Replacing full rectangles by dense rectangles: concept lattices and attribute implications

Radim Belohlavek and Vilem Vychodil
Dept. Comp. Science, Palacky University, Tomkova 40, CZ-779 00, Olomouc, Czech Republic
e-mail: {radim.belohlavek, vilem.vychodil}@upol.cz

***Abstract*— Maximal full rectangles in tabular data are useful in several areas of data engineering. This paper presents a survey of results in which we replace "full rectangles" by "dense rectangles". This way, we go from exact to approximate. We develop issues directly related to maximal dense rectangles: closure-like structures, concept lattices, attribute implications, a computationally tractable description of non-redundant bases of implications, and an algorithm for their computation. We present illustrative examples and results of experiments.**

I. Problem Setting

Data tables describing objects (table rows), attributes (table columns) and their relationship (table entry is $\times$ or blank depending on whether an object has or does not have an attribute) belong to fundamental means of data representation. An example is presented in Fig. 1. In extracting useful information from such data tables, maximal rectangles (i.e. rectangular subtables) which are full of $\times$'s proved to be very useful. Two exemplary areas are formal concept analysis (FCA), see [5], [6], and if then rules (called also attribute implications [6], association rules in data mining [1], [13] or functional dependencies in databases [2], [9]).

Maximal full rectangles are certain patterns in data with surprisingly nice properties, connections to some useful mathematical structures (lattices, Galois connections, etc.), and are computationally tractable (an algorithm with polynomial time delay for computing all maximal full rectangles from a given table exists [6]). In this paper, we present a survey of results inspired by the following question: What happens if we replace "maximal full rectangle" by "maximal dense rectangle"? By a dense rectangle we mean a rectangle which contains at most a reasonable small number of reasonably distributed blanks. This question is legitimate since full rectangles do not capture all interesting rectangular patterns and can be seen as extremal case of dense rectangles. Replacing "full" by "dense" corresponds to replacing strict conditions by approximate ones. Note that there exist related approaches. For instance, the authors in [4] consider full rectangles instead of maximal full rectangles.

Section II recalls preliminaries. Section III presents basic considerations and definitions related to the concept of a dense rectangle. Section IV surveys results concerning formal concepts and concept lattices built over dense rectangles. Section V is devoted to attribute implications with validity defined by means of dense rectangles. Illustrative examples and further issues are the content of Section VI. Due to a limited extent of the paper, we omit proofs of theorems (they will be published in a full version of the paper).

The main message of the paper is the following: By replacing full rectangles with dense rectangles we focus on different patterns. Still, many of the nice properties due to which maximal full rectangles are useful, are available for dense rectangles as well (related structures like Galois connections and concept lattices, approximate validity of if-then rules based on dense rectangles and computationally tractable description of non-redundant bases of all approximately valid if-then rules, etc.). Therefore, dense rectangles are worth of further study, both experimental and theoretical.

		a	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*
leech	1	×	×					×		
bream	2	×	×					×	×	
frog	3	×	×	×				×	×	
dog	4	×		×				×	×	×
spike-weed	5	×	×		×		×			
reed	6	×	×	×	×		×			
bean	7	×		×	×	×				
maize	8	×		×	×		×			

Fig. 1. Data table [6]; the attributes are: *a*: needs water to live, *b*: lives in water, *c*: lives on land, *d*: needs chlorophyll to produce food, *e*: two seed leaves, *f*: one seed leaf, *g*: can move around, *h*: has limbs, *i*: suckles its offspring.

II. Preliminaries

This section fixes terminology, notation, and recalls basic notions of data tables and formal concept analysis [6] which will be used throughout the paper. *A data table* (*with binary attributes*) can be identified with a triplet $\langle X,Y,I\rangle$ where X is a non-empty finite set (of *objects*), Y is a non-empty finite set (of *attributes*), and $I \subseteq X \times Y$ is an (object-attribute) relation. If $\langle x,y\rangle \in I$ (indicated by $\times$ in the table), we say that object x has attribute y, If $\langle x,y\rangle \notin I$ (indicated by blank in the table), we say that object x does not have attribute y. For each $A \subseteq X$ and $B \subseteq Y$ denote by $A^{\Uparrow}$ and $B^{\Downarrow}$ a subset of Y and a subset of X defined by

$$A^{\Uparrow} = \{y \in Y \mid \text{for each } x \in A\colon \langle x,y\rangle \in I\},$$
$$B^{\Downarrow} = \{x \in X \mid \text{for each } y \in B\colon \langle x,y\rangle \in I\}.$$

That is, $A^{\Uparrow}$ is the set of all attributes from Y shared by all objects from A (and similarly for $B^{\Downarrow}$). A *formal concept* in $\langle X,Y,I\rangle$ is a pair $\langle A,B\rangle$ of $A \subseteq X$ and $B \subseteq Y$ satisfying $A^{\Uparrow} = B$ and $B^{\Downarrow} = A$. That is, a formal concept consists of a set A (so-called *extent*) of objects which are covered by the concept and a set B (so-called *intent*) of attributes which are covered by the concept such that A is the set of all objects sharing all attributes from B and, conversely, B is the collection of all attributes from Y shared by all objects from A. Thus, formal concepts in $\langle X,Y,I\rangle$ represent particular clusters which are hidden in $\langle X,Y,I\rangle$. Alternatively, formal concepts can be defined as maximal rectangles of $\langle X,Y,I\rangle$ which are full of $\times$'s: For $A \subseteq X$ and $B \subseteq Y$, $\langle A,B\rangle$ is a formal concept in $\langle X,Y,I\rangle$ iff $A \times B \subseteq I$ and there is no $A' \supset A$ or $B' \supset B$ such that $A' \times B \subseteq I$ or $A \times B' \subseteq I$. The set $\mathcal{B}(X,Y,I) = \{\langle A,B\rangle \mid A^{\Uparrow} = B, B^{\Downarrow} = A\}$ of all formal concepts in $\langle X,Y,I\rangle$ can be equipped with a partial order $\leq$ (modeling the subconcept-superconcept hierarchy) defined by

$$\langle A_1,B_1\rangle \leq \langle A_2,B_2\rangle \quad \text{iff} \quad A_1 \subseteq A_2 \quad (\text{or, equivalently, } B_2 \subseteq B_1).$$

Under $\leq$, $\mathcal{B}(X,Y,I)$ happens to be a complete lattice, called a concept lattice, the basic structure of which is described by the so-called main theorem of concept lattices [6], [12].

An *attribute implication* (*over* Y) is an if-then rule of the form $A \Rightarrow B$, where $A,B \subseteq Y$ are sets of attributes. Except for FCA, rules of this form are widely used in several disciplines like data mining

0-7803-9788-6/06/$20.00 ©2006 IEEE.

(as association rules extracted from data), database systems (as rules describing functional dependencies). A primary interpretation of attribute implications in data tables is the following. An attribute implication $A \Rightarrow B$ is true in a data table $\langle X,Y,I\rangle$ if for each object $x \in X$: if x has all attributes from A then it has also all attributes from B. Realizing that "object x has all attributes from A" is equivalent to $x \in A^{\Downarrow}$, we can restate the interpretation as follows: $A \Rightarrow B$ is true in $\langle X,Y,I\rangle$ if for each object $x \in X$: if $x \in A^{\Downarrow}$ then $x \in B^{\Downarrow}$. That is, $A \Rightarrow B$ is true in $\langle X,Y,I\rangle$ if $A^{\Downarrow} \subseteq B^{\Downarrow}$.

