

Context-Dependent Deductive and Inductive Reasoning Based on Good Diagnostic Tests

Xenia Naidenova¹ and Vladimir Parkhomenko²

¹ Military Medical Academy, Saint-Petersburg, Russia
ksennaidd@gmail.com

² Peter the Great St. Petersburg Polytechnic University, St. Petersburg, Russia
parhomenko.v@gmail.com

Abstract. A sketch of classification reasoning is given in the paper. The key ideas of the reasoning are ideas of classification and its good approximations based on good diagnostic tests. Such good tests, which are maximally redundant (GMRTs), i.e. their subsets of attributes are closed, are considered. Classification reasoning embraces two interrelated processes: inductive inferring implicative assertions based on GMRTs and using these assertions in deductive steps of reasoning. A peculiarity of our inductive model of classification reasoning is the control and transformation of subcontexts during inductive phase of reasoning by the use of deductive reasoning rules, for example the rules prohibiting some pairs of attributive values.

Keywords: good diagnostic test, classification reasoning, essential attributes, essential objects, implications, subcontexts, deduction, induction, closed sets

1 Introduction

Classification reasoning is a part of commonsense (plausible) reasoning based on two kinds of logical rules extracted from observable datasets (functional and implicative dependencies). The principle concepts of this reasoning are the concepts of classification and its good approximations. We assume that objects are described by a set U of symbolic or numeric attributes. To give a target classification of objects, we use an additional attribute KL not belonging to U . A target attribute partitions a given set of objects into disjoint classes the number of which is equal to the number of values of this attribute. In Tab. 1, we have two classes: KL_+ (positive objects) and KL_- (negative objects).

We concentrate on the case of object classification into two classes and task of concept formation [2,5]. The goal of this task is to describe/classify new objects according to description/classification of existing objects. Inferring good diagnostic tests (GDTs) is the formation of the best descriptions of a given object class KL_+ against the objects not belonging to this class (KL_-).

Let $M = \{\cup \text{dom}(\text{attr}), \text{attr} \in U\}$, where $\text{dom}(\text{attr})$ is the set of all values of attr. Let $X \subseteq M$ and G be the set of indices of objects considered (objects

Table 1. Motivating Example of classification

No	Height	Color of Hair	Color of Eyes	KL
1	Low	Blond	Blue	KL ₊
2	Low	Brown	Blue	KL ₋
3	Tall	Brown	Hazel	KL ₋
4	Tall	Blond	Hazel	KL ₋
5	Tall	Brown	Blue	KL ₋
6	Low	Blond	Hazel	KL ₋
7	Tall	Red	Blue	KL ₊
8	Tall	Blond	Blue	KL ₊

for short), $G = G_+ \cup G_-$, where G_+ and G_- the sets of positive and negative objects, respectively. Denote by $d(g)$ the description of object $g \in G$. Let $P(X) = \{g \mid g \in G, X \subseteq d(g)\}$. We call $P(X)$ the interpretation of X in the power set 2^G . If $P(X)$ contains only positive objects and the number of these objects more than 2, then we call X a description of some positive objects and $(P(X), X)$ a test for positive objects.

Let us define a good test or good description of objects.

Definition 1. *A set $X \subseteq M$ of attribute values is a good description of positive (negative) objects if and only if it is the description of these objects and no such subset $Y \subseteq M$ exists, that $P(X) \subset P(Y) \subseteq G_+$.*

It can be seen [15,11,1] that this problem is reduced to searching for causal dependencies in the form $X \rightarrow v, X \subseteq M, v$ is value of attribute KL for all positive (negative) objects.

Sec.2 describes an approach to classification reasoning based on implications. Sec.3 describes the definitions of Good Test Analysis. Sec.4 presents the decomposition of inferring good tests into two kinds of subtasks. An approach to selecting and ordering subcontexts is given in Subsec.5.1. Some situations of reducing the classification context are considered in Subsec.5.2. A set of examples illustrates all the considerations.

2 Classification Reasoning

Classification reasoning based on GDTs embraces two interrelated processes: inductive inferring implicative assertions and using them for pattern recognition problems. We need the following three types of logical rules in order to realize classification reasoning (deductive and inductive):

- INSTANCES or relationships between objects or facts really observed. Instance can be considered as a logical rule with the least degree of generalization. On the other hand, instances can serve as a source of training set of positive and negative examples for inductive inference of generalized rules.

- RULES OF THE FIRST TYPE or logical assertions. These rules describe regular relationships between objects and their properties, between properties of different objects, between objects and their classes. The rules of the first type can be given explicitly by an expert or derived automatically from instances with the help of some learning process. These rules are represented, in the paper, in the form of implications.
- RULES OF THE SECOND TYPE or commonsense reasoning rules with the help of which rules of the first type are used, updated, and inferred from data (instances). The rules of the second type embrace both inductive and deductive reasoning rules.

The rules of the first type can be represented with the use of only one class of logical statements based on implicative dependencies between names. Names are used for designating concepts, things, events, situations, or any evidences. They can be considered as attributes and attributes' values in the formal representations of logical rules.

1. Implication: $a, b, c \rightarrow d$. This rule means that if the values standing on the left side of rule are simultaneously *true*, then the value on the right side of rule is always *true*.
2. Interdiction or forbidden rule (a special case of implication): $a, b, c \rightarrow false$ (*never*). This rule interdicts a combination of values enumerated on the left side of rule. The rule of interdiction can be transformed into several implications such as $a, b \rightarrow not\ c$; $a, c \rightarrow not\ b$; $b, c \rightarrow not\ a$.
3. Compatibility: $a, b, c \rightarrow VA$ where VA is the frequency of occurrence of rule. The compatibility is equivalent to the following implications: $a, b \rightarrow c, VA$; $a, c \rightarrow b, VA$; $b, c \rightarrow a, VA$.
Generally, the compatibility rule represents the most common combination of values, which is characterized by an insignificant number of exceptions (contrary examples) from regularity.
4. Diagnostic rule: $x, d \rightarrow a$; $x, b \rightarrow not\ a$; $d, b \rightarrow false$. For example, d and b can be two values of the same attribute. This rule works when the truth of ' x ' has been proven and it is necessary to determine whether ' a ' is *true* or not. If ' $x&d$ ' is *true*, then ' a ' is *true*, if ' $x&b$ ' is *true*, then ' a ' is *false*.
5. Rule of alternatives: $a\ or\ b \rightarrow true$ (*always*); $a, b \rightarrow false$. This rule says that a and b cannot be simultaneously *true*, either a or b can be *true* but not both. This rule is a variant of interdiction.

