Association Rules and Mechanizing Hypotheses Formation

Jan Rauch

Faculty of Informatics and Statistics, University of Economics, W. Churchill Sq. 4, 13067 Prague, Czech Republic, e-mail: rauch@vse.cz

1 Introduction

This paper has three goals. The first one is to remember a 35 years old GUHA method of mechanizing hypotheses formation. The development of GUHA method started in 1966. A mathematical theory related to the question "*Can computers formulate and justify scientific hypotheses?*" was developed, see monograph [3]. A suggestion of logic of discovery is described in this book. The logic of discovery has two parts, logic of induction and logic of suggestion, see section 2. GUHA is a method of the logic of suggestion. It was several times implemented, see e.g. [5], [7], [13].

Special logical calculi were defined and studied in connection with the logic of induction. Various theoretical results were achieved in last years, see e.g. [9], [10], [11]. These results can be understood as logic of (generalised) association rules. The second and main goal of this paper is to outline main achievements concerning the logic of association rules, see section 3.

The third goal is to draw attention to implementation aspects of GUHA methods. GUHA was several times implemented and many times applied. A description of a concrete application is out of range of this paper. Examples of applications are e.g. in [6] (processing of sociological data), [15] (chemical engineering), [5] (medicine), and [8] (truck reliability data analysis).

GUHA mines for generalised association rules described in section 3. Let us emphasize that association rules based on statistical hypotheses tests are also included [14]. GUHA does not use the A-priori algorithm [1]. Implementation of GUHA method is based on representation of analysed data by suitable strings of bits [7]. Some deep theoretical results are used e.g. deduction rules concerning association rules (see subsection 3.4) and tables of critical frequencies (see subsection 3.5). The recent implementation of GUHA method is the procedure 4ft–Miner. Its main features are described in section 4.

2 Logic of Discovery

The book [3] is inspired by the following questions:

- Can computers formulate and verify scientific hypotheses?
- Can computers in a rational way analyse empirical data and produce a reasonable reflection of the observed empirical world? Can it be done using mathematical logic and statistics?

The theory developed in [3] is based on the following scheme of inductive inference:

$$\frac{\text{theoretical assumptions, observational statement}}{\text{theoretical statement}}\ .$$

It means that if we accept theoretical assumptions and verify a particular statement concerning observed data, we accept a conclusion - a theoretical statement. It is crucial that an intelligent statement concerning observed data leads to theoretical conclusions, not the data themselves. The following questions are formulated in [3]:

(L0) In what languages does one formulate observational and theoretical statements? (What is the syntax and semantics of these languages? What is their relation to the classical first order predicate calculus?)
(L1) What are rational inductive inference rules bridging the gap between observational and theoretical sentences? (What does it mean that a theoretical statement is justified?)
(L2) Are there rational methods for deciding whether a theoretical statement is justified (on the basis of given theoretical assumptions and observational statements)?
(L3) What are the conditions for a theoretical statement or a set of theoretical statements to be of interest (importance) with respect to the task of scientific cognition?
(L4) Are there methods for suggesting such a set of statements, which is as interesting, as possible?

The answers to questions (LO) - (L2) constitute the logic of induction, the answers to questions (L3) and (L4) constitute the logic of suggestion. Logic of discovery is constituted by answers to questions (LO) -(L4).

The following notions are defined and used in [3] to create logic of discovery.

- **Semantic system** $\mathtt{S} = \langle\ \mathtt{Sent}, \mathtt{M}, \mathtt{V}, \mathtt{Val}\ \rangle$ is given by a non–empty set $\mathtt{Sent}$ of *sentences*, a non–empty set $\mathtt{M}$ of *models*, non–empty set $\mathtt{V}$ of *abstract values* and by an evaluation function $\mathtt{Val}\colon (\mathtt{Sent} \times \mathtt{M}) \to \mathtt{V}$. If it is $\varphi \in \mathtt{Sent}$ and $M \in \mathtt{M}$ then $\mathtt{Val}(\varphi, M)$ is the value of φ in M (see paragraph 1.2.5 in [3]).
- **Semantic system** $\mathtt{S} = \langle\ \mathtt{Sent},\ \mathtt{M},\ \mathtt{V},\ \mathtt{Val}\ \rangle$ is **observational semantic system** if $\mathtt{Sent}$, $\mathtt{M}$ and $\mathtt{V}$ are recursive sets and $\mathtt{Val}$ is a partially recursive set (see paragraph 1.2.18 in [3]).
- Let us suppose that we have a recursive set $\mathtt{Sent}$ such that items of $\mathtt{Sent}$ are called sentences and

  $$\mathtt{Sent} = \mathtt{Sent}_0 \cup \mathtt{Sent}_\mathtt{T}$$

  where $\mathtt{Sent}_0$ and $\mathtt{Sent}_\mathtt{T}$ are recursive sets not necessarily disjoint. Items of $\mathtt{Sent}_0$ are called observational sentences, items of $\mathtt{Sent}_\mathtt{T}$ are called theoretical sentences. Inference rule $\mathtt{I}$ on $\mathtt{Sent}$ is called **inductive** if it is of the form

  $$\frac{\Gamma, \Delta}{\Psi}\ .$$

Here Γ is a finite set of theoretical sentences, Δ is a non–empty finite set of observational sentences and Ψ is a theoretical sentence (see paragraph 1.2.23 in [3]).

The answers to questions **(L0)** – **(L4)** are based on considerations about observational semantic system $\mathtt{S_0} = \langle \mathtt{Sent_0}, \mathtt{M_0}, \mathtt{V_0}, \mathtt{Val_0} \rangle$ and about theoretical semantic system $\mathtt{S_T} = \langle \mathtt{Sent_T}, \mathtt{M_T}, \mathtt{V_T}, \mathtt{Val_T} \rangle$. Rationality criteria of inductive inference rules are based on statistical inference rules corresponding to statistical hypotheses tests. A more detailed description of this approach is out of range of this paper.

