Rule Models Discovery in Ordinal Classification with Monotonicity Constraints

Odkrywanie modeli regułowych w problemach klasyfikacji porządkowej z ograniczeniami monotonicznymi

Marcin Szeląg

Institute of Computing Science, Poznań University of Technology

30.10.2019

Outline

Introduction

Problem Statement

There is given a finite set U of objects described on attributes from finite set $A = C \cup D$, where C denotes a set of condition attributes, D denotes a set of decision attributes, and $C \cap D = \emptyset$.

Decision attributes make a partition of set U into a finite number of n disjoint sets of objects, called decision classes. We denote this partition, also called classification, by $\mathcal{X} = \{X_1, \ldots, X_n\}$.

Problem Statement

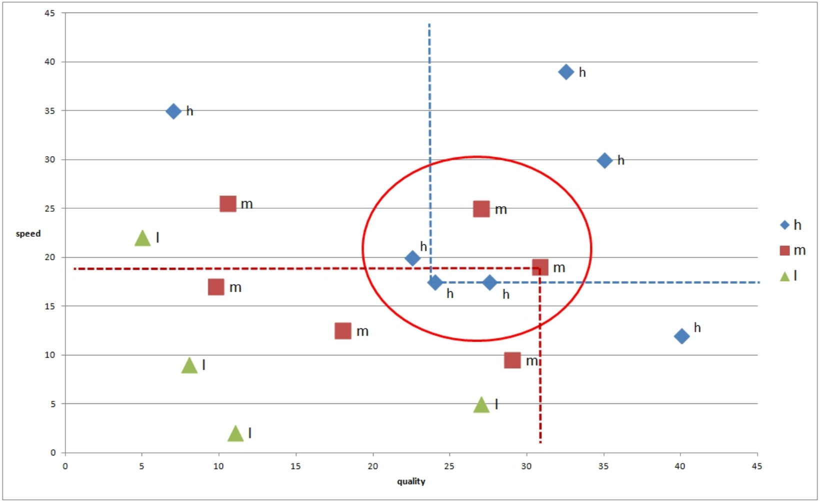
Ordinal classification problem with monotonicity constraints

- Relationship between evaluation of objects on condition attributes and their evaluation on the decision attribute.
- Knowledge about orders (of preferences) on the value sets of the attributes – criteria.
- Semantic correlation: a better evaluation of an object on a condition attribute with other evaluations being fixed should not worsen its evaluation on decision attribute.
- Inconsistency: violation of monotonicity constraints (expressed by semantic correlation).

Examples of monotonicity constraints

- "The lower the price and the higher the quality, the higher the customer's satisfaction".
- "The higher the mass and the lower the distance, the higher the gravity".

Example


45
40
35
30
25
20
15
10
5
0
speed
0
5
10
15
20
25
30
35
40
45
quality
h
m
l
h
h
h
m
l
h
m
m
h
h
h
m
m
l
l
l
h
m
l

Problem Statement

The goal is to build a classifier which, given the evaluations of object $y \in X_i$ on condition attributes, suggests assignment of y to one of the classes from $\mathcal{X}$.

A good classifier is characterized by:

- high accuracy,
- interpretability (i.e., it preserves monotonicity constraints),
- traceability (glass box).

We can estimate accuracy of prediction of the classifier by:

- the percentage of correctly classified objects (PCC),
- the mean absolute difference between index of the class suggested by a classifier and index of the class to which an object belongs; this measure is called mean absolute error (MAE).

General Motivations

- Multi-attribute/multi-criteria classification is an important, non-trivial and practical problem.
- Main dificulty consists in aggregation of different and usually conflicting attributes/criteria; usually such aggregation is performed arbitrary, using weights or aggregation operators like sum, average or distance metrics.
- Need for modeling method that allows to include domain knowledge (like preference orders), can handle possible inconsistencies in data, and avoids any aggregation operators.

Motivations for Application of DRSA

Ordinal classification problem with monotonicity constraints can be effectively solved using Dominance-based Rough Set Approach (DRSA), which:

- can handle inconsistencies in data (preprocessing), resulting e.g. from imprecise of incomplete information,
- takes into account domain knowledge:
  - domains of attributes, i.e. sets of values that an attribute may take while being meaningful for user's perception,
  - division of attributes into condition and decision attributes,
  - preference order in domains of attributes and semantic correlation between attributes, both addressed by the dominance principle,
- works with heterogenous attributes – nominal, ordinal and cardinal (no need of discretization),
- enables to infer decision rule model from decision table (disaggregation-aggregation paradigm).