Example 1: Consider data table $\langle X,Y,I\rangle$ from Fig. 1. For instance, $\{f\} \Rightarrow \{d\}$ is true in the table while $\{a,b\} \Rightarrow \{f\}$ is not (frog serves as a counterexample).

From the point of view of association rules (in sense of Agrawal et al. [1]), attribute implications which are true in data are the so-called *exact association rules*, i.e. association rules with confidence 1.

III. Dense rectangles

A *rectangle* over sets X and Y is a pair $\langle A,B\rangle$ with $A \subseteq X$ and $B \subseteq Y$. A rectangle $\langle A,B\rangle$ is a *subrectangle of a rectangle* $\langle C,D\rangle$ ($\langle C,D\rangle$ is a *superrectangle of* $\langle A,B\rangle$) if $A \subseteq C$ and $B \subseteq D$. Occasionally, we also speak of a rectangle $\langle A,B\rangle$ in a data table $\langle X,Y,I\rangle$. A rectangle $\langle A,B\rangle$ in $\langle X,Y,I\rangle$ corresponds to a subtable (submatrix) of table $\langle X,Y,I\rangle$ delineated by rows given by objects from A and columns given by attributes from B. For brevity, the corresponding subtables will also be called rectangles. $\langle A,B\rangle$ is a *proper* subrectangle of a rectangle $\langle C,D\rangle$ if $\langle A,B\rangle$ is a subrectangle of $\langle C,D\rangle$, and $A \subset C$ or $B \subset D$.

We will consider properties of rectangles over given sets X and Y. For a property $\mathcal{D}$ and a data table $\langle X,Y,I\rangle$, we denote by $\mathcal{D}(A,B)$ (or $\mathcal{D}_I(A,B)$) the fact that *rectangle* $\langle A,B\rangle$ *has property* $\mathcal{D}$ *in* $\langle X,Y,I\rangle$. In particular, we are interested in properties $\mathcal{D}$ such that $\mathcal{D}(A,B)$ means that $\langle A,B\rangle$ is a dense rectangle in $\langle X,Y,I\rangle$. By $\langle A,B\rangle$ being dense we mean that "almost all entries of a subtable of $\langle X,Y,I\rangle$ given by $\langle A,B\rangle$ contain $\times$". In the following, we will be concerned with properties which result by imposing restrictions on the number of blanks in columns. We call these properties column-like properties. A definition follows.

Definition 2: Let $\mathcal{D}$ be a property of rectangles over sets X and Y, $\langle X,Y,I\rangle$ be a data table. If there is an Y-indexed collection $\mathbf{l} = \{l_y \,|\, y \in Y\}$ of non-negative integers l_y such that $\mathcal{D}_I(A,B)$ iff, for each $y \in B$, we have $|\{x \in A \,|\, \langle x,y\rangle \notin I\}| \leq l_y$, then $\mathcal{D}$, denoted by $col(\mathbf{l})$, is called a *column-like property in* $\langle X,Y,I\rangle$. If, for each $y \in Y$, $l_y = l$ then $\mathcal{D}$ is denoted by $col(l)$.

If for each $I' \subseteq X \times Y$, $\mathcal{D}$ is a column-like property in $\langle X,Y,I'\rangle$, then $\mathcal{D}$ is called a *column-like property.*

Column-like properties can be axiomatized (we omit details). In an analogous way, one can introduce row-like properties.

Remark 3: By definition, if $\mathcal{D}$ is a column-like property $col(\mathbf{l})$, where $\mathbf{l} = \{l_y \,|\, y \in Y\}$, then a rectangle $\langle A,B\rangle$ has property $\mathcal{D}$ iff, in each column $y \in B$, the number of blanks is at most l_y. If $\mathcal{D}$ is $col(l)$, where l is a single non-negative integer, then $\langle A,B\rangle$ has property $\mathcal{D}$ iff, in each column $y \in B$, the number of blanks does not exceed l. Thus, $col(0)$ means "no blank in any column", $col(2)$ means "at most two blanks in each column", etc.

Examples will be presented in Section VI.

IV. Maximal dense rectangles: concept lattices and related structures

Let $\langle X,Y,I\rangle$ be a data table, $\mathcal{D}$ be a property of rectangles over sets X and Y. A rectangle $\langle A,B\rangle$ in $\langle X,Y,I\rangle$ is called *maximal* w.r.t. $\mathcal{D}$ if $\langle A,B\rangle$ has $\mathcal{D}$ and no proper superrectangle of $\langle A,B\rangle$ has $\mathcal{D}$. In case of maximal full rectangles in $\langle X,Y,I\rangle$ (i.e., rectangles which are full of $\times$'s), given $A \subseteq X$, there is a unique maximal (hence, the largest) B such that $\langle A,B\rangle$ is a rectangle full of $\times$'s, namely $B = A^{\Uparrow}$. Likewise, for $B \subseteq Y$, $B^{\Downarrow}$ is the largest one such that $\langle B^{\Downarrow},B\rangle$ is a rectangle full of $\times$'s. The mappings $^{\Uparrow}$ and $^{\Downarrow}$ form a so-called Galois connection between $\langle 2^X,\subseteq\rangle$ and $\langle 2^Y,\subseteq\rangle$ and maximal full rectangles are just fixed points of $^{\Uparrow}$ and $^{\Downarrow}$, i.e. pairs $\langle A,B\rangle$ satisfying $A^{\Uparrow} = B$ and $B^{\Downarrow} = A$. The set of all these fixed points is just a concept lattice $\mathcal{B}(X,Y,I)$.

For general column-like properties $\mathcal{D}$, the situation is different. While for $A \subseteq X$ there is still a largest $B \subseteq Y$ such that $\mathcal{D}(A,B)$, for $B \subseteq Y$, there might be several maximal sets $A \subseteq X$ such that $\mathcal{D}(A,B)$. In the following, we propose an approach in which maximal dense rectangles are identified by means of fixed points of mappings resembling very much Galois connections. The mapping will be described in the sequel.

Let $\leq$ be a binary relation defined on 2^{2^X} by

$$\mathcal{A}_1 \leq \mathcal{A}_2 \quad \text{iff} \quad \text{for each } A_1 \in \mathcal{A}_1 \text{ there is } A_2 \in \mathcal{A}_2 \text{ such that } A_1 \subseteq A_2, \tag{1}$$

for each sets $\mathcal{A}_1, \mathcal{A}_2 \in 2^{2^X}$. A couple $\langle 2^{2^X},\leq\rangle$ is a *quasiordered set.* That is, $\leq$ is a quasiorder (i.e., a reflexive and a transitive relation) on 2^{2^X}. Note that if $X \neq \emptyset$ then $\leq$ defined on 2^{2^X} is not antisymmetric, i.e. it is not a partial order. For a quasiordered set $\langle 2^{2^X},\leq\rangle$, we define a binary relation $\equiv_{\leq}$ on 2^{2^X} by

$$\mathcal{A}_1 \equiv_{\leq} \mathcal{A}_2 \quad \text{iff} \quad \mathcal{A}_1 \leq \mathcal{A}_2 \text{ and } \mathcal{A}_2 \leq \mathcal{A}_1.$$

It is well known that $\equiv_{\leq}$ is an equivalence in 2^{2^X}.

