0.124).

To conclude this part of the experimental comparison, we can observe that standard bagging improves results of monotonic VC-DomLEM and random monotonic VC-DomLEM. However, in this case the difference in average ranks is not supported by results of conservative, non-parametric Wilcoxon test. More visible is that VC-bagging improves both monotonic VC-DomLEM and random monotonic VC-DomLEM. VC-bagging classifiers are better than single classifiers and standard bagging with the re-

Table 6.5: Percentage of correctly classified objects in repeated 10-fold cross validation of monotonic VC-DomLEM and random monotonic VC-DomLEM classifiers as single classifiers and as component classifiers in ensembles

Id	single std. class.	single new. class.	single random std. class.	single random new. class.	bagging	VC-bagging	bagging random	VC-bagging random
1	84.48 (4) ±0.2613	84.48 (4) ±0.2613	84.48 (4) ±0.2613	**86.99** (1) ±1.064	80.7 (8) ±0.2719	84 (6) ±0.5987	82.08 (7) ±0.3456	85.44 (2) ±0.5987
2	**76.69** (1.5) ±0.3297	**76.69** (1.5) ±0.3297	75.99 (3.5) ±0.3297	75.99 (3.5) ±0.3297	75.52 (5) ±0.8565	75.41 (6) ±0.8722	71.91 (8) ±0.4361	73.31 (7) ±0.1648
3	96.19 (6) ±0.06744	96.28 (5) ±0.2023	96.09 (7.5) ±0.2432	96.09 (7.5) ±0.2432	96.42 (4) ±0.1168	96.76 (2) ±0.1349	96.61 (3) ±0.1784	**97.1** (1) ±0.2432
4	97.02 (3.5) ±0.07275	**97.15** (1.5) ±0.063	97.02 (3.5) ±0.07275	**97.15** (1.5) ±0.063	97 (5.5) ±0.063	96.91 (7) ±0.1667	96.89 (8) ±0.09624	97 (5.5) ±0.1667
5	90.43 (7) ±1.409	91.7 (4) ±1.479	88.2 (8) ±1.194	91.39 (6) ±0.6767	91.55 (5) ±0.5968	**92.34** (1) ±0.3907	**92.03** (2.5) ±0.2256	**92.03** (2.5) ±0.8132
6	94.52 (8) ±0.1456	95.46 (5) ±0.1531	94.73 (7) ±0.2191	95.4 (6) ±0.1531	95.51 (4) ±0.1205	**95.84** (2) ±0.1205	95.82 (3) ±0.1531	**95.91** (1) ±0.0884
7	65.61 (7) ±0.2328	67.34 (6) ±0.2159	65.56 (8) ±0.4	68 (5) ±0.5929	69.28 (4) ±0.4434	**69.7** (3) ±0.6772	**69.78** (2) ±0.128	**69.9** (1) ±0.284
8	87.4 (7) ±0.6861	87.68 (5.5) ±1.048	87.68 (5.5) ±1.048	88.24 (3) ±0.6861	87.11 (8) ±1.048	**89.08** (2) ±0.6861	87.96 (4) ±1.585	**89.36** (1) ±1.048
9	21.23 (7) ±0.1700	22.93 (5) ±0.411	23.47 (4) ±0.3091	**26.23** (1) ±0.1247	24.33 (3) ±0.411	24.67 (2) ±0.776	19.9 (8) ±1.283	21.43 (6) ±1.195
10	63.11 (5) ±0.6033	66.73 (3) ±1.256	52.39 (8) ±0.3864	64.21 (4) ±0.9514	68.1 (2) ±0.8253	**68.99** (1) ±0.5378	61 (6) ±0.9514	59.08 (7) ±1.022
11	67.92 (7) ±0.7626	72 (4) ±0.6521	65.55 (8) ±1.499	69.57 (6) ±0.4841	72.66 (2) ±0.1863	**74.5** (1) ±0.8984	71.74 (5) ±0.4269	72.2 (3) ±0.8122
12	55.57 (4) ±0.4028	55.63 (3) ±0.3771	54.53 (6) ±0.5185	54.7 (5) ±0.2944	59.73 (2) ±0.1247	**62.37** (1) ±0.2867	54.03 (7) ±0.04714	54 (8) ±0.2944
13	55.07 (5) ±0.3399	55.37 (3) ±0.5249	52.07 (7) ±0.33	55.2 (4) ±0.4243	56.7 (2) ±0.2944	**58.57** (1) ±0.5793	48.23 (8) ±0.9877	53.67 (6) ±0.834
14	52.69 (7) ±0.3113	**54.58** (1) ±0.7913	53.48 (6) ±0.5392	**54.15** (4) ±0.7674	52.2 (8) ±0.7477	**54.27** (2.5) ±0.8503	53.85 (5) ±1.078	**54.27** (2.5) ±0.1727
	5.64	3.68	6.14	4.11	4.46	2.68	5.46	3.82

spective classifier. The best classifier in this study is VC-bagging with monotonic VC-DomLEM. It obtains the best average rank in experiments with both measures and it has the highest number of best results. However, we failed to prove its superiority over VC-bagging with random monotonic VC-DomLEM in Wilcoxon test (and also with single random monotonic VC-DomLEM in case of results on percentage of correctly classified objects, which is surprising).

The observations made with respect to the inconsistency of the data sets, based on the comparison of values of δ and λ presented in Table 6.1 and the predictive accuracy obtained by single classifiers are also valid for ensembles of classifiers (i.e., the results presented in Tables 6.4 and 6.5).

Finally, we compared the results of VC-bagging with monotonic VC-DomLEM to those obtained by statistical ensembles of classifiers that solve ordinal classification with monotonicity constraints found in the literature (Kotłowski, 2009; Kotłowski and Słow-

iński, 2009). The results of MAE obtained by the classifiers are comparable with little differences indicating almost the same number of results in favour of one or the other method. We should however add that a complete comparison is impossible in this case due to little differences in the setup of experiments (lack of results for percentage of correctly classified objects, different partitioning in folds, and also different treatment of missing values (which are in the minority - see section 6.1.1). Moreover, we used in our experiments two additional data sets.

consistency of bootstrap samples

We continued this part of experiment with the study consistency of objects in bootstrap samples and similarity between samples. The purpose of this study is to show differences between sampling used in standard bagging and in VC-bagging. In this analysis, any object y, for which $\Theta_X^P(y) < 1$ calculated on a random subset of attributes with 50% cardinality is considered as inconsistent (see (4.12)). First, we check the average percentage of inconsistent objects in bootstrap samples, which is presented in Table 6.6. This shows the fraction of inconsistent objects in bootstrap samples created by compared versions of bagging. Then we check the average consistency of an object drawn into sample. This allows us to compare the average probability of object being drawn into the samples.

similarity of bootstrap samples

We also compare in pairs all bootstrap samples created by each of versions of bagging. We check the similarity of all objects in samples and similarity of inconsistent objects in samples, which is presented in Table 6.6. We define similarity for a pair of bootstrap samples as the value of the ratio of the sum of the same objects drawn into the samples to the number of objects in the samples. Then we calculate similarity for a given version of bagging as the average value of similarity of all pairs of samples created by the version of bagging. This allows to compare how diversified are bootstrap samples created by each version of bagging. Moreover, we can check the diversity of the samples with respect to all objects and only with respect to the inconsistent objects.

The average percentage of inconsistent objects in Table 6.6, indicate that samples used by VC-bagging are more consistent than those drawn in standard bagging. This is not the case only for three data sets: `balance`, `ERA`, and `LEV`, for which bootstrap samples are composed of 100% of inconsistent objects in both cases.

Similarity of bootstrap samples created by standard bagging is always close to 0.75 regardless of whether it is calculated for all objects or for inconsistent ones. We treat this result as a base line for our comparison. We can see that similarity measured for all objects drawn in bootstrap samples created by VC-bagging is usually lower than in case of standard bagging. The exceptions to this rule are consistent data sets: `breast-w`,

Table 6.6: Consistency and similarity of bootstrap samples created by standard bagging and by VC-bagging with measure ϵ

		bagging			VC-bagging		
Id	Data set	% inconsistent	sim. all	sim. inconsistent	% inconsistent	sim. all	sim. inconsistent
1	balance	100	0.7507	0.7507	100	0.4428	0.4426
2	breast-c	93.05	0.7564	0.7561	91.76	0.7531	0.7506
3	breast-w	15.93	0.7519	0.7492	9.77	0.7527	0.5808
4	car	67.01	0.7512	0.7508	56.29	0.7231	0.6489
5	cpu	55.17	0.7534	0.7541	53.49	0.7554	0.7429
6	bank-g	6.38	0.7521	0.7492	3.34	0.7512	0.5472
7	fame	59.47	0.7499	0.7502	57.99	0.7499	0.7422
8	denbosch	39.6	0.7525	0.7573	3.76	0.7314	0.1431
9	ERA	100	0.7514	0.7514	100	0.7385	0.7387
10	ESL	99.25	0.7526	0.7526	99.05	0.7462	0.7463
11	housing	33.23	0.7542	0.7554	14.76	0.7207	0.4745
12	LEV	100	0.7514	0.7514	100	0.6530	0.6532
13	SWD	99.89	0.7514	0.7514	99.87	0.7407	0.7409
14	windsor	92.53	0.7508	0.7507	89.58	0.7358	0.7295

cpu, bank-g, and fame. Moreover, for most of the data sets similarity of inconsistent objects in the samples is even lower. When we count out these data sets, for which bootstrap samples are composed only of inconsistent objects, this tendency is not taking place only for three data sets: cpu, ESL, and SWD. These results are concordant with our analysis of consistency and similarity of bootstrap samples created by bagging and VC-bagging on non-ordinal data sets (Błaszczyński et al., 2009b).

diversity vs. error diagrams

One further way to get insight into the behaviour of the ensemble methods is to construct diversity vs. error diagrams (Margineantu and Dietterich, 1997; Dietterich, 1998). These diagrams help to visualize the predictive accuracy and the diversity of the component classifiers. For each pair of the component classifiers, we measure their predictive accuracy as the average of MAE on the test data (in our case it was measured in 10-fold cross validation). More precisely, we calculate MAE twice for each classifier. First, we calculate a *univocal MAE*, i.e., MAE of these classifiers that suggest assignment to exactly one class. Then, we calculate a *non-univocal MAE*, i.e., MAE for classifiers that suggest assignment to one class or to multiple classes. Suggestion of assignment to multiple classes is in this case treated as assignments to each of the classes separately. Non-univocal MAE allows us to get more insight into behaviour of a single component classifier applying the standard classification method since in most cases it suggests assignment to a set of contiguous classes.

univocal and non-univocal MAE

We measure the diversity of component classifiers by computing a degree of agreement statistics κ, which is defined, for a pair of classifiers, as follows. For n classes, let C be an $n \times n$ square array such that C_{ij} contains the number of test objects assigned

to class i by the first classifier and into j by the second classifier. We define:

$$\Theta_1 = \frac{\sum_{i=1}^{n} C_{ii}}{|U|}, \tag{6.4}$$

which is an estimate of the probability that the two classifiers in pair agree.

The problem with Θ_1 is that in case of imbalanced data sets, where one class is much more common than the others, all classifiers may agree by choosing majority class and in consequence obtain high values of Θ_1. In case of κ, such situation is corrected by computing

$$\Theta_2 = \sum_{i=1}^{n} \Big(\sum_{i=1}^{n} \frac{C_{ij}}{|U|} \times \sum_{i=1}^{n} \frac{C_{ji}}{|U|} \Big), \tag{6.5}$$

which estimate the probability that the classifiers in pair agree by chance. These two statistics, are used to define diversification statistics κ as:

$$\kappa = \frac{\Theta_1 - \Theta_2}{1 - \Theta_2}. \tag{6.6}$$

When the two classifiers in pair agree only by chance $\kappa = 0$. On the other hand, when the two classifiers agree on every example $\kappa = 1$. Thus, low values of κ indicate that responses of component classifiers in an ensemble are highly diversified.

The diversity vs. error diagrams for all data sets are presented in the following Figures 6.1,8.1-8.13. We present here only the diagram for `balance` data set in Figure 6.1. The rest of diagrams were moved to the appendix 8. All these diagrams result from the repeated 10-fold cross validation experiments described so far. Each of the figures includes two diagrams for standard bagging with monotonic VC-DomLEM and random monotonic VC-DomLEM. Each of the figures also includes two diagrams for VC-bagging with monotonic VC-DomLEM and random monotonic VC-DomLEM.

We continued the analysis of experimental results with comparison of diagrams for standard bagging to these for VC-bagging. This allows us to see how far the differences in similarity and consistency of bootstrap samples created by each version of bagging propagates to diversity and predictive accuracy of component classifiers constructed on these samples.

As a general tendency, we can observe that for consistent data sets, the classifiers give very compact clouds of points. Each point in such clouds has low error rate and high value of κ, which indicates that the component classifiers are accurate but not very diverse. This is the case for data sets `breast-w`, `car`, and also but to lower extent for `bank-g` (Figures 8.2, 8.3, and 8.5). On the other hand, for inconsistent data sets such as `breast-c` and `ERA` (Figures 8.1, 8.8), the classifiers give diagrams with wide