Deductive steps of classification reasoning consist of inferring consequences from some observed facts with the use of implications (i.e. knowledge). For this goal, deductive rules of reasoning are applied the main forms of which are modus ponens, modus tollens, modus ponendo tollens, and modus tollendo ponens. Let x be a set of *true* values of some attributes observed simultaneously. Deductive reasoning rules of the second type are:

1. Using implication: Let r be an implication, $left(r)$ be the left part of r and $right(r)$ be the right part of r . If $left(r) \subseteq x$, then x can be extended by

- $\text{right}(r) : x \leftarrow x \cup \text{right}(r)$. Using implication is based on modus ponens: if A , then B ; A ; hence B .
2. Using interdiction: Let r be an implication $y \rightarrow \text{not } k$. If $\text{left}(r) \subseteq x$, then k is the forbidden value for all extensions of x . Using interdiction is based on modus ponendo tollens: either A or B (A, B — alternatives); A ; hence not B ; either A or B ; B ; hence not A .
 3. Using compatibility: Let $r = 'a, b, c \rightarrow k, VA'$. If $\text{left}(r) \subseteq x$, then k can be used to extend x along with the calculated value VA for this extension (a special consideration is required for calculating VA).
 4. Using diagnostic rules: Let r be a diagnostic rule such as ' $x, d \rightarrow a; x, b \rightarrow \text{not } a'$ ', where ' x ' is *true*, and ' a ', ' $\text{not } a$ ' are hypotheses or possible values of some attribute. Using the diagnostic rule is based on modus ponens and modus ponendo tollens. There are several ways for refuting one of the hypotheses:
 - To infer either d or b by using existing knowledge;
 - To involve new known facts (extended context) and/or propositions for inferring (with the use of inductive reasoning rules of the second type) new implications for distinguishing between the hypotheses ' a ' and ' $\text{not } a$ '; to apply the newly obtained rules;
 - To get the direct answer on whether d or b is *true* by involving an external source of knowledge.
 5. Using rule of alternatives: Let ' a ' and ' b ' be two alternatives (mutually exclusive) hypotheses about the value of some attribute, and the truth of one of hypotheses has been established, then the second hypothesis is rejected. Using the rule is based on modus tollendo ponens: either A or B (A, B — alternatives); not A ; hence B ; either A or B ; not B ; hence A .
We can call the rules listed above the rules of “forward inference”. However, we have in natural classification reasoning so called “backward inference”.
 6. Generating hypothesis or abduction rule. Let r be an implication $y \rightarrow k$. We have “if k is *true*, then y may be *true*”.
 7. Using modus tollens. Let r be an implication $y \rightarrow k$. If ' $\text{not } k$ ' is inferred, then ' $\text{not } y$ ' is also inferred.

When applied, the above given rules generate the reasoning, which is not demonstrative. The purpose of it is to infer all possible hypotheses of the value of some objective attribute. It is essential that hypotheses do not contradict with knowledge (first-type rules) and the observable real situation, where the reasoning takes place (contextual knowledge). Inference of hypotheses is reduced to constructing all intrinsically consistent extensions of a set of values in which the number of involved attributes is maximum possible (there are no prohibited pairs of values in such extensions). All hypotheses have different admissibility determined by the quantity and “quality” of compatibility rules involved in reasoning.

A very simple structure of knowledge base (KB) and an example of applying the rules of second type for a pattern recognition problem can be found in [19]. The process of deductive reasoning can require inferring new rules of the first type from data when i) the result of reasoning contains several hypotheses and

it is impossible to choose one and only one of them (uncertainty), and ii) it is impossible to infer any hypothesis.

Inductive steps of classification reasoning consist in using already known facts, observations and experimental results for inferring implications and their correcting. The main forms of induction have been formulated by J. Stuart Mill [12]. The formalization of these forms of induction has been offered by V.K. Finn [6]. For inferring GDTs, the Combined Similarity and Difference Method of the Mill are used and realized by means of inductive inference rules of the second type proposed in [13].

Deductive rules of the second type described above also work for inductive inferring implications since implications obtained are involved immediately to reduce the search space. In the paper, we shall illustrate using the forbidden and diagnostic rules of the second type for GDTs inferring.

3 Key Notions of Good Test Analysis

Assume that $G = \overline{1, N}$ be the set of objects and $M = \{m_1, m_2, \dots, m_j, \dots, m_m\}$ be the set of values. The object descriptions are represented by rows of a table, columns of which are associated with the attributes taking their values in M (see, please, Tab.1).

Denote by $\{B_g | B_g \subseteq M, g \in G\}$ the description of an object. The Galois connection [21] between the ordered sets $(2^G, \subseteq)$ and $(2^M, \subseteq)$, i.e. $2^G \rightarrow 2^M$ and $2^M \rightarrow 2^G$, is defined by the following mappings called derivation operators (see although different notations in [16,9]): for $A \subseteq G$ and $B \subseteq M$, $A' = \text{val}(A) = \{\text{intersection of all } B_g | B_g \subseteq M, g \in A\}$ and $B' = \text{obj}(B) = \{g | g \in G, B \subseteq B_g\}$. We consider $\text{obj}(B) = \{\text{intersection of all } \text{obj}(m) | \text{obj}(m) \subseteq G, m \in B\}$. In the Formal Concept Analysis, for $g \in G$ and $m \in M$, $\{g\}'$ or simply g' is called object intent, and $\{m\}'$ or simply m' is called attribute extent [7].

There are two closure operators [16]: $\text{generalization_of}(B) = B'' = \text{val}(\text{obj}(B))$ and $\text{generalization_of}(A) = A'' = \text{obj}(\text{val}(A))$. A set A is closed if $A = \text{obj}(\text{val}(A))$ and a set B is closed if $B = \text{val}(\text{obj}(B))$. If $A' = B$ & $B' = A$, then a pair (A, B) is called a formal concept, subsets A and B of which are called concept extent and intent, respectively.