An observational semantic system sentences of which correspond to association rules is outlined in the next section.

3 Logic of Association Rules

Various observational calculi were defined and studied in [3]. Observational calculus is a logical calculus formulae of which can be understood as the observational semantic system. We are going to outline the observational calculus formulae of which correspond to association rules and to argue for usefulness of this calculus.

The association rule is here understood as an expression $\varphi \approx \psi$ where φ and ψ are derived Boolean attributes. These attributes correspond to columns of analysed data matrix (database relation). Symbol $\approx$ denotes a condition concerning four-fold contingency table of φ and ψ. The intuitive meaning of association rule $\varphi \approx \psi$ is that Boolean attributes φ and ψ are associated in the way corresponding to the condition $\approx$. Examples of such conditions are in subsection 3.1.

There are various classes of association rules, e.g. implication, double implication or equivalency, see subsection 3.2. Each particular class contains both association rules verified using simple relations concerning four-fold contingency table and association rules corresponding to statistical hypotheses tests.

Formal definition of logical calculus formulae of which correspond to association rules is outlined in subsection 3.3. There are practically useful results concerning logic of association rules. The most important results concerns deduction rules in the logic of association rules, see subsection 3.4 and tables of critical frequencies, see subsection 3.5. Deduction rules and tables of critical frequencies are used in the procedure 4ft-Miner mining for association rules, see section 4.

3.1 Data Matrix, Boolean attributes and Four - fold Table

The association rule concerns analysed data matrix (database relation). An example of data matrix $\mathcal{M}$ is in Tab. 1.

object	A_1	A_2	$\ldots$	A_K	φ	ψ
o_1	$a_{1,1}$	$a_{1,2}$	$\ldots$	$a_{1,K}$	1	0
$\vdots$	$\vdots$	$\vdots$	$\vdots\vdots\vdots$	$\vdots$	$\vdots$	$\vdots$
o_n	$a_{n,1}$	$a_{n,2}$	$\ldots$	$a_{n,K}$	0	1

Tab. 1 - Data matrix $\mathcal{M}$

Rows of data matrix $\mathcal{M}$ correspond to observed objects, columns of data matrix correspond to attributes $A_1, \ldots, A_K$. We suppose that $a_{1,1}$ is the value of attribute A_1 for object o_1, $a_{n,K}$ is the value of attribute A_K for object o_n, etc.

Association rule is an expression

$$\varphi \approx \psi$$

where φ and ψ are derived Boolean attributes. An example is

$$A_1(6,7) \approx A_2(3,4) \wedge A_4(2,5).$$

It concerns derived Boolean attributes $A_1(6,7)$ and $A_2(3,4) \wedge A_4(2,5)$. Boolean attribute $A_1(6,7)$ is true in row o_1 if there is a value 6 or 7 in the column A_1 in the row o_1 ($a_{1,1} = 6$ or $a_{1,1} = 7$) etc. A {0,1}-column of analysed data matrix $\mathcal{M}$ (a database relation) correspond to each derived Boolean attribute, see Tab. 1.

Symbol $\approx$ used in association rule $\varphi \approx \psi$ is called a *4ft quantifier* [13]. The 4ft quantifier is a special case of a *generalised quantifier* [3]. Symbol $\approx$ denotes a condition concerning four-fold contingency table of φ and ψ, see Tab. 2.

$\mathcal{M}$	ψ	$\neg\psi$
φ	a	b
$\neg\varphi$	c	d

Tab. 2 - 4ft table of $\mathcal{M}$, φ and ψ

Here a is the number of objects (rows of $\mathcal{M}$) satisfying both φ and ψ, b is the number of objects satisfying φ and not satisfying ψ, etc. The quadruple $< a, b, c, d >$ is *4ft-table of* φ and ψ in $\mathcal{M}$, we abbreviate it by $4ft(\mathcal{M},\ \varphi,\ \psi)$.

The intuitive meaning of association rule $\varphi \approx \psi$ is that Boolean attributes φ and ψ are associated (related) in the way corresponding to 4ft quantifier $\approx$. Association rule $\varphi \approx \psi$ **is true in data matrix** $\mathcal{M}$ if and only if the condition corresponding to 4ft quantifier $\approx$ is satisfied in the 4ft-table $4ft(\mathcal{M},\ \varphi,\ \psi)$ of φ and ψ in $\mathcal{M}$.

Various kinds of dependencies of φ and ψ can be expressed by suitable 4ft quantifiers. Some examples follows ([3], [4], [10]):

- 4ft-quantifier $\Rightarrow_{p,s}$ of *founded implication* for $0 < p \leq 1$ and $s > 0$ is defined by the condition
$$\frac{a}{a+b} \geq p \wedge a \geq s\ .$$
It means that at least $100p$ per cent of objects of $\mathcal{M}$ satisfying φ satisfy also ψ and that there are at least s objects of $\mathcal{M}$ satisfying both φ and ψ.

– 4ft-quantifier $\Rightarrow^!_{p,\alpha,s}$ of *lower critical implication* for $0 < p \leq 1$, $0 < \alpha < 0.5$ and $s > 0$ is defined by the condition

$$\sum_{i=a}^{a+b} \binom{a+b}{i} p^i (1-p)^{a+b-i} \leq \alpha \wedge a \geq s \ .$$

It corresponds to a statistical test (on the level α) of the null hypothesis $H_0 : P(\psi|\varphi) \leq p$ against the alternative one $H_1 : P(\psi|\varphi) > p$. Here $P(\psi|\varphi)$ is the conditional probability of the validity of ψ under the condition φ.