Advantages of decision rules:

- comprehensible form of knowledge representation,
- can represent any function (more general than utility functions or binary relations),
- resistant to irrelevant attributes (to certain degree),
- do not require aggregation operators,
- support "backtracing",
- can explain past decisions and predict future decisions.

Dominance-based Rough Set Approach

In rough set approaches, learning of a classifier is preceded by rough set analysis of data presented as decision table. It consists in checking the data for possible inconsistencies by calculation of lower approximations of considered sets of objects.

Due to this type of data structuring, one may restrict a priori the set of objects on which the classifier is learned to a subset of sufficiently consistent objects belonging to lower approximations.

This restriction is motivated by a postulate for learning from (sufficiently) consistent data, so that the knowledge gained from this learning is (sufficiently) certain.

Basic notions and definitions

The original Rough Set Approach proposed by Pawlak, called Indiscernibility-based Rough Set Approach (IRSA):

- concerns non-ordinal classification,
- employs indiscernibility relation,
- involves approximations of decision classes X_i.

Dominance-based Rough Set Approach (DRSA) proposed by Greco, Matarazzo and Słowiński:

- concerns ordinal classification with monotonicity constraints,
- employs dominance relation,
- involves approximations of unions of ordered 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 $i = 1, 2, \ldots, n-1$.

Basic notions and definitions

In case of non-ordinal classification handled by IRSA, set of attributes A is composed of regular attributes only. Indiscernibility relation makes a partition of U into disjoint blocks of objects called granules. Moreover, $I(y)$ denotes a set of objects indiscernible with object $y \in U$.

In case of ordinal classification with monotonicity constraints handled by DRSA, among condition attributes from C there is at least one criterion, decision attribute d has preference-ordered value set, and there exists a monotonic relationship between evaluation of objects on criteria and their values on the decision attribute.

The positive dominance cone $D^+(y)$ is composed of objects that for each $q_i \in C$ are not worse than object y.

The negative dominance cone $D^-(y)$ is composed of objects that for each $q_i \in C$ are not better than object y.

In the following, we are going to consider DRSA only.

In order to simplify notation, we use:

- symbol X to denote a set of objects belonging to union of classes $X_i^{\geq}$ or $X_i^{\leq}$,
- symbol $E(y)$ to denote any set $D^+(y)$ or $D^-(y)$, $y \in U$.

Moreover, if X and $E(y)$ are used in the same equation, then for X representing union of ordered classes $X_i^{\geq}$ (resp. $X_i^{\leq}$), $E(y)$ stands for dominance cone $D^+(y)$ (resp. $D^-(y)$).

Basic notions and definitions

The lower approximation of set X is defined as:

$$\underline{X} = \{y \in X : E(y) \subseteq X\}. \tag{1}$$

This definition of the lower approximation appears to be too restrictive in practical applications. Therefore, various probabilistic rough set approaches were proposed which extend the lower approximation of set X by inclusion of objects with sufficient evidence for membership to X. This evidence is quantified by different object consistency measures.

In the following, we consider Variable-Consistency DRSA (VC-DRSA), where the definition of probabilistic lower approximation of set X involves object consistency measure $\Theta_X : U \to \mathbb{R}^+ \cup \{0\}$:

- given a gain-type measure Θ_X and a gain-threshold θ_X:

$$\underline{X} = \{y \in X : \Theta_X(y) \geq \theta_X\}. \tag{2}$$

- given a cost-type measure Θ_X and a cost-threshold θ_X:

$$\underline{X} = \{y \in X : \Theta_X(y) \leq \theta_X\}. \tag{3}$$

Required monotonicity properties of object consistency measures[a]

[a] J. Błaszczyński, S. Greco, R. Słowiński, M. Szeląg, Monotonic Variable Consistency Rough Set Approaches. International Journal of Approximate Reasoning, 50(7), 2009, pp. 979-999