Definition 4: Let $\langle X,Y,I\rangle$ be a data table, $\mathcal{D}$ be a column-like property. For each $A \in 2^X$, let the largest B with $\mathcal{D}(A,B)$ be denoted by $A^{\uparrow}$. For each $\mathcal{A} \in 2^{2^X}$ and $B \in 2^Y$, put

$$\mathcal{A}^{\uparrow} = \textstyle\bigcap_{A \in \mathcal{A}} A^{\uparrow}, \qquad B^{\downarrow} = \{A \in 2^X \,|\, A \text{ is maximal such that } \mathcal{D}(A,B)\}.$$

Remark 5: Note that in Definition 4, we use $^{\uparrow}$ for a mapping of 2^X to 2^Y as well as for a mapping of 2^{2^X} to 2^Y. It will always be clear from the context which mapping we mean by $^{\uparrow}$. Note also that $^{\uparrow}: 2^{2^X} \to 2^Y$ is an extension of $^{\uparrow}: 2^X \to 2^Y$ in that for $A \subseteq X$ we have $A^{\uparrow} = \{A\}^{\uparrow}$. $B^{\downarrow}$ is a set of all A's such that $\langle A,B\rangle$ is a maximal dense rectangle.

The following theorem shows properties of $\uparrow$ and $\downarrow$.

Theorem 6: For all $\mathcal{A}_1, \mathcal{A}_2 \in 2^{2^X}$ and $B_1, B_2 \in 2^Y$:

$$\mathcal{A}_1 \leq \mathcal{A}_2 \text{ implies } \mathcal{A}_2^{\uparrow} \subseteq \mathcal{A}_1^{\uparrow}, \quad B_1 \subseteq B_2 \text{ implies } B_2^{\downarrow} \leq B_1^{\downarrow}, \tag{2}$$

$$\mathcal{A} \leq \mathcal{A}^{\uparrow\downarrow}, \quad B \subseteq B^{\downarrow\uparrow}. \tag{3}$$

Call any pair of mappings $^{\uparrow}: U \to V$ and $^{\downarrow}: V \to U$ between a quasiordered set $\langle U,\leq\rangle$ and a partially ordered set $\langle V,\subseteq\rangle$ satisfying (2) and (3) a *Galois connection* between $\langle U,\leq\rangle$ and $\langle V,\subseteq\rangle$. Therefore, $^{\uparrow}$ and $^{\downarrow}$ introduced above form a Galois connection between $\langle 2^{2^X},\leq\rangle$ and $\langle 2^Y,\subseteq\rangle$. Basic properties of Galois connections between a quasiordered set and a partially ordered set are slightly different from the ordinary case and are shown in the next

Theorem 7: Mappings $^{\uparrow}: 2^{2^X} \to 2^Y$ and $^{\downarrow}: 2^Y \to 2^{2^X}$ satisfying (2) and (3) have the following properties.

(i) $\mathcal{A}^{\uparrow} = \mathcal{A}^{\uparrow\downarrow\uparrow}$ for each $\mathcal{A} \in 2^{2^X}$.
(ii) $B^{\downarrow} \equiv_{\leq} B^{\downarrow\uparrow\downarrow}$ and $B^{\downarrow\uparrow\downarrow\uparrow} = B^{\downarrow\uparrow}$ for each $B \in 2^Y$.
(iii) A mapping $C_X: 2^{2^X} \to 2^{2^X}$ defined by $C_X(\mathcal{A}) = \mathcal{A}^{\uparrow\downarrow}$ satisfies: $\mathcal{A} \leq C_X(\mathcal{A})$, $\mathcal{A}_1 \leq \mathcal{A}_2$ implies $C_X(\mathcal{A}_1) \leq C_X(\mathcal{A}_2)$; $C_X(\mathcal{A}) \equiv_{\leq} C_X(C_X(\mathcal{A}))$ and $C_X(C_X(\mathcal{A})) = C_X(C_X(C_X(\mathcal{A})))$.
(iv) A mapping $C_Y: 2^Y \to 2^Y$ defined by $C_Y(B) = B^{\downarrow\uparrow}$ is a closure operator in $\langle 2^Y,\subseteq\rangle$, i.e. C_Y satisfies: $B \subseteq C_Y(B)$, $B_1 \subseteq B_2$ implies $C_Y(B_1) \subseteq C_Y(B_2)$; $C_Y(B) = C_Y(C_Y(B))$.

Definition 8: Let $\langle X,Y,I\rangle$ be a data table, $\mathcal{D}$ be a column-like property. A $\mathcal{D}$-concept lattice of $\langle X,Y,I\rangle$ is a set

$$\mathcal{B}_{\mathcal{D}}(X,Y,I) = \{\langle \mathcal{A},B\rangle \in 2^{2^X} \times 2^Y \,|\, \mathcal{A}^{\uparrow} = B, B^{\downarrow} = \mathcal{A}\}$$

equipped with a binary relation $\leq$ defined by

$$\langle \mathcal{A}_1, B_1\rangle \leq \langle \mathcal{A}_2, B_2\rangle \quad \text{iff} \quad \mathcal{A}_1 \leq \mathcal{A}_2 \ (\text{iff } B_1 \supseteq B_2).$$

Following further the terminology of formal concept analysis, pairs $\langle \mathcal{A}, B\rangle \in \mathcal{B}_{\mathcal{D}}(X,Y,I)$ are called (*formal*) $\mathcal{D}$*-concepts*. If $\langle \mathcal{A}, B\rangle \in \mathcal{B}_{\mathcal{D}}(X,Y,I)$, $\mathcal{A}$ is called a $\mathcal{D}$*-extent*, B is called a $\mathcal{D}$*-intent*. A collection of all $\mathcal{D}$-intents will be denoted by $\mathrm{Int}_{\mathcal{D}}(X,Y,I)$, i.e.

$$\mathrm{Int}_{\mathcal{D}}(X,Y,I) = \{B \in 2^Y \mid \langle \mathcal{A}, B\rangle \in \mathcal{B}_{\mathcal{D}}(X,Y,I) \text{for some } \mathcal{A} \in 2^{2^X}\}.$$

The structure of $\mathcal{D}$-concepts is characterized by the following theorem.

Theorem 9: For a column-like property $\mathcal{D}$ and a data table $\langle X,Y,I\rangle$, $\mathcal{B}_{\mathcal{D}}(X,Y,I)$ equipped with $\leq$ is a complete lattice with infima $\bigwedge$ and suprema $\bigvee$ given by

$$\textstyle\bigwedge_{j\in J}\langle \mathcal{A}_j, B_j\rangle = \langle \{\bigcap_{j\in J} A_j | A_j \in \mathcal{A}_j\}^{\uparrow\downarrow\uparrow\downarrow}, (\bigcup_{j\in J} B_j)^{\downarrow\uparrow}\rangle,$$
$$\textstyle\bigvee_{j\in J}\langle \mathcal{A}_j, B_j\rangle = \langle (\bigcup_{j\in J} \mathcal{A}_j)^{\uparrow\downarrow\uparrow\downarrow}, \bigcap_{j\in J} B_j\rangle.$$

Remark 10: Note that the Main theorem of concept lattices [6] follows directly from Theorem 9: if we consider $\mathrm{col}(0)$ as a property $\mathcal{D}$, then we can show that $\mathcal{B}_{\mathcal{D}}(X,Y,I)$ is isomorphic to $\mathcal{B}(X,Y,I)$.