– 4ft-quantifier $\Leftrightarrow_{p,s}$ of *founded double implication* for $0 < p \leq 1$ and $s > 0$ is defined by the condition

$$\frac{a}{a+b+c} \geq p \wedge a \geq s \ .$$

It means that at least $100p$ per cent of objects of $\mathcal{M}$ satisfying φ or ψ satisfy both φ and ψ and that there are at least s objects of $\mathcal{M}$ satisfying both φ and ψ.

– 4ft-quantifier $\equiv_{p,s}$ of *p-equivalence* for $0 < p \leq 1$ and $s > 0$ is defined by the condition

$$\frac{a+d}{a+b+c+d} \geq p \wedge a \geq s \ .$$

It means that φ and ψ have the same value (either *true* or *false*) for at least $100p$ per cent of all objects of $\mathcal{M}$ and that there are at least s objects of $\mathcal{M}$ satisfying both φ and ψ.

"Classical" association rule $X \longrightarrow Y$ with confidence C and support S can be of course also expressed by condition concerning 4ft table. Let us suppose that $A_1, \ldots, A_K$ are Boolean attributes corresponding to particular items, $a_{1,1} = 1$ if and only if supermarket basket o_1 contains item A_1, etc., see Tab. 1. Then, association rule $\{A_1, A_2\} \longrightarrow \{A_3, A_4\}$ with confidence C and support S is the same as association rule (as we understand here)

$$A_1 \wedge A_2 \longrightarrow_{C,S} A_3 \wedge A_4$$

with 4ft quantifier $\longrightarrow_{C,S}$. The condition

$$\frac{a}{a+b} \geq C \wedge \frac{a}{a+b+c+d} \geq S$$

concerning 4ft table corresponds to 4ft quantifier $\longrightarrow_{C,S}$.

We write

$$\approx (a,b,c,d) = 1$$

if the condition corresponding to the 4ft quantifier $\approx$ is satisfied on the quadruple $< a,b,c,d >$ of integer non-negative numbers, otherwise we write

$$\approx (a,b,c,d) = 0 \ .$$

3.2 Classes of Association Rules

Various classes of association rules can be defined. An example is the **class of implication quantifiers**. 4ft-quantifiers $\Rightarrow_{p,s}$ of founded implication and $\Rightarrow^!_{p,\alpha,s}$ of lower critical implication are defined in the previous subsection. It is not to hard to prove that the quantifier $\Rightarrow_{p,s}$ has the following property:

$$\text{if } \Rightarrow_{p,s} (a,b,c,d) = 1 \ \wedge \ a' \geq a \ \wedge \ b' \leq b \ \text{ then } \ \Rightarrow_{p,s} (a',b',c',d') = 1$$

for non-negative integers a, b, c, d, a', b', c', d' such that $a+b+c+d > 0$ and $a'+b'+c'+d' > 0$.

The condition $a' \geq a$ and $b' \leq b$ means that the four-fold table $\langle a',b',c',d'\rangle$ is "better from the point of view of implication" than the four-fold table $\langle a,b,c,d\rangle$ (i-better, see [3]).

Let us suppose that $< a,b,c,d >$ is the 4ft-table of φ and ψ in data matrix $\mathcal{M}$ and that $< a',b',c',d' >$ is the 4ft-table of φ and ψ in data matrix $\mathcal{M}'$. Then the sentence "*Four-fold table $\langle a',b',c',d'\rangle$ is better from the point of view of implication than the four-fold table $\langle a,b,c,d\rangle$.*" means: In data matrix $\mathcal{M}'$ are more rows satisfying both φ and ψ than in data matrix $\mathcal{M}$, and in data matrix $\mathcal{M}'$ are fewer rows satisfying φ and not satisfying ψ than in data matrix in $\mathcal{M}$.

Thus it is reasonable to expect that if association rule $\varphi \Rightarrow_{p,s} \psi$ is true in data matrix $\mathcal{M}$, then it is also true in data matrix $\mathcal{M}'$.

This considerations lead to a definition of the class of **implicational quantifiers** [3]: The 4ft quantifier $\Rightarrow^*$ **is implicational** if

$$\Rightarrow^* (a,b,c,d) = 1 \ \wedge \ a' \geq a \ \wedge \ b' \leq b \ \text{ implies } \ \Rightarrow^* (a',b',c',d') = 1$$

for non-negative integers a, b, c, d, a', b', c', d' such that $a+b+c+d > 0$ and $a'+b'+c'+d' > 0$.

The 4ft-quantifier $\Rightarrow_{p,s}$ of founded implication is implicational. It is also proved that the quantifier $\Rightarrow^!_{p,\alpha,s}$ of lower critical implication is implicational [3], [12].

The condition

$$a' \geq a \ \wedge \ b' \leq b$$

used in the definition of implicational quantifiers is called **the truth preservation condition for implicational quantifiers**. We say that **association rule $\varphi \approx \psi$ is implicational** if the 4ft quantifier $\approx$ is implicational.

Further classes of 4ft quantifiers are defined by suitable truth preservation conditions. An example is a class of **double implicational quantifiers** [12] defined by the truth preservation condition

$$a' \geq a \ \wedge b' \ \leq b \ \wedge \ c' \leq c \ \ .$$

A reason for such a definition is in analogy to propositional calculus. If u and v are propositions and both $u \rightarrow v$ and $v \rightarrow u$ are true, then u is equivalent to v (symbol $\rightarrow$ is a propositional connective of implication). Thus we can try to express the relation of equivalence of attributes φ and ψ using "double implicational" 4ft-quantifier $\Leftrightarrow^*$ such that

$$\varphi \Leftrightarrow^* \psi \text{ if and only if } \varphi \Rightarrow^* \psi \wedge \psi \Rightarrow^* \varphi,$$

where $\Rightarrow^*$ is a suitable implicational quantifier.