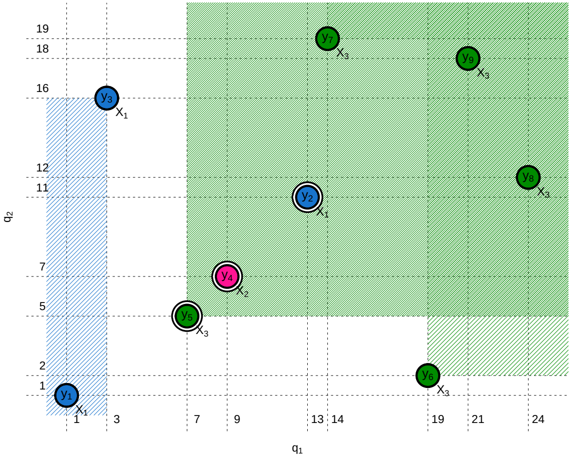
- (m1): Monotonicity w.r.t. set of attributes $P \subseteq C$.
- (m2): Monotonicity w.r.t. set of objects $X \subseteq U$, when set X is augmented by new objects.
- (m3): Monotonicity w.r.t. union of classes $X_i^{\geq} \subseteq U$ and $X_i^{\leq} \subseteq U$.
- (m4): Monotonicity w.r.t. dominance relation D.

Let $X, \neg X \subseteq U$, where $\neg X = U - X$, $y \in U$. We consider the following cost-type object consistency measure ϵ_X:

$$\epsilon_X(y) = \frac{|E(y) \cap \neg X|}{|\neg X|}. \tag{4}$$

This measure has properties (m1), (m2), and (m4).

Basic notions and definitions



$$\epsilon_{X_3^{\geq}}(y_5) = \frac{2}{4}, \quad \epsilon_{X_1^{\leq}}(y_3) = 0$$

We define the positive region of X as:

$$POS(X) = \bigcup_{y \in \underline{X}} E(y). \tag{5}$$

One can observe that $POS(X) \supseteq \underline{X}$.

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

$$NEG(X) = POS(\neg X) - POS(X), \tag{6}$$

$$BND(X) = (U - POS(X)) - NEG(X). \tag{7}$$

Decision Rules

Decision Rules

- Probabilistic lower approximations are basis for induction of (probabilistic) decision rules which are a simple and comprehensive representation of knowledge:

  if elementary conditions **then** decision (prediction).

- Condition part of a rule is a conjunction of elementary conditions concerning individual attributes/criteria.
- Decision part of a rule suggests an assignment to a union of decision classes.
- Rules are characterized by rule consistency measures.
- Rules explain decisions observed in data and can be used to classify a new object to one of the predefined decision classes.

Decision Rules

- Set $\underline{X}$ is the basis for induction of a set R_X of decision rules that suggest assignment to X.
- Each rule from R_X is supported by at least one object from $\underline{X}$, and it covers object(s) from $POS(X)$.
- The elementary conditions (selectors) that form a rule $r_X \in R_X$ are built using evaluations of objects belonging to $\underline{X}$ only.

Induced rules have the following syntax:

$$\textit{if } q_{i_1}(y) \succeq t_{i_1} \wedge \ldots \wedge q_{i_p}(y) \succeq t_{i_p} \wedge q_{i_{p+1}}(y) = t_{i_{p+1}} \wedge \ldots \wedge q_{i_z}(y) = t_{i_z} \quad \textit{then } y \in X_i^{\geq},$$

$$\textit{if } q_{i_1}(y) \preceq t_{i_1} \wedge \ldots \wedge q_{i_p}(y) \preceq t_{i_p} \wedge q_{i_{p+1}}(y) = t_{i_{p+1}} \wedge \ldots \wedge q_{i_z}(y) = t_{i_z} \quad \textit{then } y \in X_i^{\leq},$$

where $q_{i_1}, \ldots, q_{i_p}$ denote criteria, and $q_{i_{p+1}}, \ldots, q_{i_z}$ denote regular attributes; moreover, t_{i_j} denotes a value taken from the value set of attribute q_{i_j}, $i_j \in \{i_1, \ldots, i_z\} \subseteq \{1, \ldots, |C|\}$.
Symbols $\succeq$ and $\preceq$ indicate weak preference and inverse weak preference w.r.t. single criterion, respectively. If $q_{i_j} \in C$ is a gain (cost) criterion, then elementary condition $q_{i_j}(y) \succeq t_{i_j}$ means that the evaluation of object $y \in U$ on criterion q_{i_j} is not worse than t_{i_j}, $i_j \in \{i_1, \ldots, i_p\}$. Elementary conditions for regular attributes are of the type $q_{i_j}(y) = t_{i_j}$, $i_j \in \{i_{p+1}, \ldots, i_z\}$.