We have $\mathcal{B}_{\mathcal{D}}(X,Y,I) = \{\langle B^{\downarrow}, B\rangle \mid B \in \mathrm{fix}(C_Y)\}$ where $\mathrm{fix}(C_Y) = \{B \subseteq Y \mid B = C_Y(B)\}$ is a set of all fixed points of C_Y (note that $C_Y(B) = B^{\downarrow\uparrow}$). Thus, B is a $\mathcal{D}$-intent iff $B = B^{\downarrow\uparrow}$. Therefore, in order to obtain $\mathcal{B}_{\mathcal{D}}(X,Y,I)$ it suffices to compute $\mathrm{fix}(C_Y)$. Theorem 7 (iv) says that C_Y is a closure operator in $\langle 2^Y, \subseteq\rangle$. An algorithm for computing of all fixed points of a given closure operator is known (NEXTCLOSURE, also known as Ganter's algorithm, see [6]) and works with polynomial time delay. In order to apply NEXTCLOSURE, we need to be able to compute $C_Y(B)$ (for $B \subseteq Y$).

To find an efficient algorithm for computation of C_Y for a general column-like property $\mathcal{D}$ seems to be an interesting problem. Note that for particular choices of $\mathcal{D}$, we can use the following idea. Let $\mathcal{D}$ be given by $col(\mathbf{l})$, where $\mathbf{l} = \{l_y \mid y \in Y, l_y = 0 \text{ or } l_y = 1\}$, see above. That is, we allow for at most one zero in columns y with $l_y = 1$. In order to compute $C_Y(B) = B^{\downarrow\uparrow}$, we need to compute $\mathcal{A} = B^{\downarrow}$ and $\mathcal{A}^{\uparrow}$ (namely, $C_Y(B) = \mathcal{A}^{\uparrow}$).

Given $\mathcal{A} \in 2^{2^X}$, computation of $\mathcal{A}^{\uparrow}$ is obvious. Namely, we have $\mathcal{A}^{\uparrow} = \bigcap_{A\in\mathcal{A}} A^{\uparrow}$ and $A^{\uparrow} = \{y \in Y \mid \mathcal{D}(A, \{y\})\}$. In order to compute $B^{\downarrow}$, put $B_0 = \{y \in B \mid l_y = 0\}$, $B_1 = \{y \in B \mid l_y = 1\}$, and consider the following undirected graph G. Vertices: The set of vertices of G is the set $X - (B^{\Downarrow} \cup Z)$ where

$$Z = \{x \in X \mid \text{there is } y \in B_0 : \langle x, y\rangle \notin I\}.$$

That is, vertices are particular objects from X. Edges: There is an edge between vertices x_1 and x_2 of G iff there is no $y \in B_1$ such that $\langle x_1, y\rangle \notin I$ and $\langle x_2, y\rangle \notin I$, i.e. neither of x_1 and x_2 has attribute y.

Recall that a clique in G is any set M of vertices of G such that for each $x_1, x_2 \in M$ there is an edge between x_1 and x_2. A clique M is maximal if no other vertex can be added to M so that M be still a clique. For technical reasons, if the set of vertices of G is empty, we consider $\emptyset$ (empty set) as a clique of G (this is then the only maximal clique of G). It is then easy to see the following assertion.

Lemma 11: For $B \subseteq Y$ we have

$$B^{\downarrow} = \{B^{\Downarrow} \cup M \mid M \text{ is a maximal clique in } G\}.$$

Recall that efficient algorithms for listing all maximal cliques exist (see e.g. [8]). For more general properties $\mathcal{D}$, the same idea leads to analogous clique-characterizations. For instance, for D being $col(l)$, we get maximal cliques in uniform hypergraph with edges of size $l-1$. These topics need to be explored both theoretically and experimentally.

V. ATTRIBUTE IMPLICATIONS BASED ON DENSE RECTANGLES: FROM APPROXIMATE VALIDITY TO NON-REDUNDANT BASES

In this section we develop attribute implications from the point of view of dense rectangles: their validity and results leading to computationally tractable definition of non-redundant bases.

A. Approximate validity by means of dense rectangles

First, we introduce a notion of a $\mathcal{D}$-truth (a kind of approximate validity) of a given attribute implication in a data table. We have seen above that in the ordinary case, $A \Rightarrow B$ is true in $\langle X,Y,I\rangle$ iff $A^{\Downarrow} \subseteq B^{\Downarrow}$. Realizing that the partial order $\subseteq$ is replaced by a quasiorder $\leq$ in the setting of dense rectangles leads to the following definition.

Definition 12: Let $\langle X,Y,I\rangle$ be a data table, $\mathcal{D}$ be a column-like property in $\langle X,Y,I\rangle$, $A \Rightarrow B$ be an attribute implication over Y. $A \Rightarrow B$ is called $\mathcal{D}$-true in $\langle X,Y,I\rangle$, written $||A \Rightarrow B||_I^{\mathcal{D}} = 1$, if $A^{\downarrow} \leq B^{\downarrow}$. If $A \Rightarrow B$ is not $\mathcal{D}$-true in $\langle X,Y,I\rangle$, we put $||A \Rightarrow B||_I^{\mathcal{D}} = 0$.

By (1) and Definition 12, $A \Rightarrow B$ is $\mathcal{D}$-true in data table $\langle X,Y,I\rangle$, if for each $M \in A^{\downarrow}$ there is $N \in B^{\downarrow}$ such that $M \subseteq N$. That is, $A \Rightarrow B$ is $\mathcal{D}$-true in the table if for each (dense) rectangle $\langle M,A\rangle$ where $M \in A^{\downarrow}$ there is a (dense) rectangle $\langle N,B\rangle$ such that $N \in B^{\downarrow}$ and $\langle M,A\rangle$ is vertically contained in $\langle N,B\rangle$. Recall that for rectangles $\langle M,A\rangle$ and $\langle N,B\rangle$, we have $\mathcal{D}(M,A)$ and $\mathcal{D}(N,B)$. If $\mathcal{D}$ differs from $col(0)$, both the rectangles can contain blanks.

Remark 13: Even if the notions of truth and $\mathcal{D}$-truth of attribute implications are different in general, for $\mathcal{D}$ being $col(0)$, the notion of a truth in a data table coincides with the notion of a $\mathcal{D}$-truth (i.e., $col(0)$-truth) in a data table.

In the sequel we show several equivalent formulations of $\mathcal{D}$-truth and show that $\mathcal{D}$-truth can be expressed as a validity in all dense intents. For technical reasons, we introduce the following notation. An attribute implication $A \Rightarrow B$ is true (valid) in $M \subseteq Y$, written $||A \Rightarrow B||_M = 1$, if we have:

$$\text{if } A \subseteq M \text{ then } B \subseteq M.$$

If $A \Rightarrow B$ is not true in M we put $||A \Rightarrow B||_M = 0$. Note that in $||\cdots||_M$ we do not use superscript $^{\mathcal{D}}$ because the definition of $||\cdots||_M$ does not depend on $\mathcal{D}$.

Lemma 14: The following assertions are equivalent:
(i) $A \Rightarrow B$ is $\mathcal{D}$-true in $\langle X,Y,I\rangle$, (ii) $B \subseteq A^{\downarrow\uparrow}$, (iii) $||A \Rightarrow B||_{A^{\downarrow\uparrow}} = 1$.