A triple (G, M, I) , where I is a binary relation $G \times M$ is called a formal context \mathbb{K} . Let $\mathbb{K}_\epsilon := (G_\epsilon, M, I_\epsilon)$ and $I_\epsilon := I \cap (G_\epsilon \times M)$, where $\epsilon \in \{+, -\}$ (one can add the value τ if there is an undefined object) [8]. $\mathbb{K}_\pm := (G_+ \cup G_-, M \cup \{\text{KL}\}, I_+ \cup I_- \cup (G_\pm \times \{\text{KL}\}))$.

Definition 2. A Diagnostic (classification) Test (DT) for G_+ is a pair (A, B) such that $B \subseteq M$, $A = \text{obj}(B) \neq \emptyset$, $A \subseteq G_+$, and $B \not\subseteq \text{val}(g), \forall g \in G_-$. Equivalently, $\text{obj}(B) \cap G_- = \emptyset$.

Definition 3. A DT (A, B) for G_+ is maximally redundant if $\text{obj}(B \cup m) \subset A$ for all $m \notin B$ and $m \in M$.

Definition 4. A DT (A, B) for G_+ is irredundant if any narrowing $B_* = B \setminus m, m \in B$ implies that $(\text{obj}(B_*), B_*)$ is not a test for G_+ .

Definition 5. A DT (A, B) for G_+ is good if and only if any extension $A_* = A \cup i, i \notin A, i \in G_+$ implies that $(A_*, \text{val}(A_*))$ is not a test for G_+ .

If a good test (A, B) for G_+ is maximally redundant (GMRT), then any extension $B_* = B \cup m, m \notin B, m \in M$ implies that $(\text{obj}(B_*), B_*)$ is not a good test for G_+ . The relation GMRTs to the Formal Concept Analysis is considered in [17].

Any object description $d(g), g \in G$ in a given classification context is a maximally redundant set of values because $\forall m \notin d(g), m \in M, \text{obj}(d(g) \cup m)$ is equal to \emptyset .

In Tab.1, $((1, 8), \text{Blond Blue})$ is a GMRT for KL_+ , $((4, 6), \text{Blond Hazel})$ is a maximally redundant test for KL_- but not a good one, and $((3, 4, 6), \text{Hazel})$ is a good maximally redundant and, simultaneously, good irredundant test for KL_- .

4 Decomposition of Inferring GMRTs into Subtasks

There are two kinds of subtasks for inferring GMRTs:

1. given a set of values B , where $B \subseteq M, \text{obj}(B) \neq \emptyset, B \not\subseteq d(g), \forall g \in G_-$, find all GMRTs $(\text{obj}(B_*), B_*)$ such that $B_* \subset B$;
2. given a non-empty set of values $X \subseteq M$ such that $(\text{obj}(X), X)$ is not a test for positive objects, find all GMRTs $(\text{obj}(Y), Y)$ such that $X \subset Y$.

For solving these subtasks we need to form subcontexts of a given classification context \mathbb{K}_\pm . In the first case, we have to include in subcontext all positive objects, the descriptions of which intersect B . The subtask is useful to find all GMRTs intents of which are contained in the description $d(g)$ of an object g . This subtask is considered in [5] for fast incremental concept formation. The following notions of object and value projections are developed to form subcontexts [18].

Definition 6. The projection $\text{proj}(d(g)), g \in G_+$ on the set D_+ , where D_+ is the set of descriptions of all positive objects, is denoted by $\{z \mid z = d(g) \cap d(g_*) \neq \emptyset, d(g_*) \in D_+ \text{ \& } (\text{obj}(z), z) \text{ is a test for } G_+\}$. Note that $d(g) \in \text{proj}(d(g))$.

Definition 7. The value projection on a given set D_+ $\text{proj}(B)$ is $\{d(g) \mid B \subseteq d(g), d(g) \in D_+\}$ or, equivalently, $\text{proj}(B) = \{d(g) \mid g \in (\text{obj}(B) \cap G_+)\}$.

The projections define the methods to construct two kinds of subcontexts of \mathbb{K}_\pm . The following theorem gives the foundation of reducing subcontexts [14].

Theorem 1. Let $X \subseteq M, (\text{obj}(X), X)$ be a maximally redundant test for positive objects and $\text{obj}(m) \subseteq \text{obj}(X), m \in M$. Then m can not belong to any GMRT for positive objects different from $(\text{obj}(X), X)$.

Let $\text{splus}(m)$ be $\text{obj}(m) \cap G_+$ ($\text{obj}(m) \cap G_-$) and $\text{SPLUS} = \{\text{splus}(m), m \in M\}$. Consider an example of reducing subcontext in Tab. 1.

For values Hazel, Brown, Tall, Blue, Blond, and Low, $\text{SPLUS} = \{\text{obj}(m) \cap G_-\} = \{\{3, 4, 6\}, \{2, 3, 5\}, \{3, 4, 5\}, \{2, 5\}, \{4, 6\}, \{2, 6\}\}$.

We have $\text{val}(\text{obj}(\text{Hazel})) = \text{Hazel}$, hence $((3,4,6), \text{Hazel})$ is a test for G_- . Then value Blond can be deleted from consideration, because $\text{splus}(\text{Blond}) \subset \text{splus}(\text{Hazel})$.

It is essential that the projection is a subset of object descriptions defined on a certain restricted subset $t_* \subseteq M$ of values. Let s_* be the subset of objects, descriptions of which produce the projection. In the projection, $\text{splus}(m) = \text{obj}(m) \cap s_*$, $m \in t_*$.

Let STGOOD be the partially ordered set of elements s such that $(s, \text{val}(s))$ is a good test for D_+ .