If we apply the truth preservation condition for implicational quantifier to $\varphi \Rightarrow^* \psi$, we obtain $a' \geq a \;\wedge\; b' \leq b$. If we apply it to $\psi \Rightarrow^* \varphi$, we obtain $a' \geq a \;\wedge\; c' \leq c$, ($c$ is here instead of b, see Tab.1). This leads to the truth preservation condition for double implicational quantifiers $a' \geq a \wedge b' \leq b \wedge c' \leq c$.

Further example is a class of **Σ-double implicational 4ft quantifiers** defined by the truth preservation condition

$$a' \geq a \;\wedge\; b' + c' \leq b + c\ .$$

It means that 4ft quantifier $\Leftrightarrow$ is Σ-double implicational if the condition

$$\Leftrightarrow (a,b,c,d) = 1 \;\wedge\; a' \geq a \;\wedge\; b' + c' \leq b + c \text{ implies } \Leftrightarrow (a',b',c',d') = 1$$

for non-negative integers a, b, c, d, a', b', c', d' such that $a+b+c+d > 0$ and $a'+b'+c'+d' > 0$. It is very easy to prove that each Σ-double implicational 4ft-quantifier is also double implicational.

The example of Σ-double implicational 4ft-quantifier is quantifier $\Leftrightarrow_{p,s}$ of *founded double implication* defined in the previous subsection 3.1. Further example of Σ-double implicational 4ft-quantifier is 4ft-quantifier $\Leftrightarrow^!_{p,\alpha,s}$ of **lower critical double implication** [4]. It is defined for $0 < p \leq 1$, $0 < \alpha < 0.5$ and $s > 0$ by the condition

$$\sum_{i=a}^{a+b+c} \binom{a+b+c}{i} p^i (1-p)^{a+b+c-i} \leq \alpha \;\wedge a \geq s.$$

4ft-quantifier $\Leftrightarrow^!_{p,\alpha,s}$ of **lower critical double implication** corresponds to a statistical test (on the level α) of the null hypothesis $H_0 : P((\varphi \wedge \psi) \mid (\varphi \vee \psi)) \geq p$ against the alternative one $H_1 : P((\varphi \wedge \psi) \mid (\varphi \vee \psi)) > p$. Here $P((\varphi \wedge \psi) \mid (\varphi \vee \psi))$ is the conditional probability of the validity of $\varphi \wedge \psi$ under the condition $\varphi \vee \psi$. It is proved in [12] that $\Leftrightarrow^!_{p,\alpha,s}$ is Σ-double implicational.

Similar situation is for equivalence. If u and v are propositions and both $u \rightarrow v$ and $\neg u \rightarrow \neg v$ are true, then u is equivalent to v. Thus we can try to express the relation of equivalence of attributes φ and ψ using an "equivalence" 4ft-quantifier $\equiv^*$ such that

$$\varphi \equiv^* \psi \text{ if and only if } \varphi \Rightarrow^* \psi \wedge \neg\varphi \Rightarrow^* \neg\psi\ ,$$

where $\Rightarrow^*$ is a suitable implicational quantifier. If we apply the truth preservation condition for implicational quantifier to $\varphi \Rightarrow^* \psi$ we obtain $a' \geq a \;\wedge\; b' \leq b$, if we apply it to $\neg\varphi \Rightarrow^* \neg\psi$, we obtain $d' \geq d \;\wedge\; c' \leq c$, ($c$ is here instead of b and d instead of a, see Tab.1).

This leads to the *truth preservation condition for equivalence quantifiers*:

$$a' \geq a \ \wedge \ b' \leq b \ \wedge \ \ c' \leq c \ \wedge \ d' \geq d$$

and consequently to the definition of **equivalence quantifiers** [10]. Let us emphasize that this condition is used in [3]) in definition of association generalised quantifiers. In [4] are defined several 4ft quantifiers as equivalence quantifiers. The 4ft-quantifier $\equiv_{p,s}$ of p-equivalence mentioned in the previous subsection is one of them.

Several further classes of 4ft quantifiers are defined and studied in [12], e.g., *pure double implicational*, *typical double implicational*, *pure equivalence,* and *typical Σ-equivalence.* Each of these classes of 4ft quantifiers contains useful quantifiers. Some of them are implemented in the procedure 4ft-Miner, see section 4.

Let us emphasise that there are remarkable differences among 4ft quantifiers $\equiv_{p,s}$, $\Leftrightarrow_{p,s}$, $\Rightarrow_{p,s}$ and $\rightarrow_{C,S}$. We use examples with $p = 0.9$, $s = 100$, $C = 0.9$ and $S = 0.1$, with Boolean attributes φ, ψ and with three data matrices $\mathcal{M}_1$, $\mathcal{M}_2$, $\mathcal{M}_3$ with different 4ft tables, see Tab. 3, Tab. 4 and Tab. 5. It is easy to verify that values $Val(\varphi \approx \psi \ , \mathcal{M})$ of particular 4ft association rules are according to Tab. 6. We suppose $\mathcal{M} = \mathcal{M}_1$, $\mathcal{M} = \mathcal{M}_2$ or $\mathcal{M} = \mathcal{M}_3$.