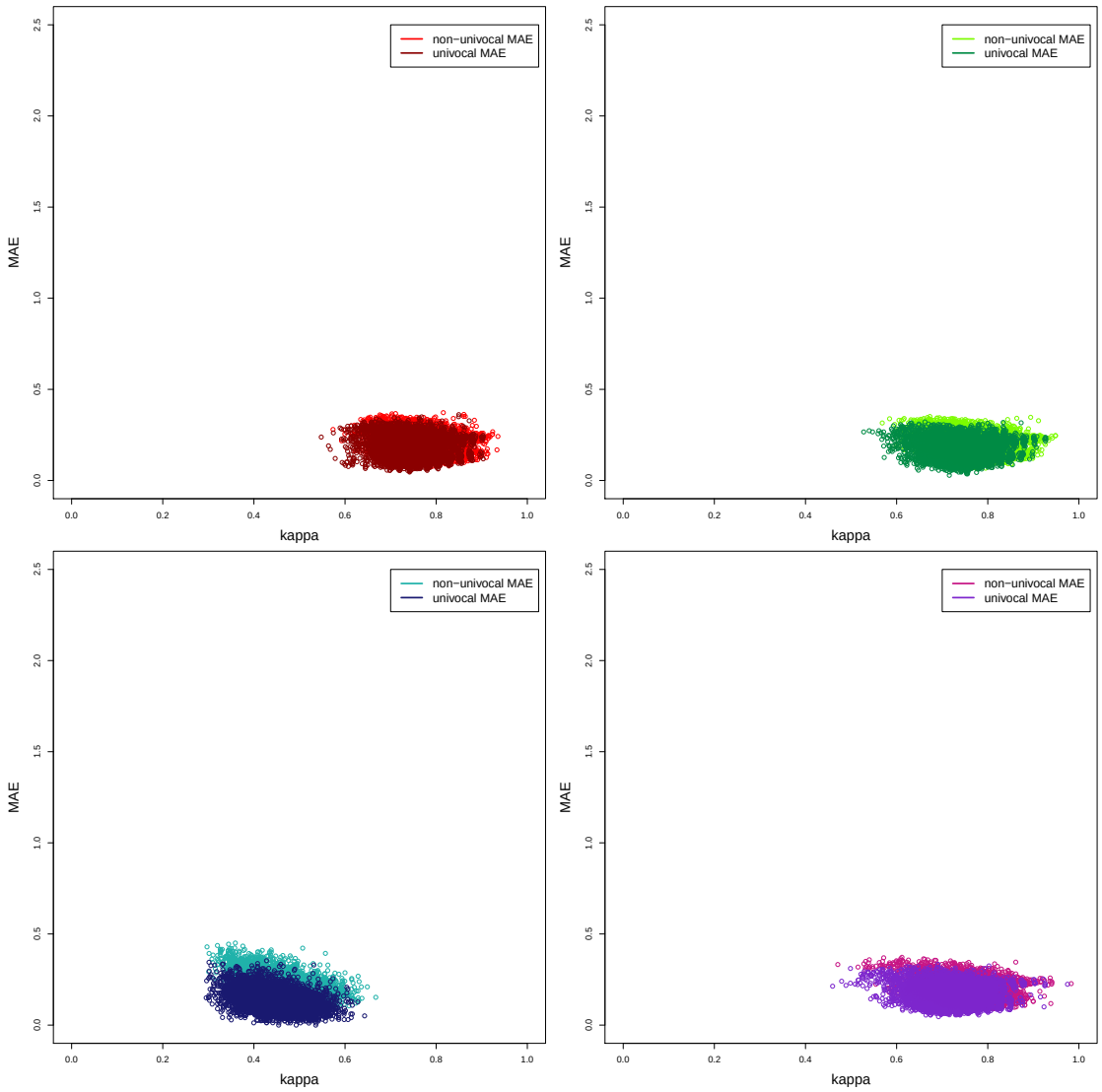


Figure 6.1: Diversity vs. error diagrams for `balance` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

ranges of accuracy and diversity (in case of `ERA` this effect is increased by the number of classes). This clearly show the trade off between accuracy and diversity. As the component classifiers become more accurate, the must also become less diverse.

The shape of the diagrams usually does not change for the same data set. This can be explained by the same type of component classifier, i.e., VC-DomLEM and bagging used in all cases. Different shapes of the diversity vs. error diagrams are produced by completely different classifiers as for example bagging and boosting (Dietterich, 1998).

There are, however, two data set, for which our classifiers produced different shapes of diagrams. In case of small and inconsistent data set `breast-c` (Figure 8.1), the shape of diagrams for monotonic VC-DomLEM are different from those of diagrams for random montonic VC-DomLEM. Moreover, in case of inconsistent data set `SWD`, the shape of diagram for bagging with random monotonic VC-DomLEM is dramatically different from the shape of the rest of diagrams for this data set. It seems that VC-bagging is stabilizing the behaviour of random monotonic VC-DomLEM in this case (which is also confirmed by better overall result of the VC-bagging ensemble).

Increased diversity of bootstrap samples created by VC-bagging results in better diversity of component classifiers. This effect is visible for most of the data sets. More precisely, in case of VC-bagging with monotonic VC-DomLEM, we observed this effect for the following data sets: `balance`, `breast-w`, `car`, `bank-g`, `fame`, `denbosch`, `ERA`, `housing`, `LEV`, and `windsor`. In case of VC-bagging with random monotonic VC-DomLEM, we observed this effect for the following data sets: `breast-w`, `car`, `cpu`, `bank-g`, slightly for `fame`, `denbosch`, `ERA`, `LEV`, and `windsor`. A visible decrease of diversity is visible only for VC-bagging with random monotonic VC-DomLEM for `SWD`. It is however, combined with a visible increase of accuracy of component classifiers in the ensemble.

Changes in the accuracy of component classifiers are less apparent even though we know from previous results that these classifiers were learned on more consistent bootstrap samples. Increased accuracy can be, however, observed in case of VC-bagging with monotonic VC-DomLEM for the following data sets: `balance`, slightly for `cpu`, `ERA`, `ESL`, `housing`, `LEV`, slightly for `SWD`, and `windsor`. In case of VC-bagging with random monotonic VC-DomLEM, we can observe increase of accuracy for data sets `breast-c`, `ERA`, and slightly for `housing`. We can, however, also observe decrease of accuracy of random montonic VC-DomLEM component classifiers in case of data set `LEV`. Since this decrease is not visible in case of monotonic VC-DomLEM component classifiers, we can attribute this change to additional randomization introduced to the ensemble by random monotonic VC-DomLEM. This decrease resulted in decrease of the overall predictive accuracy of the ensemble.

6.3 Interpretability

Interpretability of results of classification is an important issue from the view point of decision aiding (see section 1.2.1). In decision aiding, a recommendation suggested by a classifier needs to be interpretable for a human decision maker. This is why the recom-

mendation needs to be consistent and traceable. It is known, that since interpretability is not measurable, it is only possible to assess subjectively this aspect for various classifiers. Taking this all into account, we comment on each of the classifiers compared in the experiments showing its strong and weak points.

First, we can distinguish "black-box" classifiers, like Naïve Bayes and SVM. Estimates of distributions and support vectors in space that is higher dimensional than the original problem are hardly interpretable. Without additional processing, results of these methods are oracle suggestions. Moreover, these methods do not take into account domain knowledge about the orders and monotonicity constraints, which can make their recommendations inconsistent.

The well interpretable decision tree models constructed by C4.5 and discriminative rule models induced by RIPPER, also do not take into account the domain knowledge. In Figure 6.2, we show the inconsistency that may occur in interpretation of such models on example. The decision tree model has been constructed for data set `bank-g`. The leaf marked in red shows that this model is inconsistent. It means that when a firm obtains a better evaluation on gain ordered criterion "Net worth / Total liabilities" it makes this firm less attractive for investments (i.e., firm is classified as distressed). Such suggestion may be confusing because it is inconsistent with the domain knowledge.

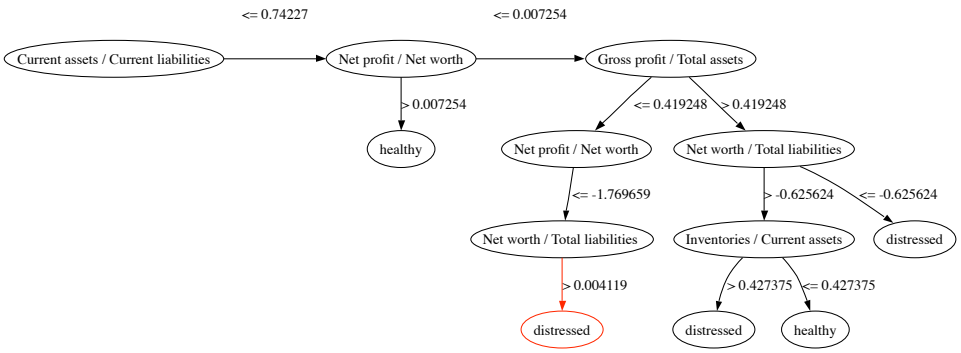


Figure 6.2: Inconsistency of decision tree model for `bank-g` data set.

This kind of inconsistent suggestions does not occur, of course, in results of ordinal classifiers that preserve monotonicity constraints. Among such classifiers, we compare two instance-based methods: OLM and OSDL to decision rule classifier VC-DomLEM and ensemble of VC-DomLEM classifiers. OLM stores its model as a subset of the training objects, which is called a rule base. OSDL, on the other hand stores all objects. These models are using the stored objects during classification to check dominance relation between them and the classified object. The suggestions given by these models and

the process in which they are constructed is consistent and rather comprehensible. On the other hand, it may be hardly traceable.

Finally, VC-DomLEM (in all presented variants, including bagged and VC-bagged version) builds a model which consists of a set of rules that are covering all sufficiently consistent objects in the learning set. This feature is important from the model consistency point of view, as it ensures that information required to classify any object (from the learning set) is included in the model. Decision rule model is well known to be one of the most interpretable forms of knowledge representation. It is also traceable, which means that when an object is assigned a suggestion, one can see all rules supporting this suggestion. Moreover, when analysis of rules supporting the suggestion is not enough, one can also see the objects that are supporting the rules. The rules are characterized by the consistency measures that are part of the domain knowledge. The same consistency measures characterize objects. The set of rules induced by VC-DomLEM is also minimal, meaning that it does not contain any unnecessary rules and that rules does not contain unnecessary elementary conditions. Stronger and shorter rules are particularly relevant since they represent strongly established relationships between causes and effects. Monotonic VC-DomLEM is known to induce shorter rules than non-monotonic VC-DomLEM (Błaszczyński et al., 2009d). These rules have comparable strength.

Ensembles of rule sets allow to obtain better predictive accuracy, but this result comes with reduced interpretability of the induced model. One can expect that analyzing multiple sets of decision rules suggesting a particular assignment may be more difficult than analysis of suggestions given by one set of decision rules. On the other hand, the learning of the ensemble in VC-bagging has a clear interpretation since it promotes objects that are good candidates for consistent rules in samples on which the component classifiers are learned. The same consistency measures being part of domain knowledge are used to this end.

6.4 Summary

We have shown that the algorithms presented in the thesis are competitive with the other existing approaches to both ordinal classification with monotonicity constraints and non-ordinal classification. In case of single classifiers, our classifiers obtained the best results of all compared classifiers. Non-ordinal classifiers, which do not take monotonicity constraints into account performed not worse than some ordinal methods, i.e., OLM, in this comparison. However, non-ordinal classifiers can give models inconsistent with domain knowledge. This can further lead to problems with interpretation of suggestions

given by these models, which is unacceptable when a method is applied in decision aiding. Better interpretability, is one more aspect that distinguishes our models form the others. We propose a holistic approach, in which the information how to treat the inconsistencies in data, expressed by measure of consistency, is a part of domain knowledge. This information allows to say which objects are sufficiently consistent. The same information allows then to induce sufficiently consistent decision rules.

Moreover, we have shown that, the measures λ and δ introduced in this thesis allow to identify data sets with a significant number of inconsistencies which make these data sets hard to learn. To increase predictive accuracy in such cases, we proposed variable consistency bagging (VC-bagging) ensembles of classifiers. These ensembles proved to improve the predictive accuracy with respect to single classifiers and standard bagging ensembles. Also in this case, our ensemble algorithms are at least comparable to the best existing ensemble approaches to ordinal classification with monotonicity constraints. We have shown the source of improved performance of variable consistency bagging, which lies in higher diversity and consistency of samples on which classifiers are learned. Improved predictive performance of ensemble classifiers comes, however, at the expense of decreased interpretability of the model.

CHAPTER 7

Summary and Conclusions

In this thesis, we considered the problem of ordinal classification with monotonicity constraints. According to the list of goals that we presented in section 1.3, we defined and characterized the variable consistency dominance-based rough set approach (VC-DRSA) to this problem. First, we introduced monotonicity properties for consistency measures. Monotonicity properties were necessary to define monotonic probabilistic rough set approaches. These subjects are covered in chapters 2, and 3. Then, we proposed a new rule induction algorithm from probabilistic lower approximations, called VC-DomLEM. VC-DomLEM is a sequential covering algorithm that induces sets of decision rules satisfying constraints on monotonic or non-monotonic consistency measures. We also proposed a new method of constructing ensembles of classifiers that uses consistency measures, which is called variable consistency bagging (VC-bagging). Finally, we introduced a new classification method for dominance-based rough set approaches, which solves conflicts between rules assigning an object to multiple classes. These subjects are covered in chapters 4, and 5.

Our approach allows to construct classifiers that are competitive in terms of the predictive accuracy and that are favorable in terms of interpretability. These classifiers are consistent with the domain knowledge about the order and monotonicity. We verified these claims in the computational experiment presented in chapter 6. Thus, in our opinion, the goal of the thesis has been achieved. Below, we provide a more detailed summary of our results together with some plans for future research.

- **Monotonic consistency measures**. We proposed different measures of the overlap between a granule of knowledge based on a considered object and the approxi-

mated set or its complement. We called such measures *consistency measures*. The consistency measures are meant to be easy in interpretation so that one can directly specify properties of objects included in the probabilistic lower approximation. Different consistency measures were used to express different view on the consistency. Furthermore, we proposed four monotonicity properties for consistency measures: (m1), (m2), (m3), and (m4). Two of these properties concern measures defined for indiscernibility-based granules and dominance-based granules, namely: monotonicity with respect to the set of attributes (m1), and monotonicity with respect to the set of objects (m2). Two additional properties concern only dominance-based granules, these are: monotonicity with respect to unions of classes (m3), and monotonicity with respect to the dominance relation (m4). Monotonicity properties guarantee that any object from a monotonic lower approximation will belong to this lower approximation after the data set is extended with respect to the set of attributes, set of objects or union of ordered classes. Monotonicity properties guarantee also the same behavior of objects from lower approximation when improvement of evaluation of any object in the data set takes place. We have shown that consistency measures used so far in the definition of probabilistic rough approximations lack some of these monotonicity properties. This observation led us to propose new measures enjoying desirable properties. Moreover, monotonicity properties of consistency measures proved also to be important in further stages of construction of the decision rules classifiers.

- **Monotonic probabilistic rough set approaches**. We used consistency measures having desirable monotonic properties to define monotonic probabilistic rough set approaches. Our proposal is a general probabilistic extension of the rough set approach. We defined two monotonic probabilistic rough set approaches:

  - monotonic Variable Consistency Indescernibility-based Rough Set Approach (VC-IRSA), that involves granules of knowledge defined by the indiscernibility relation,
  - monotonic Variable Consistency Dominance-based Rough Set Approach (VC-DRSA), that involves granules of knowledge defined by the dominance relation.

  According to our best knowledge, no such general extension was proposed so far. Moreover, we used monotonic probabilistic lower approximations to define positive, negative and boundary regions, which are more desirable as a basis for induction

of decision rules. Basing on the positive regions, we defined measures that allow to estimate attainable predictive accuracy of rough-set-based classifiers.

- **Decision rules induction algorithm VC-DomLEM**. We developed a sequential covering algorithm that induces sets of probabilistic decision rules covering the positive regions. This algorithm produces rules for ordinal classification with monotonicity constraints. The resulting rules satisfy constraints specified on the consistency measures. This property makes the set of rules to be traceable. Each of the rules is characterized by consistency measure that corresponds to consistency measure used to define the probabilistic lower approximation. The VC-DomLEM algorithm is designed in VC-DRSA and it involves two steps. In the first step, probabilistic lower approximations are constructed. These approximations consist of objects that are sufficiently consistent according to the consistency measure. Then, on the basis of the lower approximations, positive regions are determined. In the second step, a set of probabilistic decision rules is induced to cover the positive regions. The type of rules depends on the consistency measure that is used in the first step. We proved that it is possible to induce rules that cover monotonic and non-monotonic probabilistic lower approximations. We have shown that induction of rules that satisfy monotonic consistency measures is more effective than induction of those that satisfy non-monotonic consistency measures. Moreover, VC-DomLEM with monotonic consistency measures induces rule sets that serve as more accurate classifiers. These rule sets are composed of shorter decision rules that are easier to interpret. Monotonic VC-DomLEM achieved the best predictive accuracy results among all the single classifiers, which was shown experimentally on real-world data sets.