If we are given a data table $\langle X,Y,I\rangle$ and a column-like property $\mathcal{D}$, the set $\mathrm{Int}_{\mathcal{D}}(X,Y,I)$ of $\mathcal{D}$-intents is a set of subsets of Y. Thus, for an attribute implication $A \Rightarrow B$, we can ask if $||A \Rightarrow B||_M = 1$ for each $\mathcal{D}$-intent $M \in \mathrm{Int}_{\mathcal{D}}(X,Y,I)$. This way we obtain a natural notion of a truth (validity) of attribute implications in a collection of all dense intents. Interestingly enough, the next assertion shows that the attribute implications which are $\mathcal{D}$-true in $\langle X,Y,I\rangle$ are exactly the attribute implications which are true in each $\mathcal{D}$-intent.

Theorem 15: Let $\langle X,Y,I\rangle$ be a data table, $\mathcal{D}$ be a column-like property in $\langle X,Y,I\rangle$, $A \Rightarrow B$ be an attribute implication over Y. Then $||A \Rightarrow B||_I^{\mathcal{D}} = 1$ if and only if, for each $M \in \mathrm{Int}_{\mathcal{D}}(X,Y,I)$, $||A \Rightarrow B||_M = 1$.

B. Completeness of sets of implications

In this section we characterize $\mathcal{D}$-true attribute implications using entailments from particular sets of attribute implications.

Given a set T of attribute implications, $M \subseteq Y$ is called a *model* of T if, for each $A \Rightarrow B \in T$, $||A \Rightarrow B||_M = 1$. The system of all models of T will be denoted by $\mathrm{Mod}(T)$. An attribute implication $A \Rightarrow B$ *semantically follows* from a set T of attribute implications, written $||A \Rightarrow B||_T = 1$, if $A \Rightarrow B$ is true in each model of T, see also [6].

Definition 16: A set T of attribute implications is called $\mathcal{D}$-complete in data table $\langle X,Y,I\rangle$ if, for each $A \Rightarrow B$, $||A \Rightarrow B||_I^{\mathcal{D}} = 1$ if and only if $||A \Rightarrow B||_T = 1$.

Remark 17: (1) By definition, T is $\mathcal{D}$-complete in data table $\langle X,Y,I\rangle$ if each attribute implication follows from T iff it is $\mathcal{D}$-true

in $\langle X,Y,I\rangle$. In other words, a $\mathcal{D}$-complete set describes all attribute implications, which are $\mathcal{D}$-true in data, via semantic entailment.

(2) If $\mathcal{D}$ is equivalent to $col(0)$, then T is $\mathcal{D}$-complete in $\langle X,Y,I\rangle$ if and only if T is complete in $\langle X,Y,I\rangle$ in the usual sense [6].

In Definition 16, we defined $\mathcal{D}$-completeness using a semantic entailment from a set of attribute implications. One might as well define it in terms of syntactic entailment because reasoning with attribute implications is syntactico-semantically complete. In more detail, $A \Rightarrow B$ semantically follows from T iff $A \Rightarrow B$ is derivable from T using the so-called Armstrong inference rules [2], [6], [9]. Hence, we have the following

Theorem 18: Let T be $\mathcal{D}$-complete in $\langle X,Y,I\rangle$. Then the following assertions are equivalent:

(i) $A \Rightarrow B$ is $\mathcal{D}$-true in $\langle X,Y,I\rangle$,
(ii) $A \Rightarrow B$ semantically follows from T,
(iii) $A \Rightarrow B$ is derivable from T using Armstrong inference rules *[2], [9]*.

The following assertion shows an important characterization of $\mathcal{D}$-completeness: a set T is $\mathcal{D}$-complete in data iff the models of T are exactly the $\mathcal{D}$-intents.

Theorem 19: T is $\mathcal{D}$-complete in data table $\langle X,Y,I\rangle$ if and only if $\mathrm{Mod}(T) = \mathrm{Int}_{\mathcal{D}}(X,Y,I)$.

C. Non-redundant bases of approximately valid implications

In this section we describe particular $\mathcal{D}$-complete sets of attribute implications which are minimal.

Definition 20: A set T of attribute implications over Y is called a non-redundant $\mathcal{D}$-basis of $\langle X,Y,I\rangle$, if T is $\mathcal{D}$-complete in $\langle X,Y,I\rangle$ and no proper subset of T is $\mathcal{D}$-complete in $\langle X,Y,I\rangle$.

In order to describe particular non-redundant $\mathcal{D}$-bases, we introduce a notion of a pseudo $\mathcal{D}$-intent as follows:

Definition 21: Let $\langle X,Y,I\rangle$ be a data table, $\mathcal{D}$ be a column-like property. $P \subseteq Y$ is called a pseudo $\mathcal{D}$-intent of $\langle X,Y,I\rangle$ if $P \neq P^{\downarrow\uparrow}$ and, for each pseudo $\mathcal{D}$-intent Q of $\langle X,Y,I\rangle$ such that $Q \subset P$, we have $Q^{\downarrow\uparrow} \subseteq P$. The collection of all pseudo $\mathcal{D}$-intents of $\langle X,Y,I\rangle$ will be denoted by $\mathcal{P}$.

Remark 22: (1) Described verbally, P is a pseudo $\mathcal{D}$-intent iff P is not a $\mathcal{D}$-intent and each $\mathcal{D}$-intent $Q^{\downarrow\uparrow}$, which is a closure of a pseudo $\mathcal{D}$-intent $Q \subset P$, is a subset of P.

(2) Since Y is supposed to be finite, given $\langle X,Y,I\rangle$ and $\mathcal{D}$, Definition 21 recursively defines a unique system $\mathcal{P}$ of all pseudo $\mathcal{D}$-intents of $\langle X,Y,I\rangle$.

(3) The notion of a pseudo $\mathcal{D}$-intent is an analogy of the notion of a pseudo intent, see [7], [6]. However, one cannot directly adopt results from [7], [6] in case of dense rectangles because $^{\downarrow}$ and $^{\uparrow}$ no longer form a Galois connection in the classical sense. On the other hand, we show that, with appropriate modifications, all important properties of pseudo intents are preserved in case of our pseudo $\mathcal{D}$-intents and arbitrary column-like property $\mathcal{D}$.

The following assertion says that the collection $\mathcal{P}$ of all pseudo $\mathcal{D}$-intents can be used to obtain a non-redundant $\mathcal{D}$-basis.

Theorem 23: Let $\langle X,Y,I\rangle$ be a data table, $\mathcal{D}$ be a column-like property, $T = \{P \Rightarrow P^{\downarrow\uparrow} \,|\, P \in \mathcal{P}\}$. Then

(i) T is a non-redundant $\mathcal{D}$-basis of $\langle X,Y,I\rangle$.
(ii) If T' is $\mathcal{D}$-complete in $\langle X,Y,I\rangle$, then $|T| \leq |T'|$.