$\mathcal{M}_1$	ψ	$\neg\psi$
φ	100	1
$\neg\varphi$	1	1

Tab. 3 - $4ft(\mathcal{M}_1,\varphi,\psi)$

$\mathcal{M}_2$	ψ	$\neg\psi$
φ	100	1
$\neg\varphi$	100	1

Tab. 4 - $4ft(\mathcal{M}_2,\varphi,\psi)$

$\mathcal{M}_3$	ψ	$\neg\psi$
φ	100	1
$\neg\varphi$	100	10 000

Tab. 5 - $4ft(\mathcal{M}_3,\varphi,\psi)$

data matrix $\mathcal{M}$	$\mathcal{M}_1$	$\mathcal{M}_2$	$\mathcal{M}_3$
$Val(\varphi \equiv_{0.9,100} \psi \ , \mathcal{M})$	*true*	$false$	*true*
$Val(\varphi \Leftrightarrow_{0.9,100} \psi \ , \mathcal{M})$	*true*	$false$	$false$
$Val(\varphi \Rightarrow_{0.9,100} \psi \ , \mathcal{M})$	*true*	*true*	*true*
$Val(\varphi \rightarrow_{0.9,0.1} \psi \ , \mathcal{M})$	*true*	*true*	$false$

Tab. 6 - Values $Val(\varphi \approx \psi \ , \mathcal{M})$ of particular association rules

3.3 Logical Calculus of Association Rules

Let us remember that the theory developed in [3] is based on the following scheme of inductive inference:

$$\frac{\text{theoretical assumptions, observational statement}}{\text{theoretical statement}}$$

It is also emphasized in [3] that it is an intelligent observation of the data (observational statement) that leads to theoretical conclusions, not the data themselves. Five questions (L0)–(L4) are also formulated in [3], see introduction. Let us remember first two of them:

(L0) In what languages does one formulate observational and theoretical statements? (What is the syntax and semantics of these languages? What is their relation to the classical first order predicate calculus?)

(L1) What are rational inductive inference rules bridging the gap between observational and theoretical sentences? (What does it mean that a theoretical statement is justified?)

Various association rules $\varphi \approx \psi$ are defined in the previous subsections, e.g. $\varphi \Rightarrow^{!}_{p,\alpha,s} \psi$. All these association rules are observational statements concerning analysed data matrix that represents the observed data. Observational statement $\varphi \Rightarrow^{!}_{p,\alpha,s} \psi$ leads to the theoretical conslusion that there is a high conditional probability of the validity of ψ under the condition φ. An inductive inference rule is based on the statistical test (on the level α) of the null hypothesis $H_0 : P(\psi|\varphi) \leq p$ against the alternative one $H_1 : P(\psi|\varphi) > p$, see subsection 3.1. Thus, association rule $\varphi \Rightarrow^{!}_{p,\alpha,s} \psi$ is a suitable observational statement.

Further association rules can be also understood as suitable observational statements, the discussion concerning corresponding inductive inference rules is but out of the range of this paper. On the other side, it is undoubted that association rules based on statistical hypotheses tests are related to reasonable inductive inference rules bridging the gap between observational and theoretical sentences. Thus there is a way to answer to question **(L1)**.

We are going to get the answer to the question **(L0)**. It means that we will sketch a definition of a semantic system $\mathtt{S} = \langle\ \mathtt{Sent}, \mathtt{M}, \mathtt{V}, \mathtt{Val}\ \rangle$ (see section 2) corresponding to all association rules concerning data matrices of a given structure. We will proceed in the very informal way, more formal definition is e.g. in [10]

The example of the data matrix is in Tab. 7. It is the same data matrix as that in the Tab. 1 but without the {0,1}-columns corresponding to derived Boolean attributes φ, ψ see Tab. 1.

object	A_1	A_2	$\ldots$	A_K
o_1	$a_{1,1}$	$a_{1,2}$	$\ldots$	$a_{1,K}$
$\vdots$	$\vdots$	$\vdots$	$\vdots\vdots\vdots$	$\vdots$
o_n	$a_{n,1}$	$a_{n,2}$	$\ldots$	$a_{n,K}$

Tab. 7 - Data matrix $\mathcal{M}$

The example of association rule is

$$A_1(6,7) \wedge A_5(1,3) \approx A_2(3,4) \wedge A_4(2,5).$$

It consists of two derived Boolean attributes $A_1(6,7) \wedge A_5(1,3)$ and $A_2(3,4) \wedge A_4(2,5)$ connected by the 4ft quantifier $\approx$. Derived Boolean attributes are built from basic Boolean attributes $A_1(6,7)$, $A_5(1,3)$, $A_2(3,4)$ and $A_4(2,5)$. Basic Boolean attribute $A_1(6,7)$ is true in the row o_i if there is the value 6 or 7 in this row and in the column A_1. Analogously for other basic Boolen attributes.

Association rules concerning data matrix $\mathcal{M}$ in Tab. 7 are built from basic Boolean attributes of the form $A(\alpha)$ where A is one of the columns $A_1, \ldots A_K$. Further, α is a subset of the set of all possible values of the column A. Let us suppose that the set of all possible values for the column A_i is $\{1, \ldots, t_i\}$. Then all basic Boolean attribute derived from attribute A_i are of the form $A_i(j_1, \ldots, j_k)$ where $\{j_1, \ldots, j_k\} \subset \{1, \ldots, t_i\}$. We suppose $i = 1, \ldots, K$.

Thus the set of all possible basic Boolean attributes we can construct for data matrix $\mathcal{M}$ in Tab. 7 is given by the K–tuple $\langle t_1, \ldots t_K \rangle$ of positive integer numbers. We suppose that attribute A_i has values from the set $\{1, \ldots, t_i\}$.

We call this K–tuple by **the type of data matrix** $\mathcal{M}$. We also say that **the type of the column** A_i is t_i. Basic Boolean attributes we can construct for data matrix of type T are called **basic Boolean attributes of type** T.

The semantic system $\mathtt{S} = \langle$ `Sent`, `M`, `V`, `Val` $\rangle$ of the type $T = \langle t_1, \ldots t_K \rangle$ is defined such that

- A non–empty set `Sent` of sentences is the set of all association rules $\varphi \approx \psi$ such that
  - φ and ψ are Boolean expressions made from basic Boolean attributes of type T.
  - Symbol $\approx$ is one of given 4ft quantifiers.