- **Ensembles of classifiers in VC-bagging**. We developed a bagging scheme, in which the probability of selecting an object in the bootstrap sampling depends on the consistency of the object. The same consistency measures that are used to define probabilistic lower approximations are used to measure the consistency of objects. In the developed variable consistency bagging (VC-bagging) consistent objects are more likely to be selected to bootstrap samples than inconsistent ones. The bootstrap samples, that are shifted towards consistent object are then used to construct component classifiers of the ensemble. We considered consistency of objects with respect to description by the whole set of attributes (criteria) and by random subsets of attributes. The resulting ensembles of classifiers are learned on samples that are more diversified and more consistent than in standard bag-

ging ensembles. The VC-bagging scheme is general enough to be used with any component classifier. Moreover, it can be used to solve different classification problems depending on the consistency measure that is used to evaluate objects and aggregation rules applied to suggestions of assignment of the component classifiers. We applied VC-bagging to ordinal classification problem with monotonicity constraints. The component classifiers in such bagging ensembles were composed of decision rules induced by monotonic VC-DomLEM from bootstrap samples of objects structured using VC-DRSA. This application of VC-bagging achieved high predictive accuracy on the real data sets.

- **New classification method for dominance-based rough set approaches**. We proposed a new classification scheme for DRSA and VC-DRSA that is able to deal with imprecise and contradictory suggestions given by the matching rules. This classification scheme is based on a notion of score coefficient associated with a set of rules covering object and classes to which these rules may assign the object. The score coefficient reflects relevance between rules and class to which they assign objects. A vector of values of score coefficients calculated for an object with respect to each class can be interpreted as a distribution of relevance between rules that cover classified object and classes. This classification method produced the most accurate suggestions of assignment among the single classifiers when it was used with rules induced by VC-DomLEM algorithm.

We present the following list of subjects as a plan for future research.

- **Adaptive variable consistency ensembles of classifiers**. An interesting extension of the idea of variable consistency ensembles are adaptive variable consistency ensembles. Such ensembles are related to the methods that can improve accuracy of unstable classifiers by perturbing and combining (P&C) (Breiman, 1998, 1999, 2001). The adaptive variable consistency ensembles iteratively perturb the training data by sampling objects with probability of selecting a given object being modified by its importance and consistency. The importance is estimated at each step by the accuracy of component classifiers. The consistency is estimated at each step by consistency measures. Multiple component classifiers are constructed iteratively, with each component classifier being learned on samples composed of objects that are important according to the predictive accuracy of the component classifiers from previous iterations and that are consistent. In

this way, the ensemble is adaptively focusing on objects that are hard to learn but consistent.

- **Methods that improve interpretability of an ensemble rule classifiers**. Ensemble methods are known to increase the predictive accuracy of rule-based classifiers. However, due to their increased complexity, they are less interpretable then their components. We are working on presentation methods that should give more intuitive insight into classification results provided by multiple sets of decision rules.

- **Extension of experimental comparison**. We plan to extend experimental comparison of our methods to more real-world data sets. We are gathering data sets for such a comparison.

CHAPTER 8

Appendix

8.1 Notation

Table 8.1: Basic notation used thorough the thesis.

Symbol	Meaning
U	the universe of discourse, i.e., a set of all objects in the data set
A	a set of criteria and regular attributes, it is composed of two disjoint sets of condition attributes C and decision attributes D; further a distinction between regular attributes G and criteria Q is made
a	an attribute $a \in A$
G	a set of regular attributes $G \subseteq A$
g	a regular attribute $g \in G$
Q	a set of criteria $Q \subseteq A$
q	a criterion $q \in Q$
C	a set of condition attributes $C \subset A$
c	a condition attribute $c \in C$
D	a set of decision attributes $D \subset A$, $C \cap D = \emptyset$
d	a decision attribute $d \in D$
V_{a_i}	a value set of attribute $a_i \in A$
X_i	a decision class i, $X_i \subset U$
$X_i^{\geq}$	an upward union of decision classes $j > i$, $X_j \subset U$
$X_i^{\leq}$	a downward union of decision classes $j < i$, $X_j \subset U$

Continued on Next Page...

Table 8.1 Notation – continued

Symbol	Meaning
X	a set $X \subseteq U$ of objects that belong to one class or union of decision classes; it can be further specified to $X^{\geq}$ or $X^{\leq}$ if there is a requirement for distinction between a set of objects belonging to an upward union of decision classes from a set of objects that belong to a downward union of decision classes
Θ	an (object) consistency measure
$f_X^P(y)$	an (object) gain-type consistency measure
$g_X^P(y)$	an (object) cost-type consistency measure
A_X	the upper limit value of gain-threshold α_X
B_X	the upper limit value of cost-threshold β_X
α_X	the gain-threshold taking values from $[0, A_X]$
β_X	the cost-threshold taking values from $[0, B_X]$
(m1)	the property of monotonicity with respect to the set of attributes, defined for consistency measures in IRSA as ((2.14), (2.15)) and for consistency measures in DRSA as ((3.21), (3.22))
(m2)	the property of monotonicity with respect to the set of objects, defined for consistency measures in IRSA as ((2.16), (2.17)) and for consistency measures in DRSA as ((3.23), (3.24))
(m3)	the property of monotonicity with respect to the unions of classes defined for consistency measures in DRSA as ((3.25), (3.26))
(m4)	the property of monotonicity with respect to the dominance relation defined for consistency measures in DRSA as ((3.27), (3.28))
$\underline{P}^{\alpha_X}(X)$	a P-lower approximation of set X defined for a gain-type consistency measure
$\underline{P}^{\beta X}(X)$	a P-lower approximation of set X defined for a cost-type consistency measure
$\overline{P}^{\alpha_X}(X)$	a P-upper approximation of set X defined for a gain-type consistency measure
$\overline{P}^{\beta X}(X)$	a P-upper approximation of set X defined for a cost-type consistency measure
$POS_P^{\alpha_X}(X)$	a P-positive region of set X defined for a gain-type consistency measure
$POS_P^{\beta_X}(X)$	a P-positive region of set X defined for a cost-type consistency measure
$NEG_P^{\alpha_X}(X)$	a P-negative region of set X defined for a gain-type consistency measure

Continued on Next Page...

Table 8.1 Notation – continued

Symbol	Meaning
$NEG_P^{\beta_X}(X)$	a P-positive region of set X defined for a cost-type consistency measure
$BND_P^{\alpha_X}(X)$	a P-boundary region of set X defined for a gain-type consistency measure
$BND_P^{\beta_X}(X)$	a P-boundary region of set X defined for a cost-type consistency measure
λ	the ratio of objects in U that may be learned by a rough-set-based classifier; defined in IRSA as (2.33), and in DRSA as (3.47)
δ	the average minimal absolute difference between the index of the class to which an object may be assigned by a rough-set-based classifier and the index of the class to which the object belongs; defined in DRSA as (3.49)
$\hat{\Theta}$	a rule consistency measure
r	a decision rule
$\mathbf{R}$	a set of decision rules
$r_X^{\hat{\theta}_X}$	a decision rule assigning to X, characterized by rule consistency measure $\hat{\theta}_X$
$R_X^{\hat{\theta}_X}$	a set of decision rules assigning to X, and characterized by rule consistency measure $\hat{\theta}_X$
$\|\Phi_{r_X^{\hat{\theta}_X}}\|$	set of objects fulfilling the condition part of rule $r_X^{\hat{\theta}_X}$
$\|\Psi_{r_X^{\hat{\theta}_X}}\|$	set of objects fulfilling the decision part of rule $r_X^{\hat{\theta}_X}$

8.2 Diversity vs. error diagrams

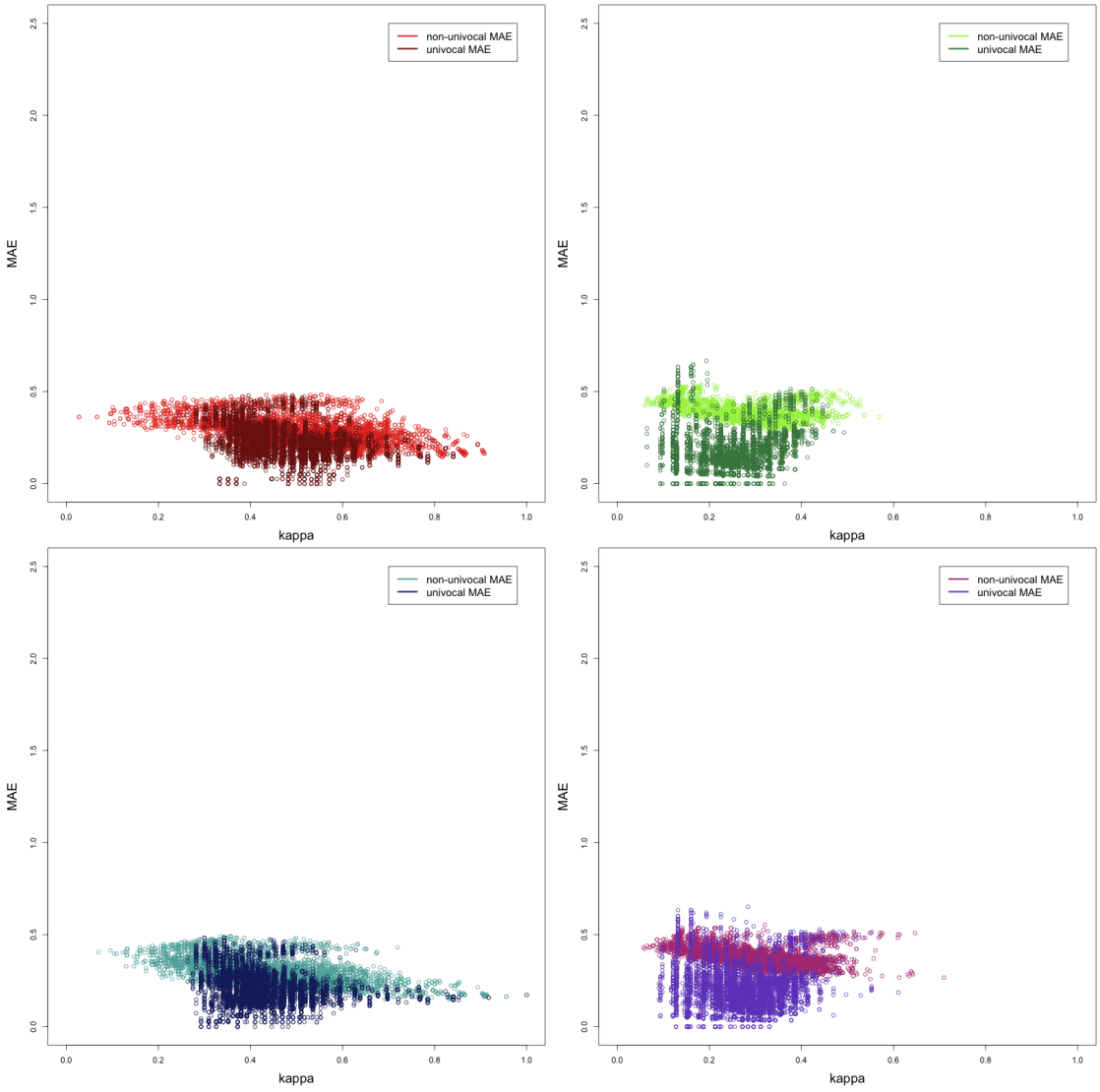


Figure 8.1: Diversity vs. error diagrams for `breast-c` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

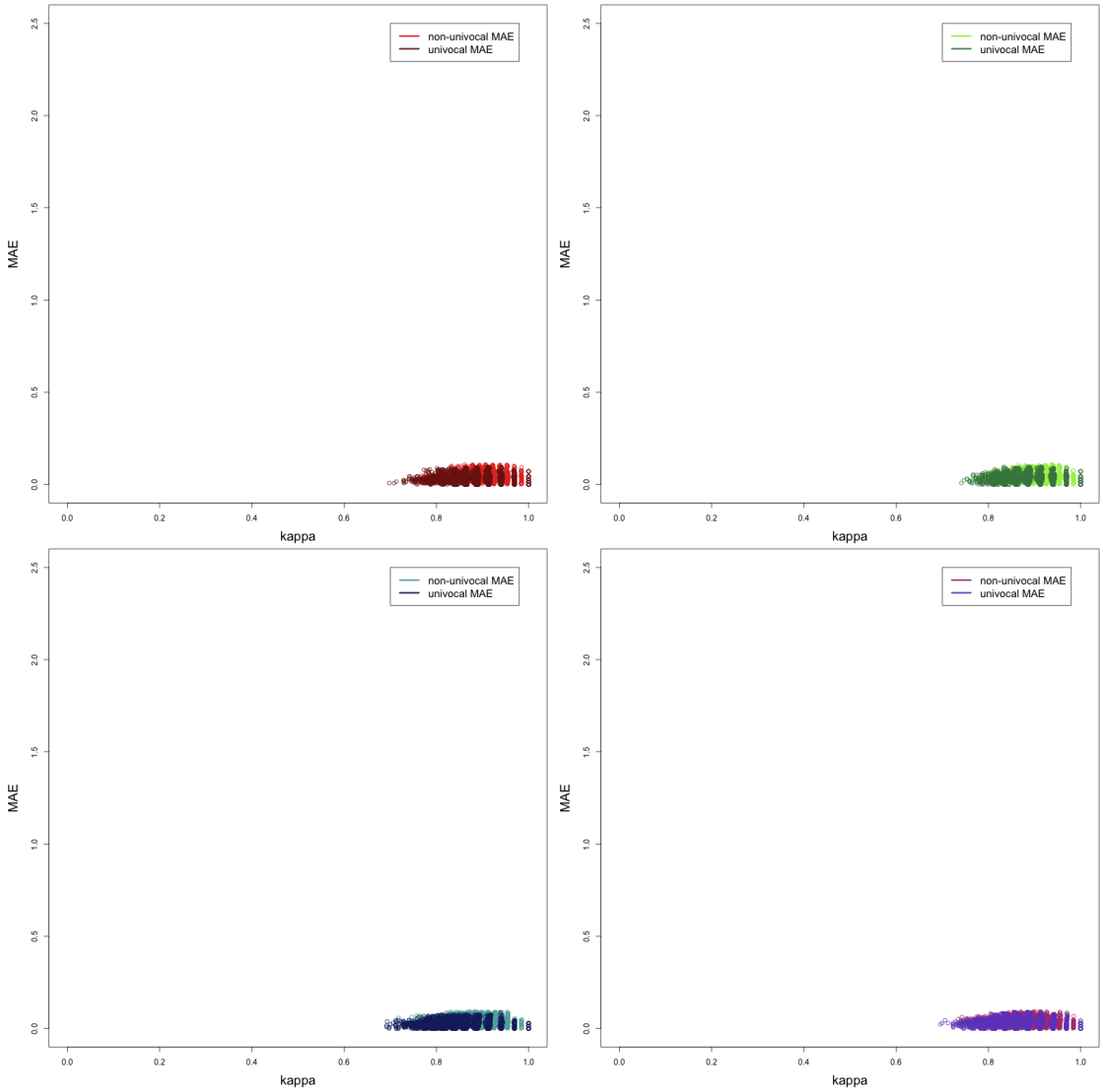


Figure 8.2: Diversity vs. error diagrams for `breast-w` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