Forming the Set STGOOD. The important part of our basic recursive algorithm is how to form the set STGOOD. Let $L = (2^S, \leq)$ be the set lattice [22]. The ordering determined in the set lattice coincides with the set-theoretical inclusion. Subset s_1 is absorbed by subset s_2 , i.e. $s_1 \leq s_2$, if and only if $s_1 \subseteq s_2$. Under formation of STGOOD, subset $s \subseteq G_+$ is stored in STGOOD if and only if it is not absorbed by any element of this set. It is necessary also to delete from STGOOD all the elements that are absorbed by s if s is stored in STGOOD. Thus, when the algorithm is over, the set STGOOD contains all the collections of objects that are the extents of GMRTs and only such collections. The set TGOOD of all the GMRTs is obtained as follows: $\text{TGOOD} = \{\text{tg} \mid \text{tg} = (s, \text{val}(s)), s \in \text{STGOOD}\}$.

The basic recursive procedure for solving any kind of subtasks has been described in [20].

5 Selecting and Ordering Subcontexts and Inferring GMRTs

Algorithms of GMRTs inferring are constructed by the rules of selecting and ordering subcontexts of the main classification context. Before entering into the details, we give some new definitions.

Definition 8. *Let t be a set of values such that $(\text{obj}(t), t)$ is a test for G_+ . The value $m \in M, m \in t$ is essential in t if $(\text{obj}(t \setminus m), (t \setminus m))$ is not a test for a given set of objects.*

Generally, we are interested in finding the maximal subset $\text{sbmax}(t) \subset t$ such that $(\text{obj}(t), t)$ is a test but $(\text{obj}(\text{sbmax}(t)), \text{sbmax}(t))$ is not a test for a given set of positive objects. Then $\text{sbmin}(t) = t \setminus \text{sbmax}(t)$ is a minimal set of essential values in t .

Definition 9. *Let $s \subseteq G_+$, assume also that $(s, \text{val}(s))$ is not a test. The object $g, g \in s$ is said to be an essential in s if $(s \setminus g, \text{val}(s \setminus g))$ proves to be a test for a given set of positive objects.*

Generally, we are interested in finding the maximal subset $\text{sbmax}(s) \subset s$ such that $(s, \text{val}(s))$ is not a test but $(\text{sbmax}(s), \text{val}(\text{sbmax}(s)))$ is a test for a given set of positive objects. Then $\text{sbmin}(s) = s \setminus \text{sbmax}(s)$ is a minimal set of essential objects in s .

Finding quasi-maximal (minimal) subsets of objects and values is the key procedure behind searching for initial content of STGOOD and determining the number of subtasks to be solved.

5.1 Using the Diagnostic Rule of the Second Type

An Approach for Searching for Initial Content of STGOOD. In the beginning of inferring GMRTs, the set STGOOD is empty. Next we describe the procedure to obtain an initial content of it. This procedure extracts a quasi-maximal subset $s_* \subseteq G_+$, which is the extent of a test for G_+ (maybe not good).

We begin with the first object g_1 of s_* , then we take the next object g_2 of s_* and evaluate the function $\text{to_be_test}(\{g_1, g_2\}, \text{val}(\{g_1, g_2\}))$. If the value of the function is *true*, then we take the next object g_3 of s_* and evaluate the function $\text{to_be_test}(\{g_1, g_2, g_3\}, \text{val}(\{g_1, g_2, g_3\}))$. If the value of the function $\text{to_be_test}(\{g_1, g_2\}, \text{val}(\{g_1, g_2\}))$ is *false*, then the object g_2 of s_* is skipped and the function $\text{to_be_test}(\{g_1, g_3\}, \text{val}(\{g_1, g_3\}))$ is evaluated. We continue this process until we achieve the last object of s_* . This procedure can be considered as a realization of the diagnostic rule of the second type.

To illustrate this procedure, we use the sets D_+ and D_- represented in Tab.2 and 3 (our illustrative example). In these tables $M = \{m_1, \dots, m_{26}\}$. The set SPLUS_0 for positive class of objects is in Tab.4. The initial content of STGOOD_0 is $\{(2,10), (3, 10), (3, 8), (4, 12), (1, 4, 7), (1, 5,12), (2, 7, 8), (3, 7, 12), (1, 2, 12, 14), (2, 3, 4, 7), (4, 6, 8, 11)\}$.

In what follows, we denote subsets of values $\{m_8, m_9\}, \{m_{14}, m_{15}\}$ by m_a and m_b , respectively. Applying operation $\text{generalization_of}(s) = s'' = \text{obj}(\text{val}(s))$ to $\forall s \in \text{STGOOD}$, we obtain $\text{STGOOD}_1 = \{(2,10), (3, 10), (3, 8), (4, 7, 12), (1, 4, 7), (1, 5,12), (2, 7, 8), (3, 7, 12), (1, 2, 12, 14), (2, 3, 4, 7), (4, 6, 8, 11)\}$.

By Th.1, we can delete value m_{12} from consideration, see $\text{splus}(m_{12})$ in Tab.4. The initial content of STGOOD allows to decrease the number of using the procedure $\text{to_be_test}()$ and the number of putting extents of tests into STGOOD.

The number of subtasks to be solved. This number is determined by the number of essential values in the set M . The quasi-minimal subset of essential values in M can be found by a procedure analogous to the procedure applicable to search for the initial content of STGOOD.

As a result of this procedure, we have quasi-maximal subset $\text{sbmax}(M)$ of M , such that $(\text{obj}(\text{sbmax}(M)), \text{sbmax}(M))$ is not a test for positive examples. Then subset $\text{LEV} = M \setminus \text{sbmax}(M)$ is quasi-minimal subset of essential values in M . We have the following *LEV*: $\{m_{16}, m_{18}, m_{19}, m_{20}, m_{21}, m_{22}, m_{23}, m_{24}, m_{26}\}$. Note, that searching for the number of subtasks to be solved is also based on using the diagnostic rule of the second type.