Example (chosen rules for windsor dataset)

(nbath $\geq$ 2) & (nstoreys $\geq$ 2) & (air_cond $\geq$ 1) & (desire_loc $\geq$ 1) => **(sale_price $\geq$ 3)**

(lot_size $\geq$ 6240.0) & (nbath $\geq$ 2) & (drive $\geq$ 1) & (air_cond $\geq$ 1) => **(sale_price $\geq$ 3)**

(nstoreys $\geq$ 4) => **(sale_price $\geq$ 2)**

(lot_size $\geq$ 10500.0) & (ngarage $\geq$ 1) => **(sale_price $\geq$ 2)**

(lot_size $\geq$ 10269.0) & (nbed $\geq$ 3) & (ngarage $\geq$ 1) => **(sale_price $\geq$ 2)**

(nbath $\geq$ 3) & (drive $\geq$ 1) => **(sale_price $\geq$ 2)**

(lot_size <= 2000.0) & (nbed <= 3) => **(sale_price <= 0)**

(nbath <= 1) & (nstoreys <= 1) & (drive <= 0) & (rec_room <= 0) => **(sale_price <= 1)**

Decision Rules

Decision rules can be characterized by many attractiveness measures, like support, confidence, etc.

A decision rule that suggests assignment to set X is denoted by r_X. Condition part of rule r_X is denoted by $\Phi(r_X)$, while its decision part is denoted by $\Psi(r_X)$. Moreover, $\|\Phi(r_X)\|$ denotes the set of objects satisfying condition part of the rule.

Rule consistency measure is any function $\hat{\Theta}_X : R_X \to \mathbb{R}^+ \cup \{0\}$, where R_X is a set of rules suggesting assignment to X.
We consider the following cost-type rule consistency measure $\widehat{\epsilon}_X(r_X)$:

$$\widehat{\epsilon}_X(r_X) = \frac{\big|\|\Phi(r_X)\| \cap \neg X\big|}{|\neg X|}.$$

Decision Rules

Induced rules must satisfy the same constraints on consistency as objects from the lower approximation which serves as a base for rule induction. In particular, each rule $r_X \in R_X$ is required to satisfy threshold $\hat{\theta}_X$, equal to threshold θ_X used to calculate probabilistic lower approximation of X:

$$\widehat{\epsilon}_X(r_X) \leq \theta_X, \forall r_X \in R_X.$$

Decision Rules

Key concepts concerning rules induced from probabilistic lower approximations:

- A decision rule suggesting assignment to set X is discriminant if it covers only objects belonging to positive region $POS(X)$.
- Rule is free of redundant conditions if removing any of its elementary conditions causes that it is no more discriminant.
- Rule is minimal if there is no other rule with not less general conditions, not less specific decision, and not worse consistency, i.e., r_X is minimal if there does not exist other rule r_Y, $Y \subseteq U$, such that $\|\Phi(r_Y)\| \supseteq \|\Phi(r_X)\|$, $Y \subseteq X$, and $\hat{\epsilon}_X(r_Y) \succeq \hat{\epsilon}_X(r_X)$ (i.e., $\hat{\epsilon}_X(r_Y) \leq \hat{\epsilon}_X(r_X)$).
- Set of rules suggesting assignment to X is complete iff each object $y \in \underline{X}$ is covered by at least one rule $r_X \in R_X$.
- Rule $r_X \in R_X$ is non-redundant in R_X, if removing r_X causes that R_X ceases to be complete.

VCDomLEM Algorithm

VCDomLEM Algorithm

VC-DomLEM is a sequential covering rule induction algorithm that induces a minimal set of rules satisfying constraints on consistency.

VCDomLEM algorithm is composed of two parts:

- Algorithm 1 – the main routine,
- Algorithm 2 – $VC\text{-}SequentialCovering^{mix}$ subroutine.