We now focus on computing non-redundant $\mathcal{D}$-bases given by collections of pseudo $\mathcal{D}$-intents. Note first that due to Theorem 23 (ii), $T = \{P \Rightarrow P^{\downarrow\uparrow} \,|\, P \in \mathcal{P}\}$ is a minimal non-redundant $\mathcal{D}$-basis of $\langle X,Y,I\rangle$. That is, there is no $\mathcal{D}$-complete set which has strictly lesser number of attribute implications than T has. Since $^{\downarrow\uparrow}$ is a closure operator, we can use the ideas from [6] to compute pseudo $\mathcal{D}$-intents (and thus the desired set T) as fixed points of a special closure operator. Given $M \subseteq Y$ and a set T of attribute implications over Y, we define a sequence $M^{\mathcal{D}}_{T,0} \subseteq M^{\mathcal{D}}_{T,1} \subseteq \cdots$ of subsets of Y by

$$M^{\mathcal{D}}_{T,0} = M, \quad M^{\mathcal{D}}_{T,i+1} = M^{\mathcal{D}}_{T,i} \cup \bigcup\{B \,|\, A \Rightarrow B \in T \text{ and } A \subset M^{\mathcal{D}}_{T,i}\}.$$

Furthermore, we define an operator $C^{\mathcal{D}}_T : 2^Y \to 2^Y$ by

$$C^{\mathcal{D}}_T(M) = \bigcup_{i=0}^{\infty} M^{\mathcal{D}}_{T,i}.$$

The following assertion shows that $C^{\mathcal{D}}_T$ can be used to obtain pseudo $\mathcal{D}$-intents.

Theorem 24: Let $\langle X,Y,I\rangle$ be a data table, $\mathcal{D}$ be a column-like property, $T = \{P \Rightarrow P^{\downarrow\uparrow} \,|\, P \in \mathcal{P}\}$. Then

$$\mathcal{P} = \{M \subseteq Y \,|\, M = C^{\mathcal{D}}_T(M) \text{ and } M \neq M^{\downarrow\uparrow}\}. \tag{4}$$

From Theorem 24 it follows that pseudo $\mathcal{D}$-intents are particular fixed points of $C^{\mathcal{D}}_T$. For computation of all fixed points of $C^{\mathcal{D}}_T$ we can use NextClosure [6], see Section IV. In addition to that, each closure $C^{\mathcal{D}}_T(M)$ of M can be computed using a modification of LinClosure (see [9] for details) which has linear complexity with respect to the size of T. Combining these two algorithms together with Theorem 23 and (4), we get the following algorithm for computing of non-redundant $\mathcal{D}$-bases:

Algorithm 25: Denote by $\textsc{NextClosure}(M, C^{\mathcal{D}}_T)$ a subset of Y which is the lexically smallest fixed point of $C^{\mathcal{D}}_T$ strictly greater than $M \subseteq Y$, see [6]. The algorithm goes as follows:

Input: data table $\langle X,Y,I\rangle$, column-like property $\mathcal{D}$
Output: non-redundant $\mathcal{D}$-basis T of $\langle X,Y,I\rangle$

$M := \emptyset$, $T := \emptyset$
if $M \neq M^{\downarrow\uparrow}$: *add* $M \Rightarrow M^{\downarrow\uparrow}$ *to* T
while $M \neq Y$:
  $M := \textsc{NextClosure}(M, C^{\mathcal{D}}_T)$
  if $M \neq M^{\downarrow\uparrow}$: *add* $M \Rightarrow M^{\downarrow\uparrow}$ *to* T

Remark 26: Correctness of Algorithm 25 follows from Theorem 24 and Theorem 23. The only place we need to take care about is that during the computation, we use operator $C^{\mathcal{D}}_T$ where $T = \{P \Rightarrow P^{\downarrow\uparrow} \,|\, P \in \mathcal{P}'\}$, however, $\mathcal{P}'$ may not contain all pseudo $\mathcal{D}$-intents of $\langle X,Y,I\rangle$, cf. Theorem 24. On the other hand, NextClosure generates all fixed points in lexical order [6] which is a total order extending the strict subsethood relation $\subset$, i.e. in each computational step, we already have computed all (strictly smaller) pseudo $\mathcal{D}$-intents which are necessary to determine the lexically-next one.

VI. Illustrative examples and further issues

In this section we present illustrative examples and results of experiments. For brevity, we adopt the following convention for denoting column-like properties. Given a data table $\langle X,Y,I\rangle$, we assume that $Y = \{y_1,\ldots,y_n\}$ is ordered by $y_1 < y_2 < \cdots < y_n$. Then, each column-like property $\mathcal{D}$ for $\langle X,Y,I\rangle$ is uniquely given by a sequence $l_{y_1}, l_{y_2}, \ldots, l_{y_n}$ of nonnegative integers, meaning that $\mathcal{D}$ is equivalent to $col(\mathbf{l})$, where $\mathbf{l} = \{l_{y_1},\ldots,l_{y_n}\}$, see above. If there is no danger of confusion, we write $l_{y_1} l_{y_2} \cdots l_{y_n}$ instead of $l_{y_1}, l_{y_2}, \ldots, l_{y_n}$ and we denote $\mathcal{D}$ by $col(l_{y_1} l_{y_2} \cdots l_{y_n})$. For instance, if $Y = \{y_1,\ldots,y_4\}$, then $col(0101)$ represents column-like property $\mathcal{D}$ which allows one blank in columns y_2 and y_4 and disallow any blanks elsewhere.

Example 27: Consider a data table $\langle X,Y,I\rangle$ presented in Fig. 1. The ordinary concept lattice $\mathcal{B}(X,Y,I)$ induced by $\langle X,Y,I\rangle$ has 19 formal concepts (maximal rectangles), which are denoted by $C_0,\ldots,C_{18}$:
$C_0 = \langle X, \{a\}\rangle$, $C_1 = \langle \{1,2,3,4\}, \{a,g\}\rangle$,
$C_2 = \langle \{2,3,4\}, \{a,g,h\}\rangle$, $C_3 = \langle \{5,6,7,8\}, \{a,d\}\rangle$,
$C_4 = \langle \{5,6,8\}, \{a,d,f\}\rangle$, $C_5 = \langle \{3,4,6,7,8\}, \{a,c\}\rangle$,
$C_6 = \langle \{3,4\}, \{a,c,g,h\}\rangle$, $C_7 = \langle \{4\}, \{a,c,g,h,i\}\rangle$,
$C_8 = \langle \{6,7,8\}, \{a,c,d\}\rangle$, $C_9 = \langle \{6,8\}, \{a,c,d,f\}\rangle$,