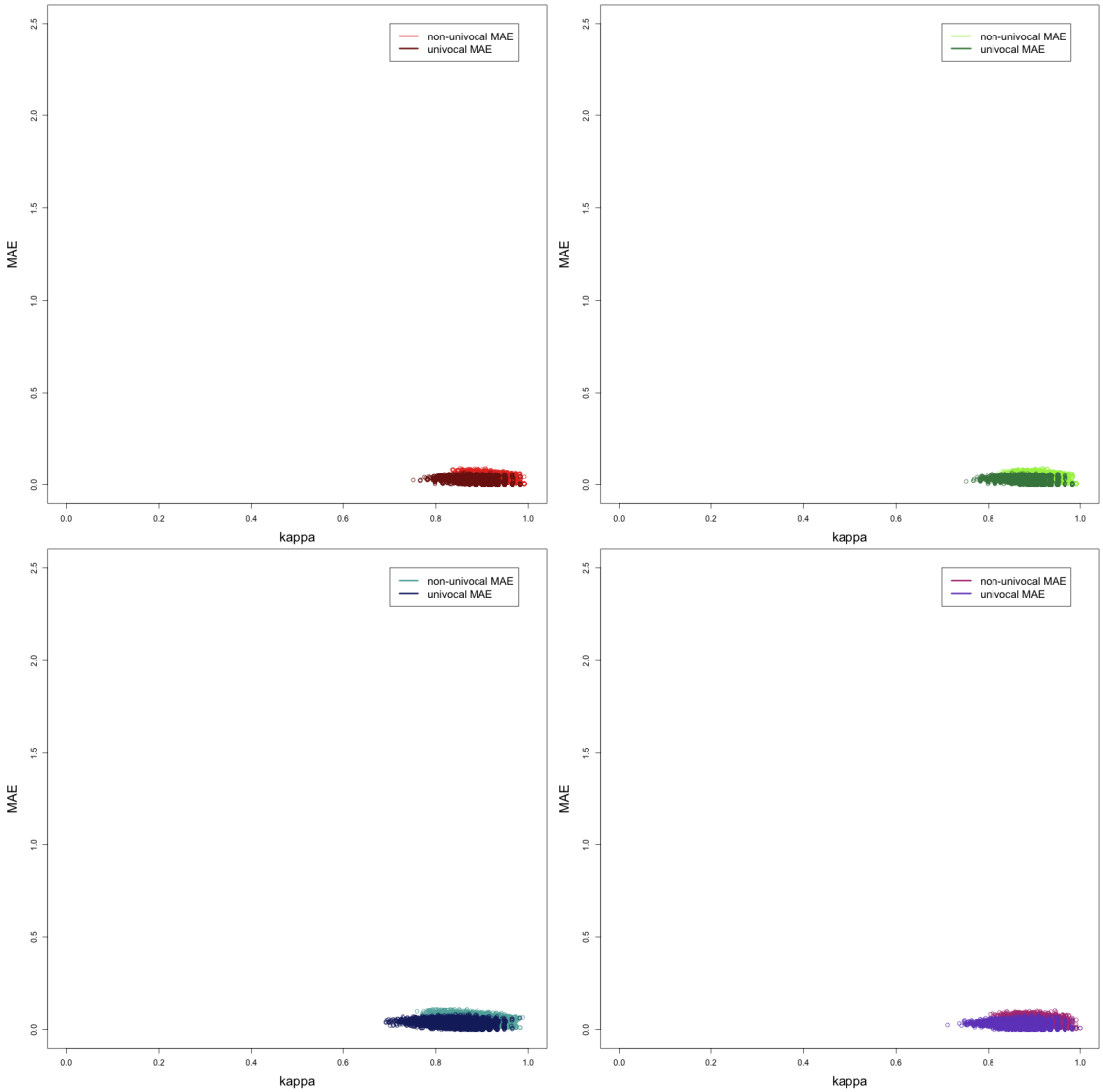


Figure 8.3: Diversity vs. error diagrams for `car` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

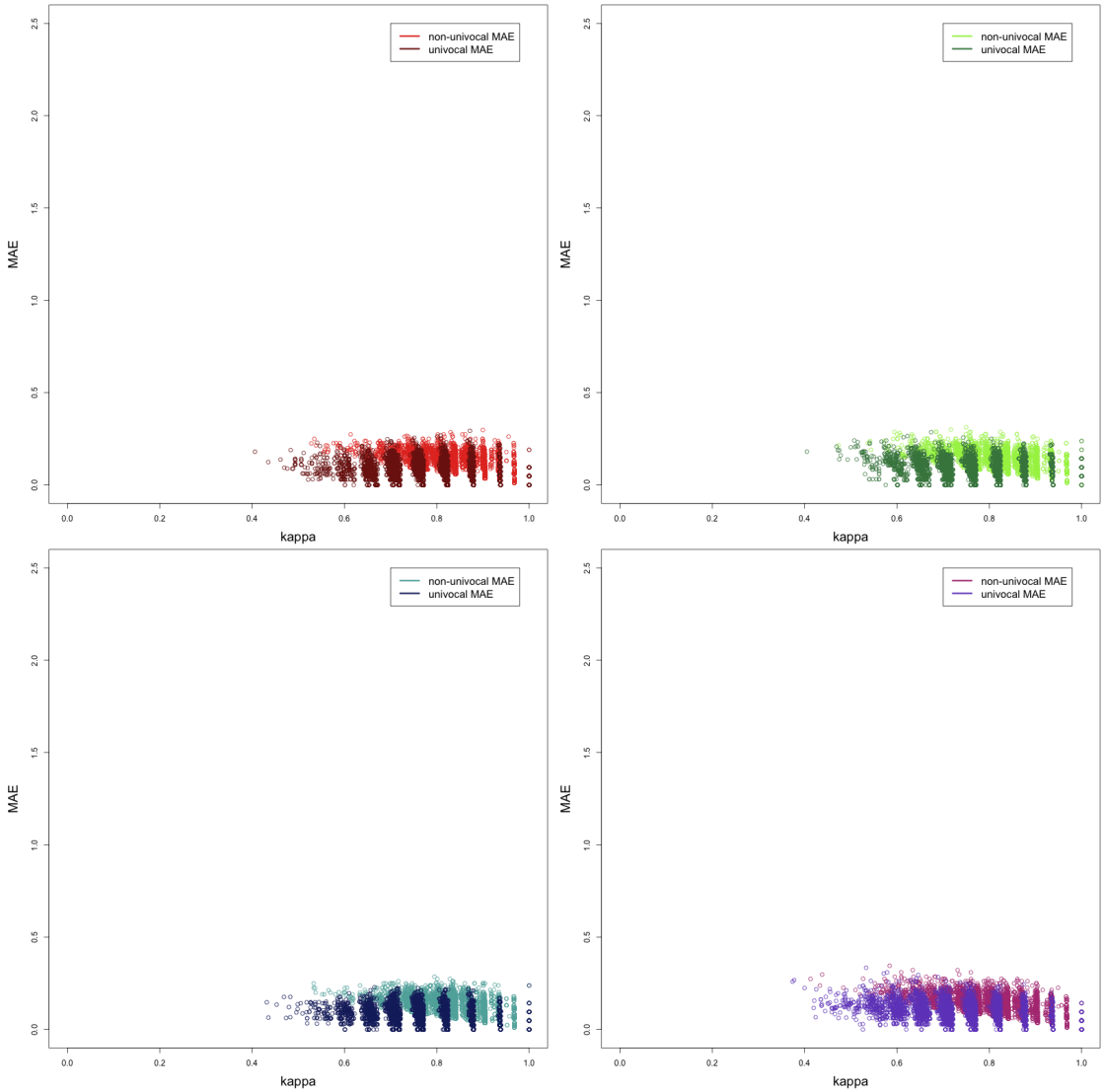


Figure 8.4: Diversity vs. error diagrams for `cpu` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

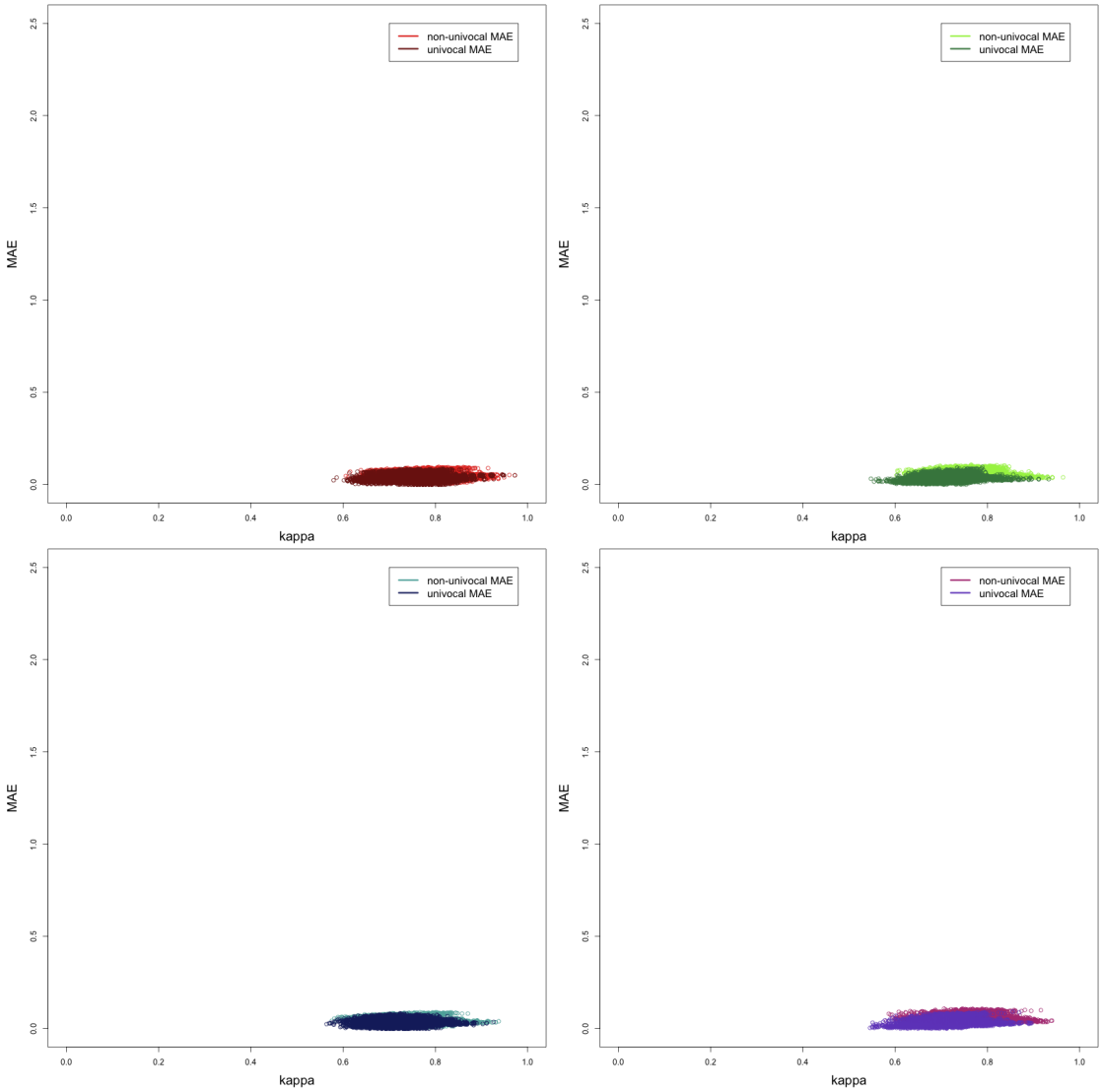


Figure 8.5: Diversity vs. error diagrams for `bank-g` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

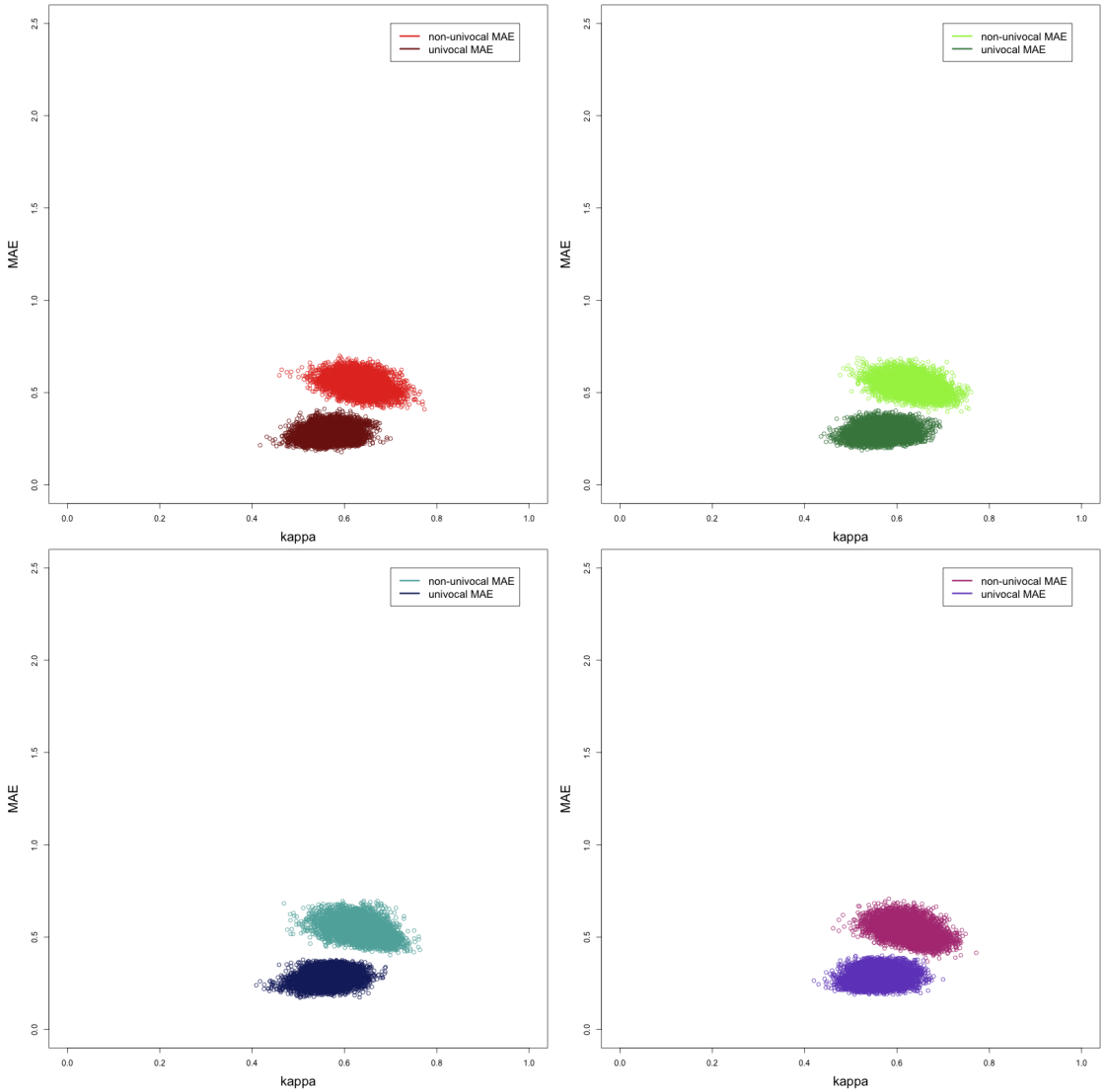


Figure 8.6: Diversity vs. error diagrams for `fame` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