Table 2. The set D_+ of positive object descriptions

G	D_+
1	$m_1 m_2 m_5 m_6 m_{21} m_{23} m_{24} m_{26}$
2	$m_4 m_7 m_8 m_9 m_{12} m_{14} m_{15} m_{22} m_{23} m_{24} m_{26}$
3	$m_3 m_4 m_7 m_{12} m_{13} m_{14} m_{15} m_{18} m_{19} m_{24} m_{26}$
4	$m_1 m_4 m_5 m_6 m_7 m_{12} m_{14} m_{15} m_{16} m_{20} m_{21} m_{24} m_{26}$
5	$m_2 m_6 m_{23} m_{24}$
6	$m_7 m_{20} m_{21} m_{26}$
7	$m_3 m_4 m_5 m_6 m_{12} m_{14} m_{15} m_{20} m_{22} m_{24} m_{26}$
8	$m_3 m_6 m_7 m_8 m_9 m_{13} m_{14} m_{15} m_{19} m_{20} m_{21} m_{22}$
9	$m_{16} m_{18} m_{19} m_{20} m_{21} m_{22} m_{26}$
10	$m_2 m_3 m_4 m_5 m_6 m_8 m_9 m_{13} m_{18} m_{20} m_{21} m_{26}$
11	$m_1 m_2 m_3 m_7 m_{19} m_{20} m_{21} m_{22} m_{26}$
12	$m_2 m_3 m_{16} m_{20} m_{21} m_{23} m_{24} m_{26}$
13	$m_1 m_4 m_{18} m_{19} m_{23} m_{26}$
14	$m_{23} m_{24} m_{26}$

Table 3. The set D_- of negative object descriptions

G	D_-	G	D_-
15	$m_3 m_8 m_{16} m_{23} m_{24}$	32	$m_1 m_2 m_3 m_7 m_9 m_{13} m_{18}$
16	$m_7 m_8 m_9 m_{16} m_{18}$	33	$m_1 m_5 m_6 m_8 m_9 m_{19} m_{20} m_{22}$
17	$m_1 m_{21} m_{22} m_{24} m_{26}$	34	$m_2 m_8 m_9 m_{18} m_{20} m_{21} m_{22} m_{23} m_{26}$
18	$m_1 m_7 m_8 m_9 m_{13} m_{16}$	35	$m_1 m_2 m_4 m_5 m_6 m_7 m_9 m_{13} m_{16}$
19	$m_2 m_6 m_7 m_9 m_{21} m_{23}$	36	$m_1 m_2 m_6 m_7 m_8 m_{13} m_{16} m_{18}$
20	$m_{19} m_{20} m_{21} m_{22} m_{24}$	37	$m_1 m_2 m_3 m_4 m_5 m_6 m_7 m_{12} m_{14} m_{15} m_{16}$
21	$m_1 m_{20} m_{21} m_{22} m_{23} m_{24}$	38	$m_1 m_2 m_3 m_4 m_5 m_6 m_9 m_{12} m_{13} m_{16}$
22	$m_1 m_3 m_6 m_7 m_9 m_{16}$	39	$m_1 m_2 m_3 m_4 m_5 m_6 m_{14} m_{15} m_{19} m_{20} m_{23} m_{26}$
23	$m_2 m_6 m_8 m_9 m_{14} m_{15} m_{16}$	40	$m_2 m_3 m_4 m_5 m_6 m_7 m_{12} m_{13} m_{14} m_{15} m_{16}$
24	$m_1 m_4 m_5 m_6 m_7 m_8 m_{16}$	41	$m_2 m_3 m_4 m_5 m_6 m_7 m_9 m_{12} m_{13} m_{14} m_{15} m_{19}$
25	$m_7 m_{13} m_{19} m_{20} m_{22} m_{26}$	42	$m_1 m_2 m_3 m_4 m_5 m_6 m_{12} m_{16} m_{18} m_{19} m_{20} m_{21} m_{26}$
26	$m_1 m_2 m_3 m_5 m_6 m_7 m_{16}$	43	$m_4 m_5 m_6 m_7 m_8 m_9 m_{12} m_{13} m_{14} m_{15} m_{16}$
27	$m_1 m_2 m_3 m_5 m_6 m_{13} m_{18}$	44	$m_3 m_4 m_5 m_6 m_8 m_9 m_{12} m_{13} m_{14} m_{15} m_{18} m_{19}$
28	$m_1 m_3 m_7 m_{13} m_{19} m_{21}$	45	$m_1 m_2 m_3 m_4 m_5 m_6 m_7 m_8 m_9 m_{12} m_{13} m_{14} m_{15}$
29	$m_1 m_4 m_5 m_6 m_7 m_8 m_{13} m_{16}$	46	$m_1 m_3 m_4 m_5 m_6 m_7 m_{12} m_{13} m_{14} m_{15} m_{16} m_{23} m_{24}$
30	$m_1 m_2 m_3 m_6 m_{12} m_{14} m_{15} m_{16}$	47	$m_1 m_2 m_3 m_4 m_5 m_6 m_8 m_9 m_{12} m_{14} m_{16} m_{18} m_{22}$
31	$m_1 m_2 m_5 m_6 m_{14} m_{15} m_{16} m_{26}$	48	$m_2 m_8 m_9 m_{12} m_{14} m_{15} m_{16}$

Table 4. The set SPLUS₀

splus(m), $m \in M$	splus(m), $m \in M$
splus(m_a) \rightarrow {2, 8, 10}	splus(m_{22}) \rightarrow {2, 7, 8, 9, 11}
splus(m_{13}) \rightarrow {3, 8, 10}	splus(m_{23}) \rightarrow {1, 2, 5, 12, 13, 14}
splus(m_{16}) \rightarrow {4, 9, 12}	splus(m_3) \rightarrow {3, 7, 8, 10, 11, 12}
splus(m_1) \rightarrow {1, 4, 11, 13}	splus(m_4) \rightarrow {2, 3, 4, 7, 10, 13}
splus(m_5) \rightarrow {1, 4, 7, 10}	splus(m_6) \rightarrow {1, 4, 5, 7, 8, 10}
splus(m_{12}) \rightarrow {2, 3, 4, 7}	splus(m_7) \rightarrow {2, 3, 4, 6, 8, 11}
splus(m_{18}) \rightarrow {3, 9, 10, 13}	splus(m_{24}) \rightarrow {1, 2, 3, 4, 5, 7, 12, 14}
splus(m_2) \rightarrow {1, 5, 10, 11, 12}	splus(m_{20}) \rightarrow {4, 6, 7, 8, 9, 10, 11, 12}
splus(m_b) \rightarrow {2, 3, 4, 7, 8}	splus(m_{21}) \rightarrow {1, 4, 6, 8, 9, 10, 11, 12}
splus(m_{19}) \rightarrow {3, 8, 9, 11, 13}	splus(m_{26}) \rightarrow {1, 2, 3, 4, 6, 7, 9, 10, 11, 12, 13, 14}