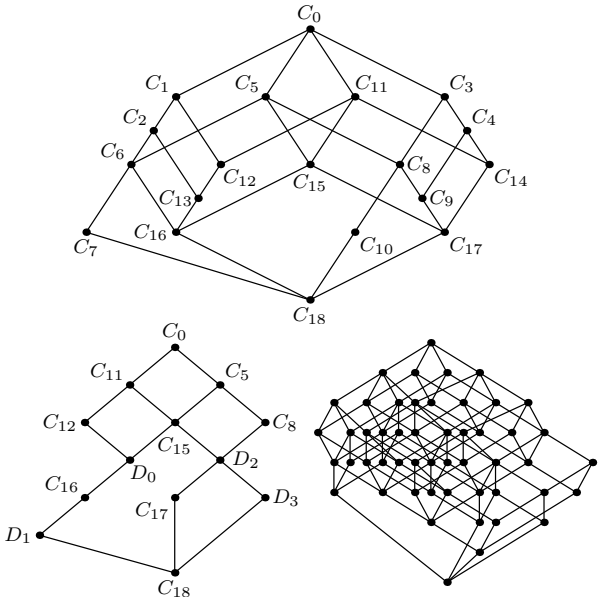

Fig. 2. Hierarchies of $\mathcal{D}$-concepts

$C_{10} = \langle\{\{7\},\{a,c,d,e\}\rangle$, $C_{11} = \langle\{\{1,2,3,5,6\},\{a,b\}\rangle$,
$C_{12} = \langle\{\{1,2,3\},\{a,b,g\}\rangle$, $C_{13} = \langle\{\{2,3\},\{a,b,g,h\}\rangle$,
$C_{14} = \langle\{\{5,6\},\{a,b,d,f\}\rangle$, $C_{15} = \langle\{\{3,6\},\{a,b,c\}\rangle$,
$C_{16} = \langle\{\{3\},\{a,b,c,g,h\}\rangle$, $C_{17} = \langle\{\{6\},\{a,b,c,d,f\}\rangle$, $C_{18} = \langle\{\},Y\rangle$.
Fig. 2 (left) depicts the concept lattice $\mathcal{B}(X,Y,I)$ [6], i.e. the partially ordered hierarchy of formal concepts $C_0,\ldots,C_{18}$. As mentioned above if we take $\mathcal{D}$ to be $col(0)$ (no blanks allowed), the $\mathcal{D}$-concept lattice $\mathcal{B}_{col(0)}(X,Y,I)$ is "the same" as the ordinary concept lattice $\mathcal{B}(X,Y,I)$. In more detail, we have

$$\mathcal{B}_{col(0)}(X,Y,I) = \{\langle\{A\},B\rangle \,|\, \langle A,B\rangle \in \mathcal{B}(X,Y,I)\}.$$

On the other hand, by various choices of $\mathcal{D}$ we can get simplified or extended sets (hierarchies) of $\mathcal{D}$-concepts. For instance, if $\mathcal{D}$ is $col(011000000)$, i.e. if we allow one blank in columns corresponding to attributes "lives in water" and "lives on land", we get the following set of $\mathcal{D}$-concepts:
$C_0 = \langle\{X\},\{a\}\rangle$, $C_5 = \langle\{\{3,4,5,6,7,8\},\{2,3,4,6,7,8\},$
$\{1,3,4,6,7,8\}\},\{a,c\}\rangle$, $C_8 = \langle\{\{5,6,7,8\}\},\{a,c,d\}\rangle$,
$C_{11} = \langle\{\{1,2,3,5,6,8\},\{1,2,3,5,6,7\},\ \{1,2,3,4,5,6\}\},\{a,b\}\rangle$,
$C_{12} = \langle\{\{1,2,3,4\}\},\{a,b,g\}\rangle$,
$C_{15} = \langle\{\{3,5,6,8\},\{3,5,6,7\},\{3,4,5,6\},\{2,3,6,8\},$
$\{2,3,6,7\},\{2,3,4,6\},\{1,3,6,8\},\{1,3,6,7\},$
$\{1,3,4,6\}\},\{a,b,c\}\rangle$,
$D_0 = \langle\{\{2,3,4\},\{1,3,4\}\},\{a,b,c,g\}\rangle$,
$C_{16} = \langle\{\{2,3,4\}\},\{a,b,c,g,h\}\rangle$, $D_1 = \langle\{\{4\}\},\{a,b,c,g,h,i\}\rangle$,
$D_2 = \langle\{\{5,6,8\},\{5,6,7\}\},\{a,b,c,d\}\rangle$,
$C_{17} = \langle\{\{5,6,8\}\},\{a,b,c,d,f\}\rangle$,
$D_3 = \langle\{\{7\}\},\{a,b,c,d,e\}\rangle$, $C_{18} = \langle\{\{\}\},Y\rangle$.
The hierarchy of $\mathcal{D}$-concepts is depicted in Fig. 2 (middle). Observe that $\mathcal{D}$-concepts denoted by C_i have the same intents as the corresponding $col(0)$-concepts. Extents of the corresponding $\mathcal{D}$-concepts and $col(0)$-concepts do not coincide in general because we use two different column-like properties. $\mathcal{B}_{\mathcal{D}}(X,Y,I)$ is smaller than $\mathcal{B}(X,Y,I)$. Thus, $\mathcal{B}_{\mathcal{D}}(X,Y,I)$ can be seen as a simplified view on $\mathcal{B}(X,Y,I)$ in which we allow $\mathcal{D}$-concepts which are not represented by rectangles full of 1's. $\mathcal{B}_{\mathcal{D}}(X,Y,I)$ contains four $\mathcal{D}$-concepts which do not have their analogies in $\mathcal{B}(X,Y,I)$: D_0 ($\mathcal{D}$-concept of organisms living in water and on land which can move around), D_1 ($\mathcal{D}$-concept of a dog), D_2 ($\mathcal{D}$-concept of organisms living in water and on land which need chlorophyll to produce food), D_3 ($\mathcal{D}$-concept of a bean). Extents of $\mathcal{D}$-concepts D_1 (a dog) and D_3 (a bean) are contained in $\mathcal{B}(X,Y,I)$ (see C_7 and C_{10}), however, intents of concepts C_7 and C_{10} differ from intents of D_1 and D_3.

Let us mention that other choices of $\mathcal{D}$ may extend the structure. As an example, consider column-like property $col(1)$ (one blank in each column). In this particular case, we have 51 $\mathcal{D}$-concepts, see Fig. 2 (right).

The next example deals with non-redundant $\mathcal{D}$-bases of implications. We will use non-redundant bases ($\mathcal{D}$-bases) which have shorter description than bases described by Theorem 23. Instead of taking a set $T = \{P \Rightarrow P^{\downarrow\uparrow} \,|\, P \in \mathcal{P}\}$ of attribute implications, where $\mathcal{P}$ is a collection of pseudo intents ($\mathcal{D}$-intents), we will use sets of the form $T = \{P \Rightarrow P^{\circ} \,|\, P \in \mathcal{P}\}$, where $P^{\circ} = \{y \in Y \,|\, y \in P^{\downarrow\uparrow} \text{ and } y \notin P\}$. That is, the attribute set P° results from $P^{\downarrow\uparrow}$ by removing attributes which appear in P. Obviously, if $\mathcal{P}$ is a collection of pseudo intents ($\mathcal{D}$-intents) then T is a (minimal) non-redundant basis ($\mathcal{D}$-basis).

Example 28: Consider again a data table $\langle X,Y,I\rangle$ from Fig. 1. The non-redundant basis given by pseudo intents (i.e., pseudo $col(0)$-intents) is the following:

$$\begin{aligned} T_0 = \{&\{a,b,c,g,h,i\} \Rightarrow \{d,e,f\}, \{a,b,d\} \Rightarrow \{f\},\\ &\{a,c,d,e,f\} \Rightarrow \{b,g,h,i\}, \{a,c,g\} \Rightarrow \{h\},\\ &\{a,d,g\} \Rightarrow \{b,c,e,f,h,i\}, \{a,e\} \Rightarrow \{c,d\},\\ &\{a,f\} \Rightarrow \{d\}, \{a,h\} \Rightarrow \{g\}, \{a,i\} \Rightarrow \{c,g,h\}, \{\} \Rightarrow \{a\}\}. \end{aligned}$$

In case of $\mathcal{D}$ being $col(1)$, the non-redundant $\mathcal{D}$-basis T_1 has only 7 implications (T_0 consists of 10 implications):

$$\begin{aligned} T_1 = \{&\{a,b,c,d,g,h,i\} \Rightarrow \{f\}, \{a,c,d,e,f,g\} \Rightarrow \{b,h,i\},\\ &\{a,e\} \Rightarrow \{c,d\}, \{a,f\} \Rightarrow \{d\}, \{a,h\} \Rightarrow \{g\}, \{a,i\} \Rightarrow \{c,g,h\},\\ &\{\} \Rightarrow \{a\}\}. \end{aligned}$$

Observe that all attribute implications from T_1 except for $\{a,b,c,d,g,h,i\} \Rightarrow \{f\}$ and $\{a,c,d,e,f,g\} \Rightarrow \{b,h,i\}$ are contained in T_0. Nevertheless, these two implications are true in the usual sense in the data table. Thus, each intent ($col(0)$-intent) is a model of T_1. On the other hand, Theorem 19 and Theorem 23 say that T_1 is not complete ($col(0)$-complete) in $\langle X,Y,I\rangle$ because $|T_1| < |T_0|$, i.e. some models of T_1 are not intents ($col(0)$-intents). Of course, T_1 is $col(1)$-complete in $\langle X,Y,I\rangle$ because it is a non-redundant $col(1)$-basis.