  Examples of association rules are ($\rightarrow$ is a symbol of implication of propositional calculus)

  $$A_1(6,7) \Rightarrow_{0.9,100} \neg A_2(3,4), \quad A_1(6,7) \rightarrow A_5(1,3) \Leftrightarrow_{0.9,100} A_2(3,4) \vee A_4(2,5),$$

  $$A_1(6,7) \wedge A_5(1,3) \Rightarrow^{!}_{0.9,0.05,50} \neg A_2(3,4) \wedge A_4(2,5), .$$
- The non–empty set `M` of models of semantic system of type T is the set of all data matrices $\mathcal{M}$ of type T.
- Non–empty set `V` of abstract values is the set $\{0,1\}$ where 1 means *true* and 0 means *false*.
- Evaluation function `Val`: (`Sent` $\times$ `M`) $\rightarrow$ `V` is defined such that

  $$\mathtt{Val}(\mathcal{M}, \varphi \approx \psi) = 1 \text{ if and only if } \approx (a,b,c,d) = 1$$

  where $< a,b,c,d >$ is 4ft-table $4ft(\mathcal{M}, \varphi, \psi)$ of φ and ψ in $\mathcal{M}$.

In other words a logical calculus is given such that its formulas are association rules concerning data matrices of the given type, formulas are interpreted in these data matrices and the formula is true in a given data matrix if the corresponding association rule is satisfied in this data matrix.

3.4 Deduction Rules

Deduction rule is a relation of the form

$$\frac{\alpha_1, \ldots, \alpha_n}{\beta},$$

where $\alpha_1, \ldots, \alpha_n, \beta$ are formulae. This **deduction rule is correct** if for each model M holds: if $\alpha_1, \ldots, \alpha_n$ are true in M, then also β is true in M. We are interested in correct deduction rules of the the form $\frac{\varphi \approx \psi}{\varphi' \approx \psi'}$, where $\varphi \approx \psi$ and $\varphi' \approx \psi'$ are association rules. Such deduction rules can be used, e.g., in the following ways:

- **To reduce the output of a data mining procedure:** If association rule $\varphi \approx \psi$ is a part of a data mining procedure output (thus it is true in analyzed data matrix) and if $\frac{\varphi \approx \psi}{\varphi' \approx \psi'}$ is a correct deduction rule, then it is not necessary to put $\varphi' \approx \psi'$ into the output. The used deduction rule must be clear enough from the point of view of the user of the data mining procedure. An example of a simple deduction rule is dereduction deduction rule $\frac{\varphi \Rightarrow^* \psi}{\varphi \Rightarrow^* \psi \vee \chi}$, which is correct for each implicational quantifier $\Rightarrow^*$ [3].
- **To decrease the number of actually tested association rules:** If formula $\varphi \approx \psi$ is true in an analyzed model (data matrix) and if $\frac{\varphi \approx \psi}{\varphi' \approx \psi'}$ is a correct deduction rule, then it is not necessary to test $\varphi' \approx \psi'$.

Thus we are interested e.g. if the following deduction rules are correct:

$$\frac{\varphi \Rightarrow^!_{p,\alpha,s} \psi}{\varphi' \Rightarrow^!_{p,\alpha,s} \psi'}, \qquad \frac{\varphi \Rightarrow^!_{p,\alpha,s} \psi}{\varphi' \Rightarrow^!_{p,\alpha,s} \psi'} .$$

This question can be transformed to a question if a given propositional formula is a tautology for important classes of 4ft quantifiers [12], see also [11]. We will show the corresponding theorem. We need several notions.

Definition 1. 1. Let A be the column of type t. Propositional formulae $\alpha_1, \ldots, \alpha_t$ are defined such that for $i = 1, \ldots, t$ it is

$$\alpha_i \;=\; A[i] \Rightarrow \neg A[1] \wedge \ldots \wedge \neg A[i-1] \wedge \neg A[i+1] \wedge \ldots \wedge \neg A[t] \;,$$

where $A[1], \ldots, A[t]$ are propositional variables. (The formula α_i says, that if propositional variable $A[i]$ is true, then all remaining propositional variables $A[1], \ldots,$ A[i-1],A[i+1], $\ldots A[t]$ are false.)

2. Let A be the column of type t. **Propositional formula** $\lambda(A)$ **of uniqueness associated to column** A is defined such that:

$$\lambda(A) \;=\; (A[1] \vee \ldots \vee A[t]) \wedge \alpha_1 \wedge \ldots \wedge \alpha_t \;\; .$$

where $A[1], \ldots, A[t]$ are again propositional variables. Formula $\lambda(A)$ says, that exactly one of propositional variables $A[1], \ldots, A[t]$ are true.

3. Let A be the column of the type t and let $A(h_1, \ldots, h_k)$ be basic Boolean attributes such that $\{h_1, \ldots, h_k\} \subset \{1, \ldots, t\}$. Then **propositional disjunction** $\pi[A(h_1, \ldots, h_k)]$ **associated to basic Boolean attribute** $A(h_1, \ldots, h_k)$ is defined such that

$$\pi[A(h_1, \ldots, h_k)] = A[h_1] \vee \ldots \vee A[h_k]$$

where $A[h_1], \ldots, A[h_k]$ are propositional variables.