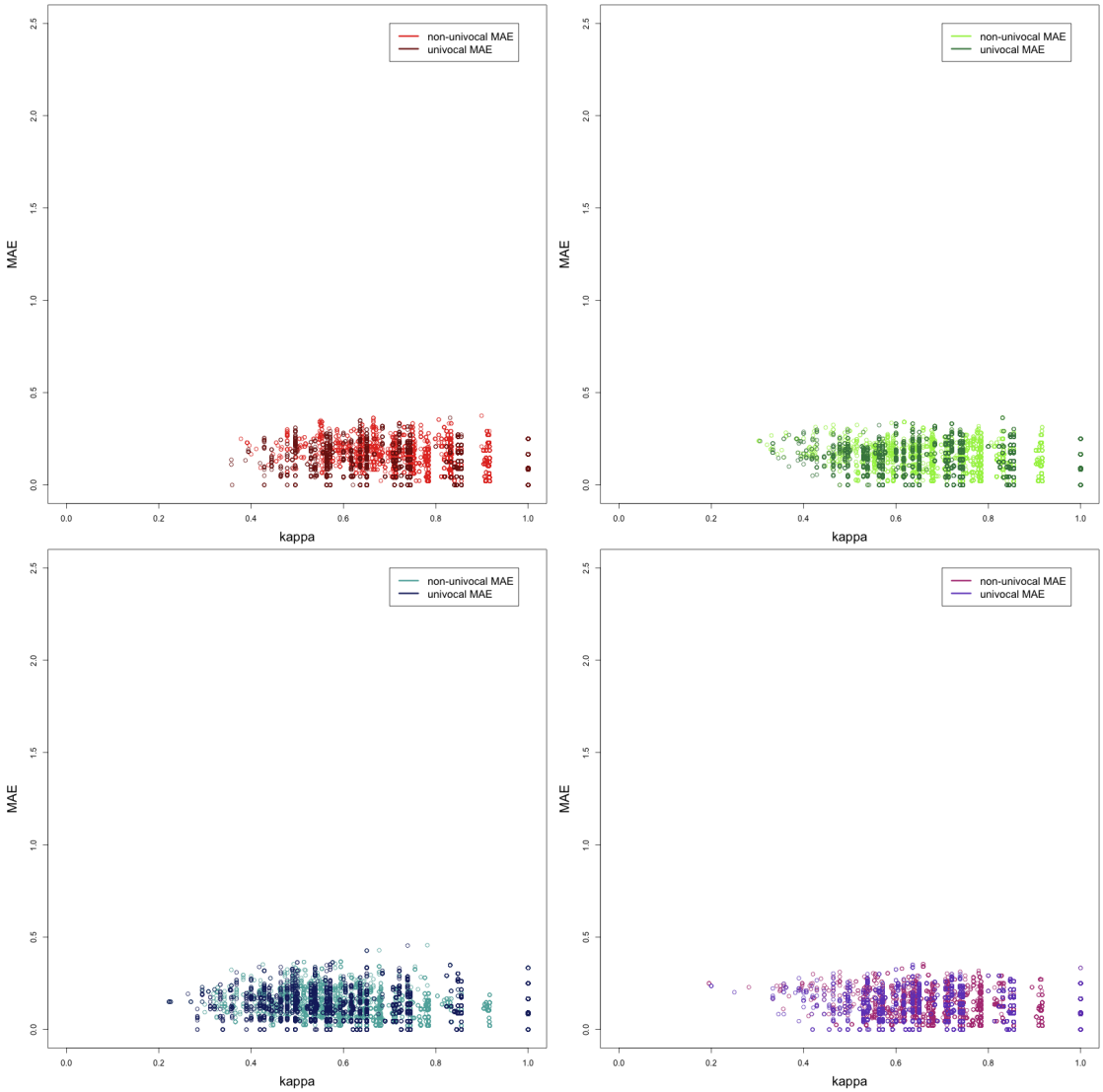


Figure 8.7: Diversity vs. error diagrams for `denbosch` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

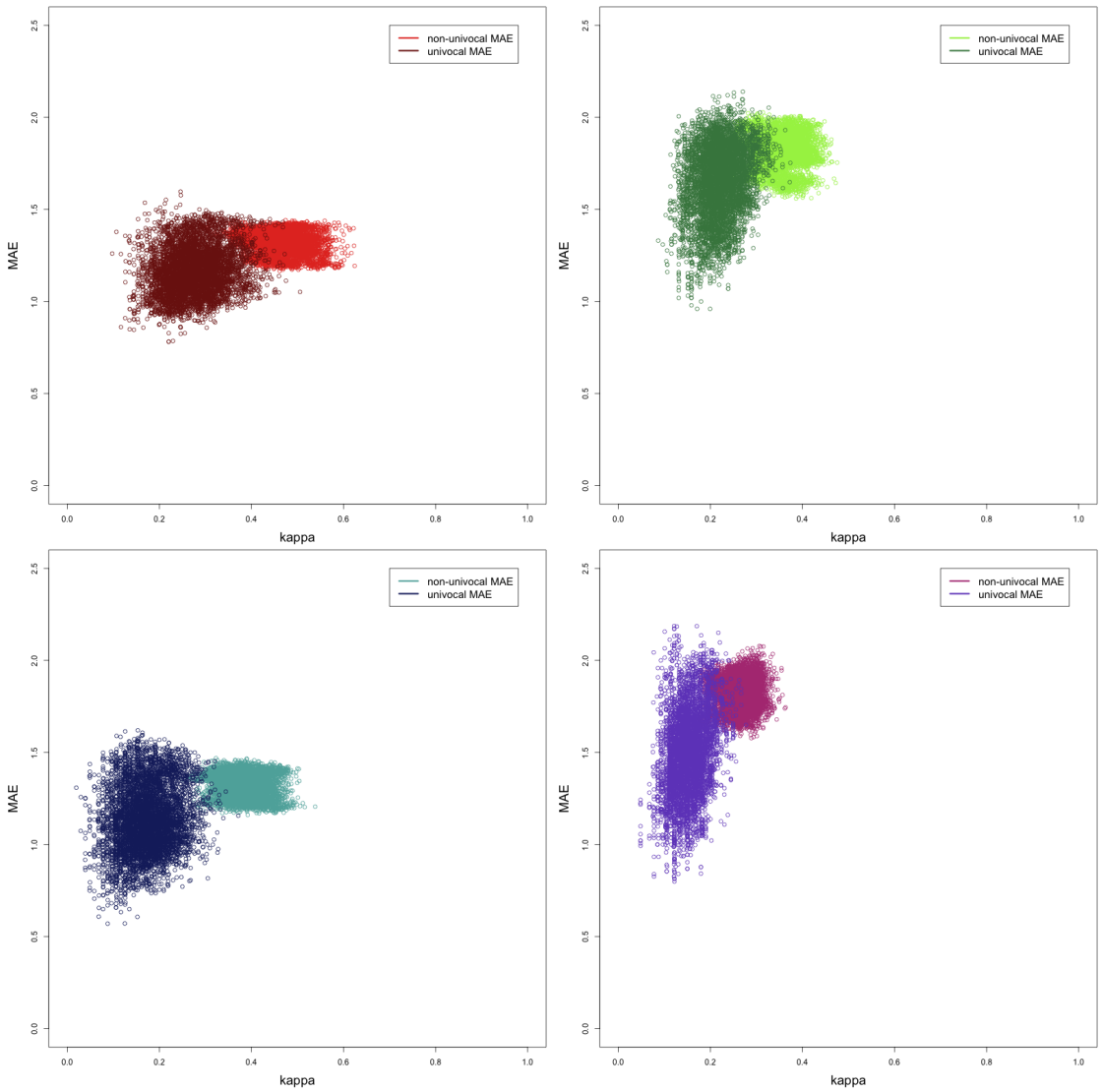


Figure 8.8: Diversity vs. error diagrams for `ERA` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

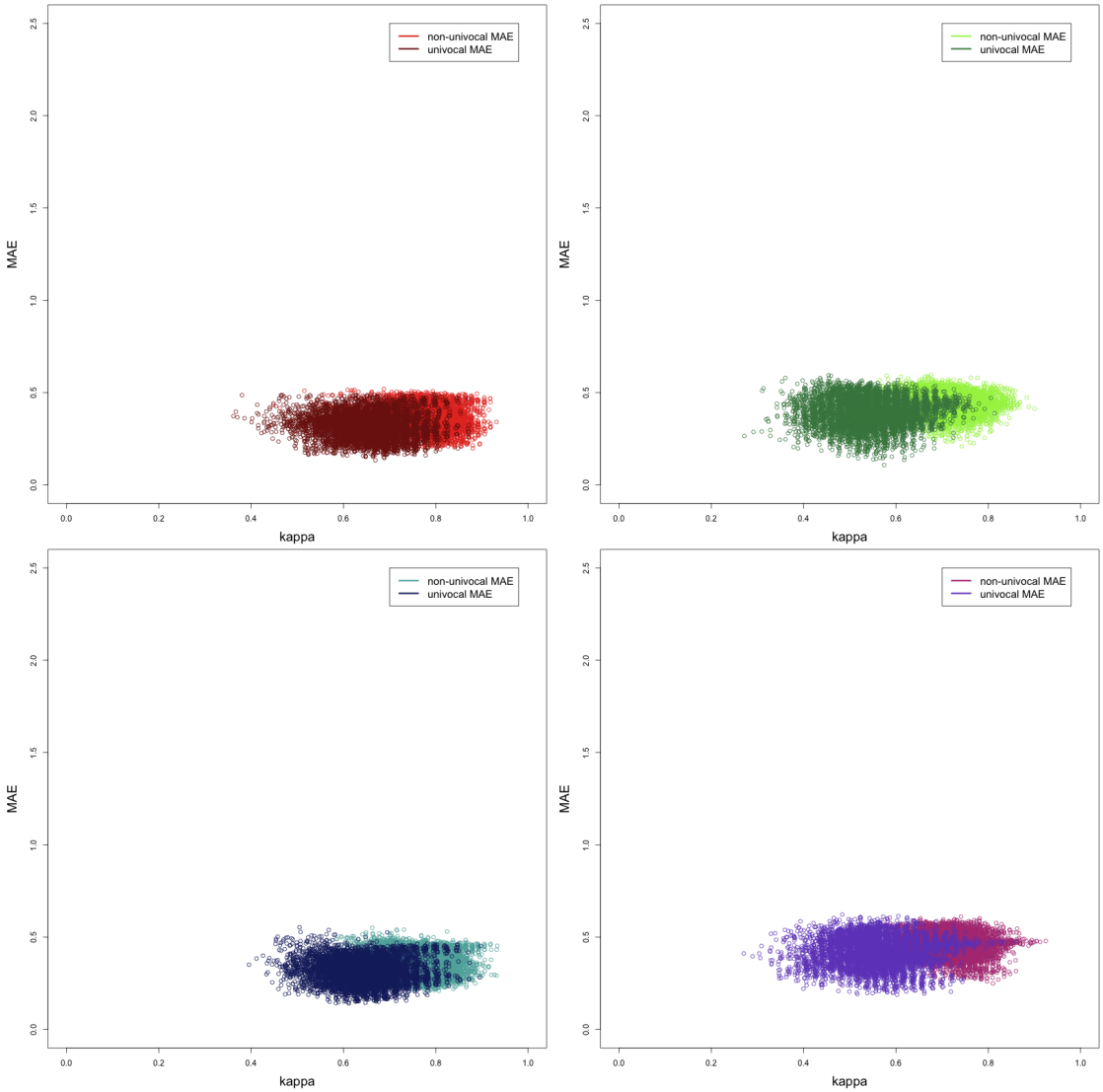


Figure 8.9: Diversity vs. error diagrams for `ESL` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