Proposition 1. *Each essential value is included at least in one positive object description.*

Proof. Assume that for an object description t_g , $g \in G_+$, we have $t_g \cap LEV = \emptyset$. Then $t_g \subseteq M \setminus LEV$. But $M \setminus LEV$ is included at least in one of the negative object descriptions and, consequently, t_g also possesses this property. But it contradicts to the fact that t_g is the description of a positive object. \square

Proposition 2. *Assume that $X \subseteq M$. If $X \cap LEV = \emptyset$, then $\text{to_be_test}(\text{obj}(X), X) = \text{false}$.*

Proposition 3. *Let \mathbb{K}_\pm be a given classification context. For finding all GRMTs containing in \mathbb{K}_\pm , it is sufficient to solve this problem only for subcontexts associated with essential values in M (the set LEV).*

Proposition 2 is the consequence of Proposition 1. The poof of Proposition 3 is obvious.

Note that the description of $t_{14} = \{m_{23}, m_{24}, m_{26}\}$ is closed because of $\text{obj}\{m_{23}, m_{24}, m_{26}\} = \{1, 2, 12, 14\}$ and $\text{val}\{1, 2, 12, 14\} = \{m_{23}, m_{24}, m_{26}\}$. We also know that $s = \{1, 2, 12, 14\}$ is closed too (we obtained this result during generalization of elements of STGOOD). So $(\text{obj}(\{m_{23}, m_{24}, m_{26}\}), \{m_{23}, m_{24}, m_{26}\})$ is a maximally redundant test for positive objects and we can, consequently, delete t_{14} from consideration. As a result of deleting m_{12} and t_{14} , we have the modified set SPLUS (Tab.5).

5.2 Using the Rule of Alternative for Splitting Subcontexts

Essentially, searching for GMRTs turns out a process of generating deductive reasoning rules of the first type, which are used immediately during this process. Now we shall give an example of constructing and using the rules of alternative (forbidden rules).

Table 5. The set SPLUS₁

splus(m), $m \in M$	splus(m), $m \in M$
splus(m_a) \rightarrow {2, 8, 10}	splus(m_{22}) \rightarrow {2, 7, 8, 9, 11}
splus(m_{13}) \rightarrow {3, 8, 10}	splus(m_{23}) \rightarrow {1, 2, 5, 12, 13}
splus(m_{16}) \rightarrow {4, 9, 12}	splus(m_3) \rightarrow {3, 7, 8, 10, 11, 12}
splus(m_1) \rightarrow {1, 4, 11, 13}	splus(m_4) \rightarrow {2, 3, 4, 7, 10, 13}
splus(m_5) \rightarrow {1, 4, 7, 10}	splus(m_6) \rightarrow {1, 4, 5, 7, 8, 10}
	splus(m_7) \rightarrow {2, 3, 4, 6, 8, 11}
splus(m_{18}) \rightarrow {3, 9, 10, 13}	splus(m_{24}) \rightarrow {1, 2, 3, 4, 5, 7, 12}
splus(m_2) \rightarrow {1, 5, 10, 11, 12}	splus(m_{20}) \rightarrow {4, 6, 7, 8, 9, 10, 11, 12}
splus(m_b) \rightarrow {2, 3, 4, 7, 8}	splus(m_{21}) \rightarrow {1, 4, 6, 8, 9, 10, 11, 12}
splus(m_{19}) \rightarrow {3, 8, 9, 11, 13}	splus(m_{26}) \rightarrow {1, 2, 3, 4, 6, 7, 9, 10, 11, 12, 13}

Form all the pairs of objects $(i, j) \subseteq G_+$; form a set FR of forbidden pairs of objects, $FR = \{(i, j) : \text{to_be_test}((i, j), \text{val}(i, j)) \text{ is } false\}$ (Tab. 6); then the set of admissible pairs is $ADM = \{(i, j) : \text{to_be_test}((i, j), \text{val}(i, j)) \text{ is } true\}$ (Tab. 6).

Each forbidden pair (i, j) generates rule of alternative: $j \text{ OR } i \rightarrow true(always)$; $i, j \rightarrow false$. This rule says that i and j cannot simultaneously enter in any extent of GMRTs, either i or j but not both.

These rules of alternatives allow splitting any $splus(m), m \in M$ into subsets not containing any forbidden pair of objects. An example is in Tab. 7.

As a result of splitting, we shall obtain the splitting tree, leaves of which are subsets of objects not containing any prohibited object pairs. Subcontexts for GMRTs inferring are leaves of the tree with the number of objects more or equal to 4. So, in Tab. 7, we have only one subcontext (4,7,8,11) for searching for tests. As far as triplets of objects, if a triplet is not the extent of a test, then we have all information about pairs of objects contained in it.

An analysis of set ADM leads to using some inductive reasoning rules [19]. If element j enters in s_{q+1} , then it must enter in q proper subsets of s_{q+1} , where s_{q+1} is a set containing $(q+1)$ elements. If we observe that j enters in only one doublet (pair), then it cannot enter in any triplet. If we observe that j enters in only one triplet, then it cannot enter in any quadruplet and so on. If an element enters in two and only two doublets, it means that it will enter only in one triplet. If an element enters in three and only three doublets, it can enter in only one quadruplet.