4. Let φ be a Boolean attribute. Then **propositional formula** $\pi[\varphi]$ **associated to Boolean attribute** φ is created from φ such that each basic Boolean attribute $A(h_1, \ldots, h_k)$ occuring in φ is replaced by propositional disjunction $\pi[A(h_1, \ldots, h_k)]$ associated to basic Boolean attribute $A(h_1, \ldots, h_k)$.
5. Let φ and ψ be Boolean attributes and let $A_{i_1}, \ldots, A_{i_J}$ be all basic Boolean attributes occurring in φ or in ψ. Then **propositional formula** $\Longrightarrow (\varphi, \psi)$ **associated to Boolean attribute** φ is defined such that
$$\Longrightarrow (\varphi, \psi) \;=\; \lambda(A_{i_1}) \wedge \;\ldots\; \wedge \lambda(A_{i_J}) \;\;\wedge\;\; \pi[\varphi] \;\rightarrow\; \pi[\psi] \;.$$
Here $\lambda(A_{i_i})$ is the propositional formula of uniqueness associated to A_{i_j} for $j = 1, \ldots, J$. Further $\pi[\varphi]$, $\pi[\psi]$ are propositional formulae associated to Boolean attributes φ and ψ respectively.

Example 1. Propositional formula $\pi[A_1(3,4) \wedge A_7(2,6)]$ associated to Boolean attribute $A_1(3,4) \wedge A_7(2,6)$ is
$$\pi([A_1(3,4) \wedge A_7(2,6)]) = (A_1[3] \vee A_1[4]) \wedge (A_7[2] \vee A_7[6])$$
where $A_1[3]$, $A_1[4]$, $A_7[2]$, $A_7[6]$ are propositional variables.

It is easy to prove that $\Longrightarrow (\varphi, \psi)$ is a tautology of propositional calculus if and only if it is true for each row o of each data matrix of the type T (see [12]): *"If φ is true in o than also ψ is true in o"*. This fact leads to the following definition:

Definition 2. Let φ and ψ be Boolean attributes. Then we say that φ **logically follows from** ψ if the formula $\Longrightarrow (\varphi, \psi)$ is a tautology of propositional calculus. Symbolically we write
$$\varphi \;\vdash\; \psi \;.$$

Definition 3. Let $\approx$ be the 4ft quantifier. Then

1. $\approx$ is **a-dependent** if there are non-negative integers a, a', b, c, d such that $\approx (a,b,c,d) \;\neq\; \approx (a',b,c,d)$. Analogously for **b-dependent** .
2. $\approx$ is $(b+c)$**-dependent** if there are non-negative integers a, b, c, d, b', c' such that $b+c \neq b'+c'$ and $\approx (a,b,c,d) \;\neq\; \approx (a,b',c',d)$. Analogously for $(a+d)$**-dependent**.

Definition 4. Interesting quantifiers:

1. An **implicational quantifier** $\Rightarrow^*$ **is interesting** if $\Rightarrow^*$ is both a-dependent and b-dependent, and if $\Rightarrow^* (0,0,c,d) = 0$.
2. A Σ**-double implicational quantifier** $\Leftrightarrow^*$ **is interesting** if $\Leftrightarrow^*$ is both a-dependent and $(b+c)$-dependent, and if $\Leftrightarrow^* (0,0,0,d) = 0$.
3. A Σ**-equivalence quantifier** $\equiv^*$ **is interesting** if $\equiv^*$ is $(a+d)$-dependent and if $\equiv^* (0,b,c,0) = 0$ for $b+c > 0$.

The following theorems proved in [12] show that it is possible to convert the question if $\frac{\varphi \approx \psi}{\varphi' \approx \psi'}$ is the correct deduction rule to question if the suitable propositional formula is the tautology. It is also proved in [12] that all practically important implication, double implication and equivalency 4ft quantifiers are interesting ones.

Theorem 5. *If $\Rightarrow^*$ is an interesting implicational quantifier, then deduction rule $\frac{\varphi \Rightarrow^* \psi}{\varphi' \Rightarrow^* \psi'}$ is correct if and only if at least one of the conditions 1) or 2) are satisfied:*

1. *Both 1.a) and 1.b) are satisfied*
   1.a $\varphi \wedge \psi \ \vdash \ \varphi' \wedge \psi'$.
   1.b $\varphi' \wedge \neg\psi' \ \vdash \ \varphi \wedge \neg\psi$.
2. $\varphi \vdash \neg\psi$.

Theorem 6. *If $\Leftrightarrow^*$ is an interesting Σ-double implicational quantifier, then deduction rule $\frac{\varphi \Leftrightarrow^* \psi}{\varphi' \Leftrightarrow^* \psi'}$ is correct if and only if a) or b) are satisfied:*

a) *Both* $(\varphi \wedge \psi) \vdash \ (\varphi' \wedge \psi')$ *and* $(\varphi' \wedge \neg\psi' \vee \neg\varphi' \wedge \psi') \vdash \ (\varphi \wedge \neg\psi \vee \neg\varphi \wedge \psi)$.
b) $\varphi \vdash \neg\psi$ *or* $\psi \vdash \neg\varphi$

Theorem 7. *If $\equiv^*$ is an interesting Σ-equivalence quantifier, then deduction rule $\frac{\varphi \equiv^* \psi}{\varphi' \equiv^* \psi'}$ is correct if and only if*

$$(\varphi \wedge \psi \vee \neg\varphi \wedge \neg\psi) \ \vdash \ (\varphi' \wedge \psi' \vee \neg\varphi' \wedge \neg\psi') \ .$$

3.5 Tables of critical frequencies

Theorem 8. *Let $\Rightarrow^*$ be an implicational quantifier. Then there is a non-negative and non-decreasing function $Tb_{\Rightarrow^*}$ with value $Tb_{\Rightarrow^*}(a) \in \{0, 1, 2, \ldots\} \cup \{\infty\}$ such that it is*

$$\Rightarrow^* (a, b) = 1 \quad \textit{if and only if } b < Tb_{\Rightarrow^*}(a)$$

for all integers $a \geq 0$ and $b \geq 0$.