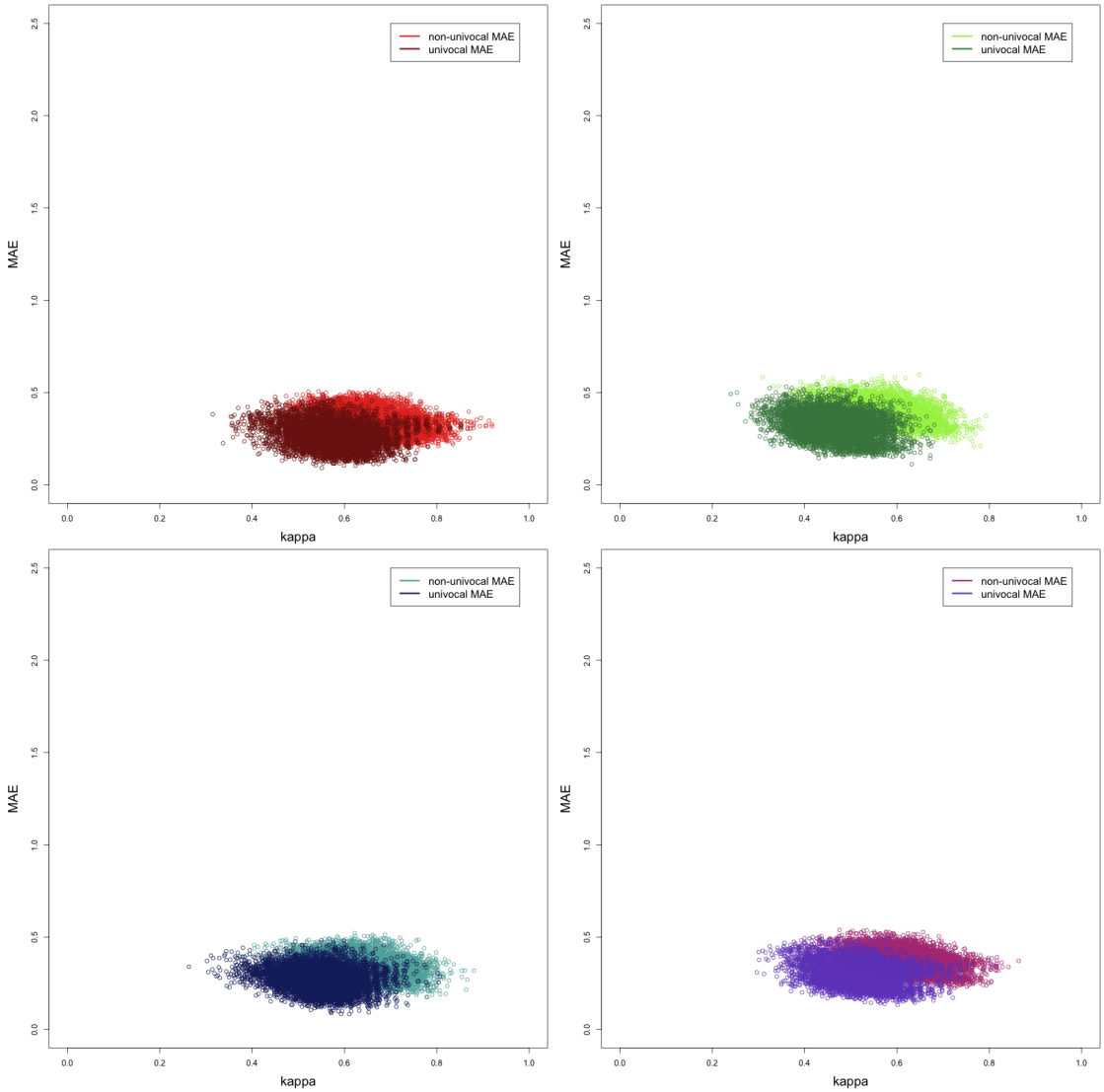


Figure 8.10: Diversity vs. error diagrams for `housing` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

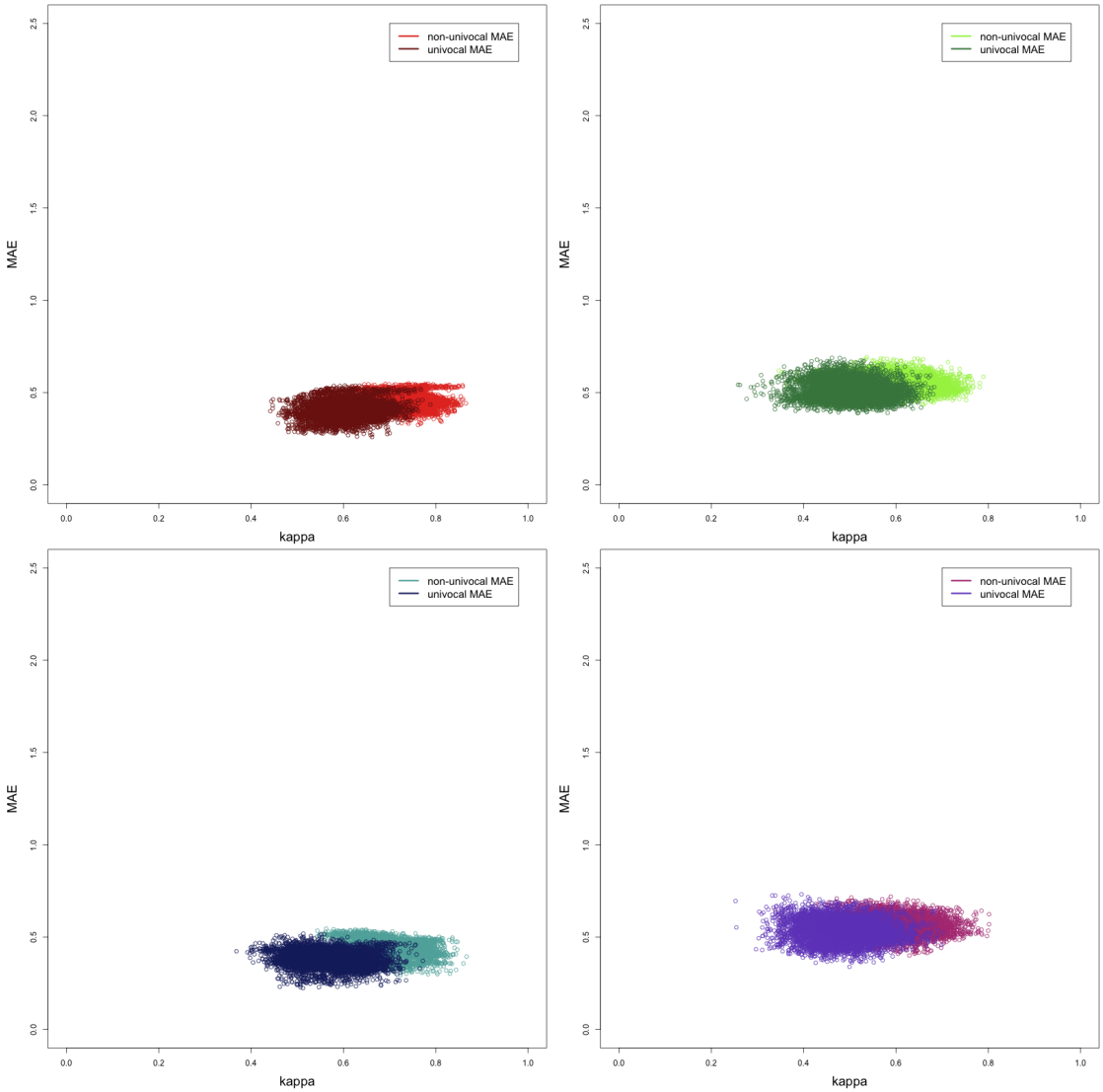


Figure 8.11: Diversity vs. error diagrams for `LEV` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

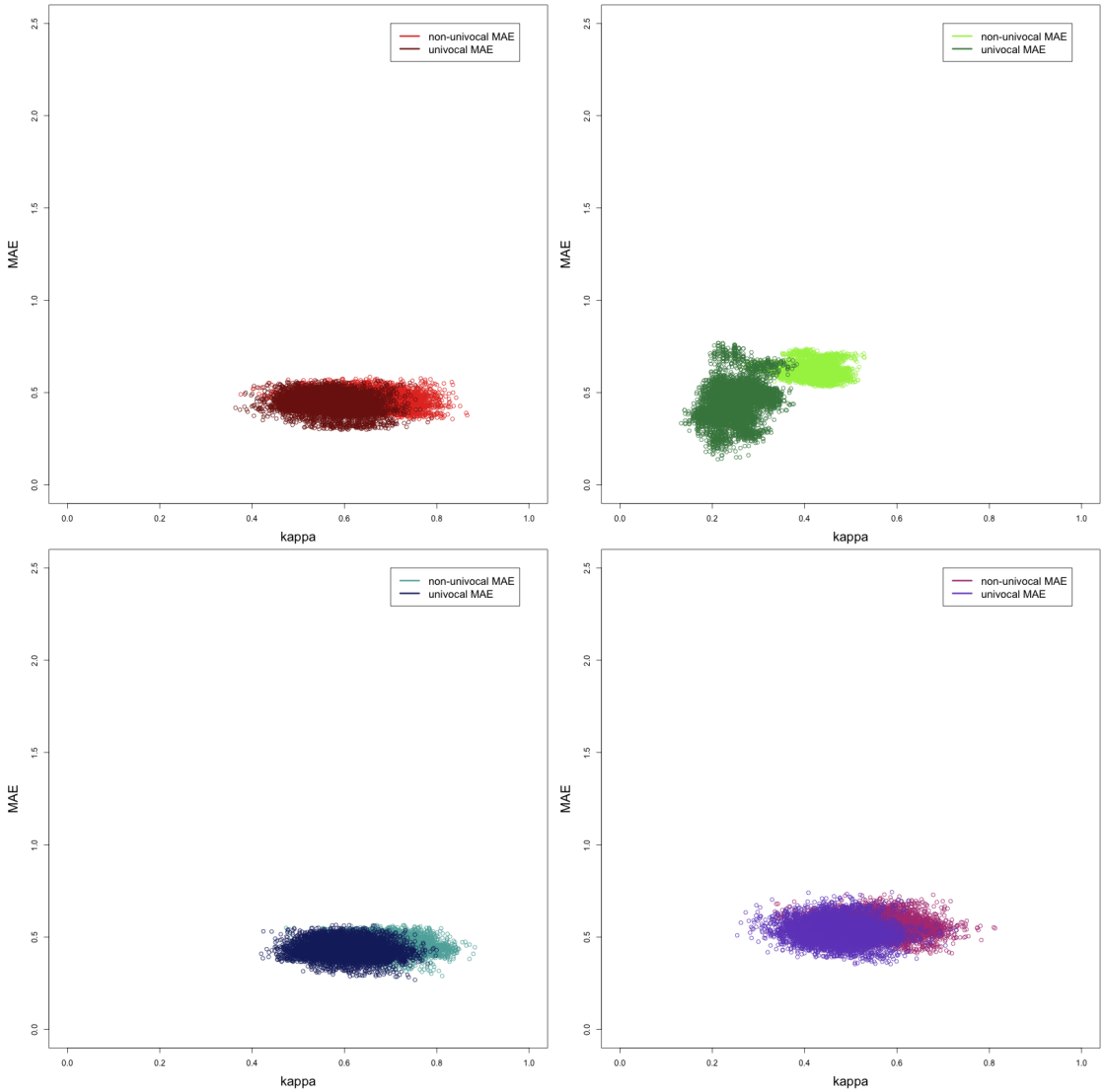


Figure 8.12: Diversity vs. error diagrams for `SWD` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

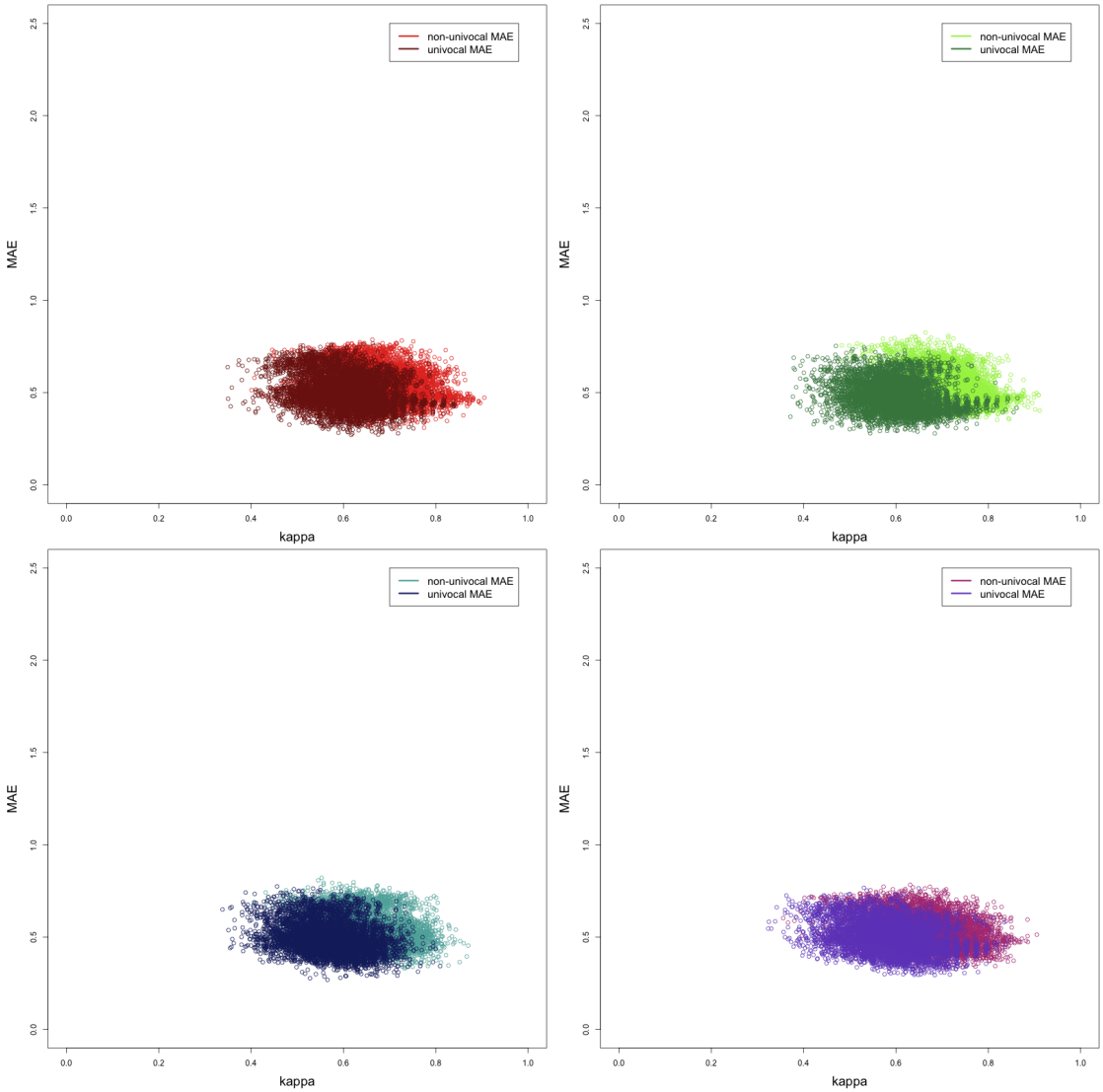


Figure 8.13: Diversity vs. error diagrams for `windsor` data set using standard bagging with monotonic VC-DomLEM (top left), random monotonic VC-DomLEM (top right), and VC-bagging with monotonic VC-DomLEM (bottom left), random monotonic VC-DomLEM (bottom right).

Bibliography

Aha, D. W. and Kibler, D. (1989). Noise-tolerant instance-based learning algorithms. In *IJCAI'89: Proceedings of the 11th international joint conference on Artificial intelligence*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 794–799.

Aha, D. W. and Kibler, D. (1991). Instance-based learning algorithms. *Machine Learning*, **6** 37–66.

Alpigini, J. J., Peters, J. F., Skowronek, J. and Zhong, N. (eds.) (2002). *Rough Sets and Current Trends in Computing, Third International Conference, RSCTC 2002, Malvern, PA, USA, October 14-16, 2002, Proceedings*, vol. 2475 of *Lecture Notes in Computer Science*. Springer.

Altendorf, E., Restificar, A. and Dietterich, T. (2005). Learning from sparse data by exploiting monotonicity constraints. In *Proceedings of the Proceedings of the Twenty-First Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-05)*. AUAI Press, Arlington, Virginia, 18–26.

Anderson, J. A. (1984). Regression and ordered categorical variables. *Journal of the Royal Statistical Society. Series B (Methodological)*, **46** 1–30. URL `http://dx.doi.org/10.2307/2345457`.