By other choices of column-like properties, we can get even smaller non-redundant $\mathcal{D}$-bases. For example, if $\mathcal{D}$ is $col(011101110)$, we get the following $\mathcal{D}$-basis:

$$\begin{aligned} T_2 = \{&\{a,e\} \Rightarrow \{b,c,d,f,g,h\}, \{a,f\} \Rightarrow \{d\}, \{a,h\} \Rightarrow \{g\},\\ &\{a,i\} \Rightarrow \{b,c,d,f,g,h\}, \{\} \Rightarrow \{a\}\}. \end{aligned}$$

Unlike T_1, T_2 contains implications which are not true in $\langle X,Y,I\rangle$ in the usual sense. For instance, $\{a,i\} \Rightarrow \{b,c,d,f,g,h\}$ is not true in $\langle X,Y,I\rangle$.

Example 29: Fig. 3 shows an experimentally assessed dependence of the number of formal $col(1)$-concepts of $\mathcal{B}_{col(1)}(X,Y,I)$ (the two graphs left) and the number of implications in the minimal non-redundant bases (the two graphs right) on the density of input data tables (proportion of $\times$'s). Experiments have shown that in dense data tables, $\mathcal{B}_{col(1)}(X,Y,I)$ is usually smaller than $\mathcal{B}_{col(0)}(X,Y,I)$. On the other hand, in data tables with average density the situation is the opposite. The first graph depicts the situation for data tables with 5 attributes, the seconds graph depicts the situation for data tables with 10 attributes. Solid line in a graph represents average number

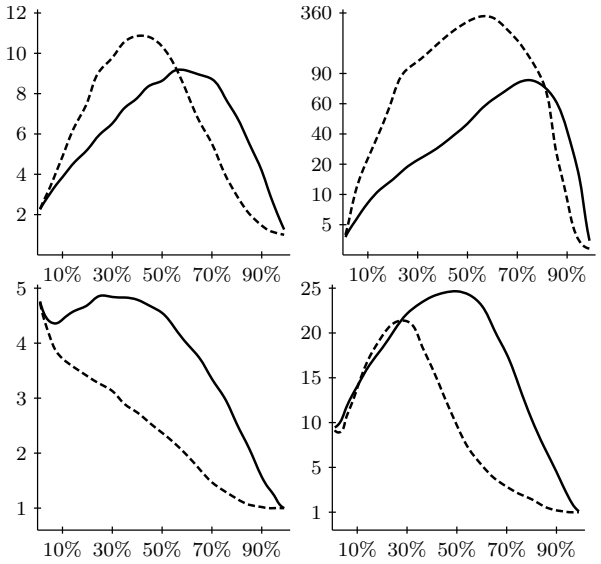

Fig. 3. The first and the second graph: Average number of $col(1)$-concepts. The third and the fourth graph: Average number of minimal non-redundant bases.

of concepts ($col(0)$-concepts); dashed line represents average number of $col(1)$-concepts.

The third and the fourth graphs show the corresponding behavior of the number of implications of minimal non-redundant bases. Surprisingly, the experiments have shown that sizes of minimal $col(1)$-bases are usually smaller than sizes of minimal $col(0)$-bases and that this observation more or less does not depend on the density of a data table. That is, with our approach to approximate validity of attribute implications we get a smaller number of implications (which are presented to the user). This is a beneficial feature. Again, the first graph depicts the situation for data tables with 5 attributes, the seconds graph depicts the situation for data tables with 10 attributes; solid line in a graph represents average number of implications in $\mathcal{D}$-bases (approximate validity), dashed line represents average number of implications in ordinary bases (exact validity).

Future research needs to focus on the following topics: connections to association rules (the concept of $\mathcal{D}$-truth is an approach to approximate validity which is different to the one based on confidence used in association rules, a comparison of these two approaches to approximate validity is an issue to be studied); experiments with large datasets; algorithms for computing of closures $^{\downarrow\uparrow}$, which are related to computing of non-redundant $\mathcal{D}$-bases; relationships between different choices of density property $\mathcal{D}$ and a study of further types of density properties.

Acknowledgment

Supported by grant No. 1ET101370417 of GA AV ČR, by grant No. 201/05/0079 of the Czech Science Foundation, and by institutional support, research plan MSM 6198959214.

References

[1] Agrawal R., Imielinski T., Swami A. N.: Mining association rules between sets of items in large databases. *Proc. ACM Int. Conf. of Management of Data*, pp. 207–216, 1993.

[2] Armstrong W. W.: Dependency structures in data base relationships. *IFIP Congress*, Geneva, Switzerland, 1974, pp. 580–583.

[3] Belohlavek R., Vychodil V.: Dense rectangles in object-attribute data. In: Proc. IEEE GrC 2006, 2006 IEEE International Conference on Granular Computing, Atlanta, GA, May 10–12, 2006, pp. 586–591.

[4] Burgmann C., Wille R.: The basic theorem on preconcept lattices. In: Missaoui R., Schmid J. (Eds.): ICFCA 2006, *Lecture Notes in Artificial Intelligence* **3874**, pp. 80–88, Springer-Verlag, Berlin/Heidelberg, 2006.

[5] Carpineto C., Romano G.: *Concept Data Analysis. Theory and Applications.* J. Wiley, 2004.

[6] Ganter B., Wille R.: *Formal Concept Analysis. Mathematical Foundations.* Springer, Berlin, 1999.

[7] Guigues J.-L., Duquenne V.: Familles minimales d'implications informatives resultant d'un tableau de données binaires. *Math. Sci. Humaines* **95**(1986), 5–18.

[8] Johnson D. S., Yannakakis M., Papadimitrou C. H.: On generating all maximal independent sets. *Inf. Processing Letters* **15**(1988), 129–133.

[9] Maier D.: *The Theory of Relational Databases.* Computer Science Press, Rockville, 1983.

[10] Norris E. M.: An algorithm for computing the maximal rectangles of a binary relation. *Journal of ACM* 21:356–266, 1974.

[11] Ore O.: Galois connections. *Trans. Amer. Math. Soc.* 55:493–513, 1944.

[12] Wille R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In: Rival I.: *Ordered Sets.* Reidel, Dordrecht, Boston, 1982, 445—470.

[13] Zhang C., Zhang S.: *Association Rule Mining. Models and Algorithms.* Springer, Berlin, 2002.