This inductive reasoning is applicable to constructing triplets from doublets, quadruplets from triplets and so on. In our example: Object 9 enters in one and only one doublet, hence (9,11) cannot be included in any triplet. We conclude that (9,11) is the extent of a GMRT. Object 5 enters in two and only two pairs, hence it is included in only one triplet (1,5,12) already contained in the initial content of STGOOD. Object 5 cannot be included in any quadruplet. Object 6

Table 6. Classification of object pairs

g	Admissible pairs of objects	Forbidden pairs of objects
1	(1,2) (1,4) (1,5) (1,7) (1,12)	(1,3) (1,6) (1,8) (1,9) (1,10) (1,11) (1,13)
2	(2,1) (2,3) (2,4) (2,7) (2,8) (2,10) (2,12)	(2,5) (2,6) (2,9) (2,11) (2,13)
3	(3,2) (3,4) (3,7) (3,8) (3,10) (3,11) (3,12)	(3,1) (3,5) (3,6) (3,9) (3,13)
4	(4,1) (4,2) (4,3) (4,6) (4,7) (4,8) (4,11) (4,12)	(4,5) (4,9) (4,10) (4,13)
5	(5,1) (5,12)	(5,2) (5,3) (5,4) (5,6) (5,7) (5,8) (5,9) (5,10) (5,11) (5,13)
6	(6,4) (6,8) (6,11)	(6,1) (6,2) (6,3) (6,5) (6,7) (6,9) (6,10) (6,12) (6,13)
7	(7,1) (7,2) (7,3) (7,4) (7,8) (7,11) (7,12)	(7,5) (7,6) (7,9) (7,10) (7,13)
8	(8,2)(8,3)(8,4) (8,6)(8,7) (8,10) (8,11)	(8,1) (8,5) (8,9) (8,12) (8,13)
9	(9,11)	All the other pairs
10	(10,2) (10,3) (10,8)	(10,1) (10,4) (5,10) (10,6) (10,7) (10,9) (10,11) (10,12) (10,13)
11	(11,3) (11,4) (6,11) (7,11) (11,9)	(11,1) (11,2) (5,11) (11,9) (11,10) (11,12) (11,13)
12	(12,1) (12,2) (12,3) (12,4) (12,5) (12,7)	(12,6) (12,8) (12,9) (12,10) (12,11) (12,13)
13	\emptyset	All the pairs

Table 7. An example of splitting $\text{splus}(m_2)$

No	$\text{splus}(m_2)$	Forbidden pair
1	(4,7,8,11,12)	(8,12)
2	(4,7,8,11), (4,7,11,12)	
3	(4,7,11,12)	(11,12)
4	(4,7,11) (4,7,12)	

enters in three and only three pairs, hence it is included in only one quadruplet (4,6,8,11), already contained in STGOOD. Object 13 is not included in any admissible pair of objects. We put {9, 11} and {13} into STGOOD and delete objects 9, 5, 6, and 13 from consideration. This action implies deleting values m_{16} , m_{18} , and m_{23} (see, please Tab. 8), because $\text{splus}(m_i)$, where i take values 16, 18 or 23, are absorbed by some elements of STGOOD.

Table 8. The set SPLUS₂

$\text{splus}(m), m \in M$	$\ \text{splus}(m), m \in M$
$\text{splus}(m_a) \rightarrow \{2, 8, 10\}$	$\text{splus}(m_{22}) \rightarrow \{2, 7, 8, 11\}$
$\text{splus}(m_{13}) \rightarrow \{3, 8, 10\}$	$\text{splus}(m_{23}) \rightarrow \{1, 2, 12\}$
$\text{splus}(m_{16}) \rightarrow \{4, 12\}$	$\text{splus}(m_3) \rightarrow \{3, 7, 8, 10, 11, 12\}$
$\text{splus}(m_1) \rightarrow \{1, 4, 11\}$	$\text{splus}(m_4) \rightarrow \{2, 3, 4, 7, 10, 13\}$
$\text{splus}(m_5) \rightarrow \{1, 4, 7, 10\}$	$\text{splus}(m_6) \rightarrow \{1, 4, 7, 8, 10\}$
	$\text{splus}(m_7) \rightarrow \{2, 3, 4, 6, 8, 11\}$
$\text{splus}(m_{18}) \rightarrow \{3, 10\}$	$\text{splus}(m_{24}) \rightarrow \{1, 2, 3, 4, 7, 12\}$
$\text{splus}(m_2) \rightarrow \{1, 10, 11, 12\}$	$\text{splus}(m_{20}) \rightarrow \{4, 7, 8, 10, 11, 12\}$
$\text{splus}(m_b) \rightarrow \{2, 3, 4, 7, 8\}$	$\text{splus}(m_{21}) \rightarrow \{1, 4, 8, 9, 10, 11, 12\}$
$\text{splus}(m_{19}) \rightarrow \{3, 8, 11\}$	$\text{splus}(m_{26}) \rightarrow \{1, 2, 3, 4, 7, 10, 11, 12\}$

We analyze triplets in SPLUS: $\text{splus}(m_*) = \{2, 8, 10\}$, $\text{splus}(m_1) = \{1, 4, 11\}$, $\text{splus}(m_{19}) = \{3, 8, 11\}$, and $\text{splus}(m_{13}) = \{3, 8, 10\}$. As a result, we obtain the extents of two new GMRTs {3, 11} and {8, 10}. After that, we can delete object 10. Then we can delete values m_5, m_2 , and m_4 after analyzing their modified splus . Now we split $\text{splus}(m_{22}), \text{splus}(m_{20}), \text{splus}(m_{21}), \text{splus}(m_{24})$, and $\text{splus}(m_{26})$, because these values are the only remaining essential values in the set LEV. Tab. 9 shows the number of subtasks in splitting trees for these essential values and new obtained tests.

Table 9. Splitting $\text{splus}(m)$, m is essential value

$\text{splus}(m)$	$\ \text{The number of subtasks}$	$\ \text{New tests}$	$\ \text{Deleted elements}$
$\text{splus}(m_{22})$	1	{7, 8, 11}	m_{26}
$\text{splus}(m_{20})$			m_{20}
$\text{splus}(m_{21})$			m_{21}, t_8
$\text{splus}(m_{24})$	2		m_{24}, t_1, t_{12}, t_7 ; remaining set of indices {2, 3, 4} is absorbed by element {2, 3, 4, 7} in STGOOD

Some words about future investigations. The most important mathematical tasks worth solving are the following:

1. For irredundant $\text{splus}(m)$ to determine whether it contains extents of new GMRTs;
2. Optimization of selecting forbidden pairs of objects in order not to obtain repetition of subsets in splitting trees.