VCDomLEM Algorithm

Algorithm 1: VC-DomLEM

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

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

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

VCDomLEM Algorithm

Each rule r_X belonging to set R_X is allowed to cover only objects from set $AO(X)$, calculated according to chosen option $s \in \{1, 2, 3\}$ (line 3).
We consider three reasonable options:

- (1): $AO(X) = POS(X)$,
- (2): $AO(X) = POS(X) \cup BND(X)$,
- (3): $AO(X) = U$.

Minimality check performed in line 6 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 can be non-minimal.

VCDomLEM Algorithm

Algorithm 2: VC-SequentialCovering^{mix}

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

Output: set R_X of rules suggesting assignment to X.

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

VCDomLEM Algorithm

The best elementary condition ec is chosen according to the following criteria considered lexicographically:

1. the best value of chosen rule consistency measure $\hat{\Theta}_X$ for rule $r_X \cup ec$,
2. the best value of $\left| \|\Phi(r_X \cup ec)\| \cap \underline{X} \right|$,

where $r_X \cup ec$ denotes a rule resulting from extension of rule r_X by new elementary condition ec.

If set R_X contains redundant rules, an iterative procedure eliminating redundancy is adopted (line 15). In each step of this procedure, the rule to be removed is chosen according to the following measures considered lexicographically:

1. the worst (i.e., the smallest) value of $\left| \|\Phi(r_X)\| \cap \underline{X} \right|$,
2. the worst value of $\hat{\Theta}_X(r_X)$,
3. the smallest index of r_X on the constructed list of rules.

Experimental Verification

Experimental Verification

The aim of the experiment was to evaluate the usefulness of VC-DomLEM algorithm in terms of its predictive accuracy (i.e., PCC and MAE).

VC-DomLEM algorithm, as implemented in the java Rough Set (jRS) library, was compared to other methods on 12 ordinal data sets.

In order to classify objects using induced rules, VC-DRSA classification scheme[a] was used.

[a] J. Błaszczyński, S. Greco, R. Słowiński, Multi-criteria classification – A new scheme for application of dominance-based decision rules, European Journal of Operational Research 181(3) (2007) 1030–1044

Experimental Verification

The other methods compared to VC-DomLEM were: two ordinal classifiers that preserve monotonicity constraints, namely: Ordinal Learning Model (OLM) and Ordinal Stochastic Dominance Learner (OSDL), and four non-ordinal classifiers: Naive Bayes, Support Vector Machine (SVM) with linear kernel, decision rule classifier RIPPER, and decision tree classifier C4.5.

Experimental Verification

Table: Characteristics of data sets

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

The predictive accuracy (PCC and MAE) was estimated by stratified 10-fold cross-validation, repeated several times.

The tables with results contain the value of measure and its standard deviation for each data set and each classifier.

For each data set we calculated a rank (given in brackets) of the result of a classifier when compared to the other classifiers.

Last row of each table shows the average rank obtained by a given classifier.

Moreover, for each data set, the best value of the predictive accuracy measure, and those values which are within standard deviation of the best value, are marked in bold.

Experimental Verification

Table: Mean absolute error (MAE)