Anderson, J. A. and Philips, P. R. (1981). Regression, discrimination and measurement models for ordered categorical variables. *Applied Statistica*, **30** 22–31.

Ayer, M., Brunk, H. D., Ewing, G. M., Reid, W. T. and Silverman, E. (1955). An empirical distribution function for sampling with incomplete information. *Annals of Mathematical Statistics*, **4** 641–647.

Bazan, J. G. (1998). Discovery of decision rules by matching new objects against data tables. In Polkowski and Skowron (1998), 521–528.

Ben-David, A. (1992). Automatic generation of symbolic multi-attribute ordinal knowledge-based dss's: methodology and applications. *Decision Sciences*, **23** 1357–1372.

Ben-David, A. (1995). Monotonicity maintenance in information-theoretic machine learning algorithms. *Machine Learning*, **19** 29–43.

Ben-David, A. and Jagerman, D. (1997). Evaluation of the number of consistent multiattribute classification rules. *Engineering Applications of Artificial Intelligence*, **10** 205–211.

Ben-David, A., Sterling, L. and Pao, Y.-H. (1989). Learning and classification of monotonic ordinal concepts. *Computational Intelligence*, **5** 45–49.

Ben-David, A., Sterling, L. and Tran, T. (2009). Adding monotonicity to learning algorithms may impair their accuracy. *Expert Systems with Applications*, **36** 6627–6634.

Błaszczyński, J., Greco, S. and Słowiński, R. (2007a). Multi-criteria classification - a new scheme for application of dominance-based decision rules. *European Journal of Operational Research*, **181** 1030–1044.

Błaszczyński, J., Greco, S. and Słowiński, R. (accepted for publication 2010). Ordinal and non-ordinal classification using monotonic rules. *8th International Conference of Modeling and Simulation - MOSIM'10 - May 10-12, 2010.*

Błaszczyński, J., Greco, S., Słowiński, R. and Szeląg, M. (2006). On Variable Consistency Dominance-based Rough Set Approaches. In *Rough Sets and Current Trends in Computing* (S. Greco, Y. Hata, S. Hirano, M. Inuiguchi, S. Miyamoto, H. S. Nguyen and R. Słowiński, eds.), vol. 4259 of *Lecture Notes in Computer Science*. Springer, 191–202.

Błaszczyński, J., Greco, S., Słowiński, R. and Szeląg, M. (2007b). Monotonic variable consistency rough set approaches. In *Rough Sets and Knowledge Technology* (Y. Y. Yao, P. Lingras, W.-Z. Wu, M. Szczuka, N. J. Cercone and D. Ślęzak, eds.), vol. LNAI 4481 of *Lecture Notes in Computer Science*. Springer, 126–133.

Błaszczyński, J., Greco, S., Słowiński, R. and Szeląg, M. (2009a). Monotonic variable consistency rough set approaches. *International Journal of Approximate Reasoning*, **50** 979–999.

Błaszczyński, J. and Słowiński, R. (2003). Incremental induction of satisfactory decision rules from dominance based rough approximations. In *Proceedings of The International Workshop on Rough Sets in Knowledge Discovery and Soft Computing* (A. Skowron and M. Szczuka, eds.). 40–51.

Błaszczyński, J., Słowiński, R. and Stefanowski, J. (2009b). Feature set-based consistency sampling in bagging ensembles. In *From Local Patterns To Global Models (LEGO), ECML/PKDD Workshop*. 19–35.

Błaszczyński, J., Słowiński, R. and Stefanowski, J. (accepted for publication 2009). Variable consistency bagging ensembles. *Transactions on Rough Sets, Lecture Notes in Computer Science, Springer*.

Błaszczyński, J., Słowiński, R. and Szeląg, M. (2009c). Sequential covering rule induction algorithm for variable consistency rough set approaches. *submitted to Information Sciences*.

Błaszczyński, J., Słowiński, R. and Szeląg, M. (2009d). VC-DomLEM: Rule induction algorithm for variable consistency rough set approaches. Tech. Rep. RA-07-09, Poznań University of Technology.

Błaszczyński, J., Słowiński, R. and Szeląg, M. (accepted for publication 2010). Probabilistic rough set approaches to ordinal classification with monotonicity constraints. *Inernational Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2010), Lecture Notes in Artificial Intelligence, Springer*.

Błaszczyński, J., Stefanowski, J. and Zając, M. (2009e). Ensembles of abstaining classifiers based on rule sets. In *ISMIS* (J. Rauch, Z. W. Ras, P. Berka and T. Elomaa, eds.), vol. 5722 of *Lecture Notes in Computer Science*. Springer, 382–391.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, **24** 123–140.

Breiman, L. (1998). Arcing classifiers. *The Annals of Statistics*, **26** 801–824.

Breiman, L. (1999). Pasting small votes for classification in large databases and on-line. *Machine Learning*, **36** 85–103.

Breiman, L. (2001). Random forests. *Machine Learning*, **45** 5–32.

Breiman, L., Friedman, J. H., Olshen, R. A. and Stone, C. J. (1984). *Classification and Regression Trees*. Wadsworth.

Brunk, H. D. (1955). Maximum likelihood estimates of monotone parameters. *Annals of Mathematical Statistics*, **4** 607–616.

Burdakov, O., Sysoev, O., Grimvall, A. and Hussian., M. (2006). An o(n^2) algorithm for isotonic regression. In *Large-Scale Nonlinear Optimization*, vol. 83 of *Nonconvex Optimization and Its Applications*. Springer-Verlag, 25–33.

Cao-Van, K. (2003). *Supervised ranking - from semantics to algorithms*. Ph.D. thesis, Ghent University, CS Department.

Chlebus, B. S. and Nguyen, S. H. (1998). On finding optimal discretizations for two attributes. In *RSCTC '98: Proceedings of the First International Conference on Rough Sets and Current Trends in Computing*. Springer-Verlag, London, UK, 537–544.

Chu, W. and Keerthi, S. S. (2005). New approaches to support vector ordinal regression. In *ICML* (L. D. Raedt and S. Wrobel, eds.), vol. 119 of *ACM International Conference Proceeding Series*. ACM, 145–152.

Clark, P. and Boswell, R. (1991). Rule induction with CN2: some recent improvements. In *In Proceedings of the Fifth European Working Session on Learning*. Springer-Verlag, 151–163.

Clark, P. and Niblett, T. (1989). The CN2 induction algorithm. *Machine Learning*, **3** 261–283.

Cohen, W. W. (1995). Fast effective rule induction. In *In Proceedings of the Twelfth International Conference on Machine Learning*. Morgan Kaufmann, 115–123.

Cohen, W. W. and Singer, Y. (1999). A simple, fast, and effective rule learner. In *In Proceedings of the Sixteenth National Conference on Artificial Intelligence*. AAAI Press, 335–342.

Daniëls, H. and Kamp, B. (1999). Application of mlp networks to house pricing and bond rating. *Neural Computing and Applications*, **8** 226–234.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, **7** 1–30.

Dietterich, T. (1998). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, **40** 139–157.

Dougherty, J., Kohavi, R. and Sahami, M. (1995). Supervised and unsupervised discretization of continuous features. In *Proceedings of ICML '95*. Morgan Kaufmann, 194–202.

Doumpos, M. and Pasiouras, F. (2005). Developing and testing models for replicating credit ratings: A multicriteria approach. *Computational Economics*, **25** 327–341.

Doumpos, M. and Zopounidis, C. (2004). Developing sorting models using preference disaggregation analysis: An experimental investigation. *European Journal of Operational Research*, **154** 585–598.

Düntsch, I. and Gediga, G. (1998). Uncertainty measures of rough set prediction. *Artificial Intelligence*, **1** 109–137.

Fayyad, U. M. and Irani, K. B. (1993). Multi-interval discretization of continuous-valued attributes for classification learning. In *Proceedings of International Joint Conferences on Artificial Intelligence*. 1022–1029.

Feelders, A. J. and Pardoel, M. (2003). Pruning for monotone classification trees. In *IDA* (M. R. Berthold, H.-J. Lenz, E. Bradley, R. Kruse and C. Borgelt, eds.), vol. 2810 of *Lecture Notes in Computer Science*. Springer, 1–12.

Figueira, J., Greco, S. and Ehrgott, M. (eds.) (2005). *Multiple Criteria Decision Analysis: State of the Art Surveys*, vol. 78 of *International Series in Operations Research & Management Science*. Springer-Verlag, Berlin.

Fitelson, B. (2001). *Studies in Bayesian Confirmation Theory*. Ph.D. thesis, University of Wisconsin-Madison.

Fitelson, B. (2007). Likelihoodism, Bayesianism, and relational confirmation. *Synthese*, **156** 473–489.

Franke, J. and Mandler, E. (1992). A comparison of two approaches for combining the votes ofcooperating classifiers. *Pattern Recognition, 1992. Vol.II. Conference B: Pattern Recognition Methodology and Systems, Proceedings., 11th IAPR International Conference on* 611–614.

Freund, Y., Schapire, R. E., Singer, Y. and Warmuth, M. K. (1997). Using and combining predictors that specialize. In *STOC '97: Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*. ACM, New York, NY, USA, 334–343.

Friedman, J. H. (2006). Recent advances in predictive (machine) learning. *Journal of Classification*, **23** 175–197.

Friedman, J. H. and Popescu, B. E. (2008). Predictive learning via rule ensembles. *Annals of applied statistics*, **2** 916–954.

Fürnkranz, J. (1999). Separate-and-conquer rule learning. *Artificial Intelligence Review*, **13** 3–54.

Fürnkranz, J. and Widmer, G. (1994). Incremental reduced error pruning. In *Proceedings of ICML' 94*. 70–77.

Giove, S., Greco, S., Matarazzo, B. and Słowiński, R. (2002). Variable consistency monotonic decision trees. In Alpigini et al. (2002), 247–254.

Greco, S., Hata, Y., Hirano, S., Inuiguchi, M., Miyamoto, S., Nguyen, H. S. and Słowiński, R. (eds.) (2006). *Rough Sets and Current Trends in Computing, 5th International Conference, RSCTC 2006, Kobe, Japan, November 6-8, 2006, Proceedings*, vol. 4259 of *Lecture Notes in Computer Science*. Springer.

Greco, S., Kadziński, M. and Słowiński, R. (to appear 2010). The most representative value function in robust multiple criteria sorting. *Computers & Operations Research*.

Greco, S., Matarazzo, B. and Słowiński, R. (1995). Rough set approach to multiattribute choice and ranking problems. ICS Research Report 38/95, Warsaw University of Technology.

Greco, S., Matarazzo, B. and Słowiński, R. (1998a). A new rough set approach to evaluation of bankruptcy risk. In *Opertational Tools in the Management of Financial Risk* (C. Zopounidis, ed.). Kluwer Academic Publishers, Boston, 121–136.

Greco, S., Matarazzo, B. and Slowinski, R. (1998b). A new rough set approach to multicriteria and multiattribute classification. In Polkowski and Skowron (1998), 60–67.

Greco, S., Matarazzo, B. and Slowinski, R. (1999a). Rough approximation of a preference relation by dominance relations. *European Journal of Operational Research*, **117** 63–83. URL `http://ideas.repec.org/a/eee/ejores/v117y1999i1p63-83.html`.

Greco, S., Matarazzo, B. and Słowiński, R. (1999b). The use of rough sets and fuzzy sets in MCDM. In *Advances in MCDM models, Algorithms, Theory, and Applications* (T. Gal, T. Stewart and T. Hanne, eds.). Kluwer Academic, Dordrecht, 14.1–14.59.

Greco, S., Matarazzo, B. and Słowiński, R. (2001a). Rough sets theory for multicriteria decision analysis. *European Journal of Operational Research*, **129** 1–47.

Greco, S., Matarazzo, B. and Słowiński, R. (2002a). Rough approximation by dominance relations. *International Journal of Intelligent Systems*, **17** 153–171.

Greco, S., Matarazzo, B. and Słowiński, R. (2002b). Rough sets methodology for sorting problems in presence of multiple attributes and criteria. *European Journal of Operational Research*, **138** 247–259.

Greco, S., Matarazzo, B. and Słowiński, R. (2005a). Decision rule approach. In *Multiple Criteria Decision Analysis: State of the Art Surveys* (J. Figueira, S. Greco and M. Ehrogott, eds.), vol. 78 of *International Series In Operations Research & Management Science*. Springer New York, 507–555.

Greco, S., Matarazzo, B. and Słowiński, R. (2005b). Rough membership and Bayesian confirmation measures for parameterized rough sets. In *Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (1)* (D. Ślęzak, G. Wang, M. S. Szczuka, I. Düntsch and Y. Yao, eds.), vol. 3641 of *Lecture Notes in Computer Science*. Springer, 314–324.

Greco, S., Matarazzo, B. and Słowiński, R. (2008a). Dominance-based rough set approach to interactive multiobjective optimization. In *Multiobjective Optimization* (J. Branke, K. Deb, K. Miettinen and R. Słowiński, eds.), vol. 5252 of *Lecture Notes in Computer Science*. Springer, 121–155.

Greco, S., Matarazzo, B. and Słowiński, R. (2008b). Granular computing for reasoning about ordered data: the dominance-based rough set approach. In *Handbook of Granular Computing* (W. Pedrycz, A. Skowron and V. Kreinovich, eds.), chap. 15. John Wiley & Sons, Ltd.

Greco, S., Matarazzo, B. and Słowiński, R. (2008c). Parameterized rough set model using rough membership and Bayesian confirmation measures. *International Journal of Approximate Reasoning*, **49** 285–300.

Greco, S., Matarazzo, B. and Słowiński, R. (2010). Algebra and topology for dominance-based rough set approach. In *Advances in Intelligent Information Systems* (Z. Raś and W. Ribarsky, eds.), vol. 265 of *Studies in Computational Intelligence*. Springer, 43–78.

Greco, S., Matarazzo, B., Słowiński, R. and Stefanowski, J. (2000a). An algorithm for induction of decision rules consistent with the dominance principle. In Ziarko and Yao (2001), 304–313.

Greco, S., Matarazzo, B., Słowiński, R. and Stefanowski, J. (2000b). Variable consistency model of dominance-based rough sets approach. In Ziarko and Yao (2001), 170–181.

Greco, S., Matarazzo, B., Słowiński, R. and Stefanowski, J. (2001b). An algorithm for induction of decision rules consistent with dominance principle. *Rough Sets and Current Trends in Computing. LNAI*, **2005** 304–313.

Greco, S., Mousseau, V. and Słowiński, R. (2009). Multicriteria sorting with a set of value functions. *Cahier du LAMSADE 282, Universite de Paris Dauphine* (to appear in European Journal of Operational Research).

Greco, S., Słowiński, R. and Pawlak, Z. (2004). Can Bayesian confirmation measures be useful for rough set decision rules? *Engineering Applications of Artificial Intelligence*, **17** 345–361.