6 Conclusion

A sketch of classification reasoning is given in the paper. The key notions of the reasoning are notions of classification and its good approximations based on good diagnostic tests. The good diagnostic tests have the dual nature: they generate functional and implicative dependencies and, in the same time, sets of objects satisfying these dependencies. The dependencies are logical assertions on the base of which deductive phase of classification reasoning is realized.

In the paper, we deal only with GMRTs as value implications. An algorithm for inferring GMRTs from a set of data is given. A decomposition of the main task of inferring GMRTs into subtasks associated with the certain types of sub-contexts is introduced.

There are various ways of using the proposed decompositions and extracted initial information about GMRTs, but we confine ourselves to the most important ideas allowing to obtain not only tests (and corresponding knowledge) but also, simultaneously, contexts inside of which these tests have been inferred. Currently, there is considerable interest in contextual information search and representation for using it to improve web search results or query expansion [3,4,10]. Classification reasoning can serve as a basic tool for constructing and using content-dependent knowledge in different problem domains and applications.

References

1. Baixeries, J., Kaytoue, M., Napoli, A.: Computing functional dependencies with pattern structures. In: Szathmary, L., Priss, U. (eds.) Proceedings of The Ninth International Conference on Concept Lattices and Their Applications. CEUR Workshop Proceedings, vol. 972, pp. 175–186. CEUR-WS.org (2012)
2. Banerji, R.B.: Theory of problem solving: an approach to artificial intelligence, vol. 17. Elsevier Publishing Company (1969)
3. Bankar, S., Nagpure, R.: Contextual information search based on domain using query expansion. In: International Journal of Emerging Trends and Technology in Computer Science. vol. 3, pp. 148 – 151 (2014)
4. Bouramoul, A., Kholadi, M.K., Doan, B.L.: PRESY: A Context Based Query Reformulation Tool for Information Retrieval on the Web. Journal of Computer Science 6(4), 470–477 (2010)
5. Ferré, S., Ridoux, O.: The use of associative concepts in the incremental building of a logical context. In: Priss, U., Corbett, D., Angelova, G. (eds.) ICCS. Lecture Notes in Computer Science, vol. 2393, pp. 299–313. Springer (2002)

6. Finn, V.: The synthesis of cognitive procedures and the problem of induction. *NTI ser.2*, 1-2, 8–44 (1999), (in Russian)
7. Ganter, B., Wille, R.: *Formal concept analysis: mathematical foundations*. Springer, Berlin (1999)
8. Ganter, B., Kuznetsov, S.O.: Formalizing hypotheses with concepts. In: *Proceedings of the Linguistic on Conceptual Structures: Logical Linguistic, and Computational Issues*. pp. 342–356. Springer-Verlag (2000)
9. Ganter, B., Kuznetsov, S.: Pattern structures and their projections. In: Delugach, H., Stumme, G. (eds.) *Conceptual Structures: Broadening the Base*, *Lecture Notes in Computer Science*, vol. 2120, pp. 129–142. Springer Berlin Heidelberg (2001)
10. Klimešová, D., Ocelíková, E.: Study on context understanding, knowledge transformation and decision support systems. *WSEAS Trans. Info. Sci. and App.* 7(7), 985–994 (Jul 2010)
11. Kuznetsov, S.: Machine learning on the basis of formal concept analysis. *Automation and Remote Control* 62(10), 1543–1564 (2001)
12. Mill, J.S.: *The System of Logic Ratiocinative and Inductive Being a Connected View of the Principles of Evidence, and the Methods of Scientific Investigation*. Longmans (1872)
13. Naidenova, K.: Reducing one class of machine learning algorithms to logical operations of plausible reasoning. *Journal of Computer and Systems Sciences International* 48(3), 401–414 (2009)
14. Naidenova, X.A., Plaksin, M.V., Shagalov, V.L.: Inductive Inferring All Good Classification Tests. In: Valkman, J. (ed.) "Knowledge-Dialog-Solution", *Proceedings of International Conference*. vol. 1, pp. 79 – 84. Kiev Institute of Applied Informatics, Kiev, Ukraine (1995)
15. Naidenova, X.: Machine Learning as a diagnostic task. In: Arefiev, I. (ed.) *Knowledge-Dialog-Solution, Materials of the Short-Term Scientific Seminar*, pp. 26 – 36. State North-West Technical University Press, St Petersburg, Russia (1992)
16. Naidenova, X.: The Data-Knowledge Transformation. In: Solovyev, V. (ed.) *Text Processing and Cognitive Technologies*, vol. 3, pp. 130–151. Pushchino (1999)
17. Naidenova, X.: Good classification tests as formal concepts. In: Domenach, F., Ignatov, D., Poelmans, J. (eds.) *Formal Concept Analysis*, *Lecture Notes in Computer Science*, vol. 7278, pp. 211–226. Springer Berlin Heidelberg, Leuven (2012)
18. Naidenova, X., Ermakov, A.: The decomposition of good diagnostic test inferring algorithms. In: Alty, J., Mikulich, L., Zakrevskij, A. (eds.) "Computer-Aided Design of Discrete Devices" (CAD DD2001), *Proceedings of the 4-th Inter. Conf.*, vol. 3, pp. 61 – 68. Minsk (2001)
19. Naidenova, X.: An incremental learning algorithm for inferring logical rules from examples in the framework of the common reasoning process. In: Triantaphyllou, E., Felici, G. (eds.) *Data Mining and Knowledge Discovery Approaches Based on Rule Induction Techniques*, *Massive Comp.*, vol. 6, pp. 89–147. Springer (2006)
20. Naidenova, X.A., Parkhomenko, V.: Attributive and object subcontexts in inferring good maximally redundant tests. In: Bertet, K., Rudolph, S. (eds.) *Proceedings of the Eleventh International Conference on Concept Lattices and Their Applications*, Košice, Slovakia, October 7-10, 2014. *CEUR Workshop Proceedings*, vol. 1252, pp. 181–193. CEUR-WS.org (2014)
21. Ore, O.: Galois connections. *Trans. Amer. Math. Soc* 55, 494–513 (1944)
22. Rasiowa, H.: *An Algebraic Approach to Non-classical Logics*. *Studies in logic and the foundations of mathematics*, North-Holland Publishing Company (1974)