Id	**VC-DomLEM**	Naive Bayes	SVM	RIPPER	C4.5	OLM	OSDL
1	**0.2331** (1) ±0.003297	0.2564 (3) ±0.005943	0.3217 (6) ±0.01244	0.2960 (4) ±0.01154	0.2424 (2) ±0.003297	0.324 (7) ±0.01835	0.3065 (5) ±0.001648
2	0.03720 (2) ±0.002023	0.03958 (3) ±0.0006744	**0.03243** (1) ±0.0006744	0.04483 (5) ±0.004721	0.05532 (6) ±0.00751	0.1764 (7) ±0.00552	0.04149 (4) ±0.001168
3	**0.03421** (1) ±0.0007275	0.1757 (6) ±0.002025	0.08668 (3) ±0.002025	0.2029 (7) ±0.01302	0.1168 (5) ±0.003108	0.09156 (4) ±0.005358	0.04141 (2) ±0.0009624
4	**0.08293** (1) ±0.01479	0.1707 (4) ±0.009832	0.4386 (7) ±0.01579	0.1611 (3) ±0.01372	0.1196 (2) ±0.01790	0.3461 (6) ±0.02744	0.3158 (5) ±0.01034
5	**0.04559** (1) ±0.001456	0.1146 (5) ±0.01371	0.1280 (6) ±0.001205	0.0489 (2) ±0.00352	0.0515 (3) ±0.005251	0.05528 (4) ±0.001736	0.1545 (7) ±0
6	**0.3406** (1.5) ±0.001878	0.4829 (5) ±0.002906	**0.3406** (1.5) ±0.001775	0.3991 (4) ±0.003195	0.3863 (3) ±0.005253	1.577 (6) ±0.03791	1.592 (7) ±0.007555
7	**0.1232** (1) ±0.01048	**0.1289** (2) ±0.01428	0.2129 (6) ±0.003961	0.1737 (5) ±0.02598	0.1653 (4) ±0.01048	0.2633 (7) ±0.02206	0.1541 (3) ±0.003961
8	1.307 (2) ±0.002055	1.325 (5) ±0.003771	1.318 (3) ±0.007257	1.681 (7) ±0.01558	1.326 (6) ±0.006018	1.321 (4) ±0.01027	**1.280** (1) ±0.00704
9	0.3702 (3) ±0.01352	**0.3456** (2) ±0.003864	0.4262 (5) ±0.01004	0.4296 (6) ±0.01608	0.3736 (4) ±0.01089	0.474 (7) ±0.01114	**0.3422** (1) ±0.005019
10	0.4813 (6) ±0.004028	0.475 (5) ±0.004320	0.4457 (4) ±0.003399	0.4277 (3) ±0.00838	0.426 (2) ±0.01476	0.615 (7) ±0.0099	**0.4033** (1) ±0.003091
11	0.454 (4) ±0.004320	0.475 (6) ±0.004320	0.4503 (2) ±0.002867	0.452 (3) ±0.006481	0.4603 (5) ±0.004497	0.5707 (7) ±0.007717	**0.433** (1) ±0.002160
12	**0.5024** (1) ±0.006226	0.5488 (3) ±0.005662	0.5891 (5) ±0.02101	0.6825 (7) ±0.03332	0.652 (6) ±0.03721	0.5757 (4) ±0.006044	0.5153 (2) ±0.006044
	2.04	4.08	4.13	4.67	4.00	5.83	3.25

Table: Percentage of correctly classified objects (PCC)

Id	VC-DomLEM	Naive Bayes	SVM	RIPPER	C4.5	OLM	OSDL
1	**76.69** (1) $\pm$0.3297	74.36 (3) $\pm$0.5943	67.83 (6) $\pm$1.244	70.4 (4) $\pm$1.154	75.76 (2) $\pm$0.3297	67.6 (7) $\pm$1.835	69.35 (5) $\pm$0.1648
2	96.28 (2) $\pm$0.2023	96.04 (3) $\pm$0.06744	**96.76** (1) $\pm$0.06744	95.52 (5) $\pm$0.4721	94.47 (6) $\pm$0.751	82.36 (7) $\pm$0.552	95.85 (4) $\pm$0.1168
3	**97.15** (1) $\pm$0.063	84.72 (6) $\pm$0.1667	92.18 (3) $\pm$0.2025	84.41 (7) $\pm$1.309	89.84 (5) $\pm$0.1819	91.72 (4) $\pm$0.4425	96.53 (2) $\pm$0.063
4	**91.7** (1) $\pm$1.479	83.41 (4) $\pm$0.9832	56.62 (7) $\pm$1.579	84.69 (3) $\pm$1.409	88.52 (2) $\pm$1.409	68.58 (6) $\pm$2.772	72.41 (5) $\pm$1.479
5	**95.44** (1) $\pm$0.1456	88.54 (5) $\pm$1.371	87.2 (6) $\pm$0.1205	95.11 (2) $\pm$0.352	94.85 (3) $\pm$0.5251	94.47 (4) $\pm$0.1736	84.55 (7) $\pm$0
6	**67.55** (1) $\pm$0.4642	56.22 (5) $\pm$0.2328	**67.1** (2) $\pm$0.2217	63.55 (4) $\pm$0.5635	64.33 (3) $\pm$0.5844	27.43 (6) $\pm$0.7179	22.04 (7) $\pm$0.128
7	**87.68** (1) $\pm$1.048	**87.11** (2) $\pm$1.428	78.71 (6) $\pm$0.3961	82.63 (5) $\pm$2.598	83.47 (4) $\pm$1.048	73.67 (7) $\pm$2.206	84.6 (3) $\pm$0.3961
8	26.9 (2) $\pm$0.3742	25.03 (3) $\pm$0.2494	24.27 (5) $\pm$0.2494	20 (7) $\pm$0.4243	**27.83** (1) $\pm$0.4028	23.97 (6) $\pm$0.4643	24.7 (4) $\pm$0.8165
9	66.73 (3) $\pm$1.256	67.49 (2) $\pm$0.3483	62.7 (5) $\pm$0.6693	61.61 (6) $\pm$1.555	66.33 (4) $\pm$0.6966	55.46 (7) $\pm$0.7545	**68.3** (1) $\pm$0.3483
10	55.63 (6) $\pm$0.3771	56.17 (5) $\pm$0.3399	58.87 (4) $\pm$0.3091	60.83 (2) $\pm$0.6128	60.73 (3) $\pm$1.271	45.43 (7) $\pm$0.8179	**63.03** (1) $\pm$0.2625
11	56.43 (6) $\pm$0.4643	56.57 (5) $\pm$0.4784	**58.23** (2) $\pm$0.2055	57.63 (3) $\pm$0.66	57.1 (4) $\pm$0.4320	47.83 (7) $\pm$0.411	**58.6** (1) $\pm$0.4243
12	54.58 (2) $\pm$0.7913	53.6 (3) $\pm$0.2284	51.83 (4) $\pm$1.813	44.08 (7) $\pm$0.8236	47.99 (6) $\pm$2.888	49.15 (5) $\pm$0.7527	**55.37** (1) $\pm$0.3763
	2.25	3.83	4.25	4.58	3.58	6.08	3.42