Greco, S., Słowiński, R. and Yao, Y. (2007). Bayesian decision theory for dominance-based rough set approach. In Yao et al. (2007), 134–141.

Grzymała-Busse, J. W. (1992). LERS - a system for learning from examples based on rough sets. In *Intelligent Decision Support: Handbook of Applications and Advances of the Rough Sets Theory* (R. Słowiński, ed.). Kluwer Academic Publishers, 3–18.

Grzymała-Busse, J. W. (1994). Managing uncertainty in machine learning from examples. *Third Intelligent Information Systems Workshop, Wigry, Poland, IPI PAN Press* 70–84.

Grzymała-Busse, J. W. (1997). A new version of the rule induction system LERS. *Fundamenta Informaticae*, **31** 27–39.

Grzymała-Busse, J. W. and Lakshmanan, A. (1996). LEM2 with interval extension: an induction algorithm for numerical attributes. In *Proceedings of the Fourth*

International Workshop on Rough Sets, Fuzzy Sets and Machine Discovery. Tokyo University Press, 67–73.

Grzymała-Busse, J. W. and Stefanowski, J. (2001). Three discretization methods for rule induction. *International Journal of Intelligent Systems*, **16** 29–38.

Grzymała-Busse, J. W. and Wang, A. Y. (1997). Modified algorithms LEM1 and LEM2 for rule induction from data with missing attribute values. In *Proc. of 5th Int. Workshop on Rough Sets and Soft Computing (RSSC'97) at JCIS'97*. 69–72.

Grzymała-Busse, J. W. and Zou, X. (1998). Classification strategies using certain and possible rules. In *RSCTC '98: Proceedings of the First International Conference on Rough Sets and Current Trends in Computing*. Springer-Verlag, London, UK, 37–44.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I. H. (2009). The WEKA data mining software: An update. *ACM SIGKDD Explorations Newsletter*, **11** 10–18.

Han, J. and Kamber, M. (2006). *Data Mining: Concepts and Techniques*. Morgan Kaufmann.

Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer.

Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20** 832–844. URL `citeseer.ist.psu.edu/ho98random.html`.

Hu, Q., Daren, Y., Zongxia, X. and Liu, J. (2006). Fuzzy probabilistic approximation spaces and their information measures. *IEEE Transactions on Fuzzy Systems*, **14** 191–201.

Iba, W., Wogulis, J. and Langley, P. (1988). Trading off simplicity and coverage in incremental concept learning. In *Proceedings of the 5th International Conference on Machine Learning (ICML 1988)*. Kaufmann, 73–79.

Inuiguchi, M. (2006). Structure-based attribute reduction in variable precision rough set models. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, **10** 657–665.

Inuiguchi, M. and Yoshioka, Y. (2006). Variable-precision dominance-based rough set approach. In Greco et al. (2006), 203–212.

Jacquet-Lagrèze, E. and Siskos, Y. (1982). Assessing a set of additive utility functions for multicriteria decision making: the UTA method. *European Journal of Operational Research*, **10** 151–164.

Kittler, J., Hatef, M., Duin, R. P. W. and Matas, J. (1998). On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20** 226–239.

Kononenko, I. and Kukar, M. (2007). *Machine Learning and Data Mining*. Horwood Pub.

Koop, G. (2000). *Analysis of Economic Data*. John Wiley & Sons, Ltd.

Kotłowski, W. (2009). *Statistical Approach to Ordinal Classification with Monotonicity Constraints*. Ph.D. thesis, Poznań University of Technology.

Kotłowski, W., Dembczyński, K., Greco, S. and Słowiński, R. (2008). Stochastic dominance-based rough set model for ordinal classification. *Information Sciences*, **178** 4019–4037.

Kotłowski, W. and Słowiński, R. (2008). Statistical approach to ordinal classification with monotonicity constraints. In *Preference Learning ECML/PKDD 2008 Workshop*.

Kotłowski, W. and Słowiński, R. (2009). Rule learning with monotonicity constraints. *In Proceedings of ICML* 537–544.

Krawiec, K., Słowiński, R. and Vanderpooten, D. (1998). Learning of decision rules from similarity based rough approximations. In *Rough Sets in Knowledge Discovery*. Physica Verlag, 37–54.

Kuncheva, L. (2004). *Combining Pattern Classifiers. Methods and Algorithms*. Wiley.

Kuncheva, L. I., Skurichina, M. and Duin, R. P. W. (2002). An experimental study on diversity for bagging and boosting with linear classifiers. *Information Fusion*, **3** 245–258.

Le, Q. V., Smola, A. J. and Gärtner, T. (2006). Simpler knowledge-based support vector machines. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*. ACM, New York, NY, USA, 521–528.

Makino, K., Suda, T., Yano, K. and Ibaraki, T. (1996). Data analysis by positive decision trees. In *CODAS*. 257–264. URL `citeseer.ist.psu.edu/makino99data.html`.

Margineantu, D. D. and Dietterich, T. G. (1997). Pruning adaptive boosting. In *ICML '97: Proceedings of the Fourteenth International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 211–218.

Marinakis, Y., Marinaki, M., Doumpos, M., Matsatsinis, N. F. and Zopounidis, C. (2008). Optimization of nearest neighbor classifiers via metaheuristic algorithms for credit risk assessment. *Journal of Global Optimization*, **42** 279–293.

McCullagh, P. (1980). Regression models for ordinal data. *Journal of Royal Statistical Society. Series B (Methodological)*, **42** 109–142.

Michalski, R. S. (1969). On the quasi-minimal solution of the covering problem. In *Proceedings of the 5th International Symposium on Information Processing (FCIP-69)*, vol. A3 (Switching Circuits). Bled, Yugoslavia, 125–128.

Michalski, R. S. (1993). A theory and methodology of inductive learning. In *Readings in knowledge acquisition and learning: automating the construction and improvement of expert systems*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 323–348.

Michalski, R. S. and Kaufman, K. A. (1998). Data mining and knowledge discovery: A review of issues and a multistrategy approach. In *Machine Learning and Data Mining: Methods and Applications*. John Wiley & Sons, Inc., 71–112.

Nguyen, H. and Nguyen, S. (1998). Discretization methods for data mining. In *Rough Sets in Knowledge Discovery* (L. Polkowski and A. Skowron, eds.). Physica-Verlag, 451–482.

Nguyen, H. S. (2006). Approximate boolean reasoning: Foundations and applications in data mining. *Transactions on Rough Sets V*, **4100** 334–506.

Pagallo, G. and Haussler, D. (1990). Boolean feature discovery in empirical learning. *Machine Learning*, **5** 71–99.

Panov, P. and Dzeroski, S. (2007). Combining bagging and random subspaces to create better ensembles. *Proceedings IDA 2007 Conference, Springer LNCS*, **4723** 118–129.

Patrice, L., Olivier, D. and Christine, D. (2000). Different ways of weakening decision trees and their impact on classification accuracy of DT combination. In *Multiple Classifier Systems*, vol. 1857 of *Lecture Notes in Computer Science*. Springer Berlin, Heidelberg, 200–209.

Pawlak, Z. (1982). Rough sets. *International Journal of Information & Computer Sciences*, **11** 341–356.

Pawlak, Z. (1991). *Rough Sets: Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers.

Pawlak, Z. (2004). Some issues on rough sets. *Transactions on Rough Sets*, **3100** 1–58.

Pawlak, Z. and Skowron, A. (1994). Rough membership functions. In *Advances in the Dempster-Shafer theory of evidence*. John Wiley & Sons, Inc., New York, NY, USA, 251–271.

Platt, J. (1998). Machines using sequential minimal optimization. In *Advances in Kernel Methods – Support Vector Learning,* (B. Schoelkopf, C. Burges and A. Smola, eds.). MIT Press, Cambridge, MA.

Polkowski, L. and Skowron, A. (eds.) (1998). *Rough Sets and Current Trends in Computing, First International Conference, RSCTC'98, Warsaw, Poland, June 22-26, 1998, Proceedings*, vol. 1424 of *Lecture Notes in Computer Science*. Springer.

Popova, V. N. (2004). *Knowledge Discovery and Monotonicity*. Ph.D. thesis, Erasmus University Rotterdam.

Potharst, R., Bioch, J. and van Dordregt, R. (1988). Quasi-monotone decision trees for ordinal classification. Tech. Rep. Tech. Rep. EUR-FEW-CS-98-01, Erasmus University Rotterdam.

Potharst, R. and Bioch, J. C. (2000). Decision trees for ordinal classification. *Intelligent Data Analysis*, **4** 97–111.

Potharst, R. and Feelders, A. J. (2002). Classification trees for problems with monotonicity constraints. *ACM SIGKDD Explorations Newsletter*, **4** 1–10.

Qian, Y., Liang, J., Dang, C., Zhang, H. and Jianmin, M. (2008a). Consistency measure, inclusion degree and fuzzy measure in decision tables. *Fuzzy Sets and Systems*, **159** 2353–2377.

Qian, Y., Liang, J., Dang, C., Zhang, H. and Jianmin, M. (2008b). On the evaluation of the decision performance of an incomplete decision table. *Data & Knowledge Engineering*, **65** 374–400.

Quinlan, J. R. (1990). Learning logical definitions from relations. *Machine Learning*, **5** 239–266.

Quinlan, J. R. (1992). *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA.

Quinlan, J. R. and Cameron-Jones, R. M. (1993). FOIL: A midterm report. In *Proceedings of the European Conference on Machine Learning*. Springer-Verlag, 3–20.

Roy, B. (1996). *Multicriteria Methodology for Decision Aiding*. Dordrecht: Kluwer Academic Publishers.

Rückert, U. and Kramer, S. (2004). Towards tight bounds for rule learning. In *ICML '04: Proceedings of the twenty-first international conference on Machine learning*. ACM, New York, NY, USA, 90–98.

Rückert, U. and Kramer, S. (2006). A statistical approach to rule learning. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*. ACM, New York, NY, USA, 785–792.

Schapire, R. E. and Singer, Y. (1999). Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, **37** 297–336.

Sill, J. (1998). Monotonic networks. *Advances in Neural Information Processing Systems*, **10** 661–667.

Sill, J. and Abu-Mostafa (1997). Monotonicity hints. *Advances in Neural Information Processing Systems*, **9** 634–640.

Skowron, A. (1993). Boolean reasoning for decision rules generation. In *ISMIS* (H. J. Komorowski and Z. W. Ras, eds.), vol. 689 of *Lecture Notes in Computer Science*. Springer, 295–305.

Ślęzak, D. (2005). Rough sets and Bayes factor. *Transactions on Rough Sets*, **3400** 202–229.

Ślęzak, D. and Ziarko, W. (2005). The investigation of the Bayesian rough set model. *International Journal of Approximate Reasoning*, **40** 81–91.

Słowiński, R., Greco, S. and Matarazzo, B. (2002a). Rough set analysis of preference-ordered data. In *Rough Sets and Current Trends in Computing*, vol. 2475/2002. Springer Berlin / Heidelberg, 949–950.

Słowiński, R., Greco, S. and Matarazzo, B. (2002b). Rough set analysis of preference-ordered data. In Alpigini et al. (2002), 44–59.

Słowiński, R., Greco, S. and Matarazzo, B. (2005). Rough set based decision support. In *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques* (E. K. Burke and G. Kendall, eds.), chap. 16. Springer-Verlag, New York, 475–527.

Słowiński, R., Greco, S. and Matarazzo, B. (2009). Rough sets in decision making. In *Encyclopedia of Complexity and Systems Science* (R. A. Meyers, ed.). Springer-Verlag, New York, 7753–7786.

Słowiński, R. and Stefanowski, J. (1994). Rough classification with valued closeness relation. In *New Approaches in Classification an Data Analysis* (E. D. et al., ed.). Springer-Verlag, 482–489.

Słowiński, R., Stefanowski, J., Greco, S. and Matarazzo, B. (2000). Rough set based processing of inconsistent information in decision analysis. *Control & Cybernetics*, **29** 27–41.

Stefanowski, J. (1995). Using valued closeness relation in classification support of new objects. In *Soft computing: rough sets, fuzzy logic, neural networks uncertainty management, knowledge discovery* (T. Y. Lin and A. Wildberger, eds.). Simulation Councils Inc., San Diego CA, 324–327.

Stefanowski, J. (1998). Handling continuous attributes in discovery of strong decision rules. In Polkowski and Skowron (1998), 394–401.

Stefanowski, J. and Kaczmarek, M. (2004). Integrating attribute selection to improve accuracy of bagging classifiers. In *Proceedings of the AI-METH 2004 Conference - Recent Developments in Artificial Intelligence Methods*. 263–268.

Stefanowski, J. and Wilk, S. (2007). Improving rule based classifiers induced by modlem by selective pre-processing of imbalanced data. In *Proceedings of the RSKD Workshop at ECML/PKDD, Warsaw, 2007*. 54–65.

Susmaga, R., Słowiński, R., Greco, S. and Matarazzo, B. (2000). Generation of reducts and rules in multi-attribute and multi-criteria classification. *Control and Cybernetics*, **29** 969–988.

Weiss, S. M. and Indurkhya, N. (2000). Lightweight rule induction. In *Proceedings of ICML* (P. Langley, ed.). Morgan Kaufmann, 1135–1142.

Wilson, D. R. and Martinez, T. R. (2000). Reduction techniques for instance-basedlearning algorithms. *Machine Learning*, **38** 257–286.

Wong, S. K. M. and Ziarko, W. (1987). Comparison of the probabilistic approximate classification and the fuzzy set model. *Fuzzy Sets and Systems*, **21** 357–362.

Yao, J., Lingras, P., Wu, W.-Z., Szczuka, M. S., Cercone, N. and Ślęzak, D. (eds.) (2007). *Rough Sets and Knowledge Technology, Second International Conference, RSKT 2007, Toronto, Canada, May 14-16, 2007, Proceedings*, vol. 4481 of *Lecture Notes in Computer Science*. Springer.

Yao, Y. (2003). Granular computing for the design of information retrieval support systems. In *Clustering and Information Retrieval* (W. Wu, H. Xiong and S. Shekhar, eds.). Kluwer Academic Publishers, 299–329.

Yao, Y. (2007). Decision-theoretic rough set models. In Yao et al. (2007), 1–12.

Yao, Y. (2008). Probabilistic rough set approximations. *International Journal of Approximate Reasoning*, **49** 255–271.

Ziarko, W. (1993). Variable precision rough set model. *Journal of Computer and System Sciences*, **46** 39–59.

Ziarko, W. (2006). Stochastic approach to rough set theory. In Greco et al. (2006), 38–48.

Ziarko, W. and Yao, Y. Y. (eds.) (2001). *Rough Sets and Current Trends in Computing, Second International Conference, RSCTC 2000 Banff, Canada, October 16-19, 2000, Revised Papers*, vol. 2005 of *Lecture Notes in Computer Science*. Springer.

Zopounidis, C. and Doumpos, M. (1999). A multicriteria decision aid methodology for sorting decision problems: The case of financial distress. *Computational Economics*, **14** 197–218.