Experimental Verification

From the results of the experiment, VC-DomLEM appears to be better than the other compared classifiers – it has the best value of the average rank of both predictive accuracy measures.

More detailed analysis of the results is presented in the literature (see references).

Conclusions

Conclusions

- DRSA is a flexible modeling method that allows to include domain knowledge and can handle possible inconsistencies in data by calculating lower approximations of sets.
- DRSA allows to work with heterogeneous attributes – nominal, ordinal and cardinal (no need of discretization).
- Rule model has many advantages, e.g., comprehensibility, lack of aggregation operators, predictive power, resistance to irrelevant attributes.
- VC-DomLEM is a sequential covering algorithm inducing minimal decision rules in (VC-)DRSA.
- Rule models that preserve monotonicity constraints are more transparent than the other models.

Software

- jRS – java Rough Sets library (jRS): `www.cs.put.poznan.pl/mszelag/Software/software.html`,
- jMAF – java Multi-criteria and Multi-attribute Analysis Framework: `www.cs.put.poznan.pl/jblaszczynski/Site/jRS.html`,
- ruleLearn – library of methods supporting learning and application of decision rules: `https://github.com/ruleLearn/rulelearn`,
- RuleVisualization – web application for visualization and exploration of monotonic decision rules: `http://www.cs.put.poznan.pl/mszelag/Software/RuleVisualization/RuleVisualization.html`

References

- S. Greco, B. Matarazzo, R. Słowiński, Rough Sets Theory for Multicriteria Decision Analysis. European Journal of Operational Research, 129(1), 2001, pp. 1–47.
- J. Błaszczyński, S. Greco, R. Słowiński, M. Szeląg, Monotonic Variable Consistency Rough Set Approaches. International Journal of Approximate Reasoning, 50(7), 2009, pp. 979–999.
- J. Błaszczyński, R. Słowiński, M. Szeląg, Sequential Covering Rule Induction Algorithm for Variable Consistency Rough Set Approaches. Information Sciences, 181, 2011, pp. 987–1002, ⇒ manuscript.
- M. Szeląg, J. Błaszczyński, R. Słowiński, Rough Set Analysis of Classification Data with Missing Values. [In]: L. Polkowski et al. (Eds.): Rough Sets, International Joint Conference, IJCRS 2017, Olsztyn, Poland, July 3–7, 2017, Proceedings, Part I. LNAI, vol. 10313, Springer, 2017, pp. 552–565 ⇒ manuscript.