Proof. We define $Tb_{\Rightarrow^*}(a) = \min\{e | \Rightarrow^* (a, e) = 0\}$, see the definition of implicational quantifiers. □

We call function $Tb_{\Rightarrow^*}$ a *table of critical frequencies for implicational quantifier* $\Rightarrow^*$. It is used in procedure 4ft-Miner. It is important that the function $Tb_{\Rightarrow^*}$ makes it possible to use a simple test of inequality instead of a rather complex computation. E.g., we can use inequality $b < Tb_{\Rightarrow^!_{p,\alpha,s}}(a)$ instead of condition $\sum_{i=a}^{a+b} \frac{(a+b)!}{i!(a+b-i)!} p^i (1-p)^{a+b-i} \leq \alpha \wedge a \geq s$ for quantifier $\Rightarrow^!_{p,\alpha,s}$ of lower critical implication, see subsection 3.1.

An other form of the table of critical frequencies for implicational quantifier is defined in [3]. Further tables of critical frequencies for Σ-double implicational quantifiers and for Σ-equivalence quantifiers are defined in [11].

4 Procedure 4ft-Miner

Procedure 4ft-Miner is a GUHA procedure that mines for association rules $\varphi \approx \psi$ and for conditional association rules $\varphi \approx \psi/\chi$. Conditional association rules are described in subsection 4.1 Fourteen types of 4ft quantifiers are implemented in procedure 4ft-Miner, see subsection 4.2. Derived Boolean attributes φ, ψ and χ are called *antecedent succedent* and *condition* respectively, see subsection 4.3. Input and output of 4ft-Miner are described in subsection 4.4 and 4.5. Some further features of 4ft-Miner are mentioned in subsection 4.6.

4.1 Conditional Association Rules

The *conditional association rule* is an expression of the form $\varphi \approx \psi/\chi$ where φ, ψ and χ are derived Boolean attributes. The intuitive meaning is that φ and ψ are in relation given by *4ft quantifier* $\approx$ when the condition χ is satisfied.

If $\mathcal{M}$ is data matrix and χ is a Boolean attribute derived from columns of $\mathcal{M}$ then we mean by $\mathcal{M}/\chi$ a data matrix containing exactly all rows of $\mathcal{M}$ satisfying χ. We say that $\varphi \approx \psi/\chi$ is true in data matrix $\mathcal{M}$ if both there is the row of $\mathcal{M}$ satisfying χ and if association rule $\varphi \approx \psi$ is true in data matrix $\mathcal{M}/\chi$.

There is no general relation between the validity of $\varphi \approx \psi$ and the validity of $\varphi \approx \psi/\chi$. Let values of φ, ψ and χ in data matrix $\mathcal{M}_1$ be according to Tab. 8.

objects	ψ	ψ	χ
$o_1 - o_{100}$	1	1	1
o_{101}	1	0	1
$o_{102} - o_{200}$	1	0	0
o_{201}	0	1	1
$o_{202} - o_{301}$	0	0	1

Tab. 8 - Data matrix $\mathcal{M}_1$

Then 4ft tables $4ft(\mathcal{M}_1, \varphi, \psi)$ and $4ft(\mathcal{M}_1/\chi, \varphi, \psi)$ are in Tab. 9 and Tab. 10.

$\mathcal{M}_1$	ψ	$\neg\psi$
φ	100	100
$\neg\varphi$	1	100

Tab. 9 - $4ft(\mathcal{M}_1$,φ,$\psi)$

$\mathcal{M}_1$	ψ	$\neg\psi$
φ	100	1
$\neg\varphi$	1	100

Tab. 10 - $4ft(\mathcal{M}_1/\chi, \varphi, \psi)$

Thus

$$Val(\varphi \Leftrightarrow_{0.9,100} \psi, \mathcal{M}_1) = 0 \;\text{ and }\; Val(\varphi \Leftrightarrow_{0.9,100} \psi/\chi, \mathcal{M}_1) = 1 \;.$$

It is easy to modify $\mathcal{M}_1$ such that

$$Val(\varphi \Leftrightarrow_{0.9,100} \psi, \mathcal{M}_1) = 1 \;\text{ and }\; Val(\varphi \Leftrightarrow_{0.9,100} \psi/\chi, \mathcal{M}_1) = 0 \;.$$

4.2 4ft-quantifiers of 4ft-Miner

Fourteen types of 4ft quantifiers are implemented in procedure 4ft-Miner. There are three implication quantifiers: $\Rightarrow_{p,s}$, $\Rightarrow^!_{p,\alpha,s}$ and the 4ft quantifier $\Rightarrow^?_{p,\alpha,s}$ of upper critical implication defined by a condition $\sum_{i=0}^{a}\binom{a+b}{i}p^i(1-p)^{a+b-i} > \alpha \wedge a \geq s$, (see subsection 3.1 and [3]).

Further there are three analogous double implication quantifiers: $\Leftrightarrow_{p,s}$, $\Leftrightarrow^!_{p,\alpha,s}$, $\Leftrightarrow^?_{p,\alpha,s}$ and three analogous equivalence quantifiers: $\equiv_{p,s}$, $\equiv^!_{p,\alpha,s}$, $\equiv^?_{p,\alpha,s}$, see also subsection 3.1 and [3].

There are also quantifiers corresponding to χ^2 test and to Fisher's test and the 4ft quantifier defined by the condition $ad > e^{\delta}.bc$ where $\delta \geq 0$. Quantifier $\rightarrow_{C,S}$ corresponding to "classical" association rule is also implemented. Last implemented type of 4ft quantifiers is the 4ft quantifier corresponding to condition $\max(\frac{b}{a+b}, \frac{c}{d+c}) < \gamma$ where $0 \leq \gamma < 1$, see [16].

4.3 Antecedent, Succedent and Condition

Antecedent, succedent and condition are conjunctions of *basic Boolean attributes*. Basic Boolean attribute is of the form $A(\omega)$ where A is the column and ω is a subset of a set of possible values of A. Boolean attribute $A(\omega)$ is true in the row o of analysed data matrix if the value in the column A and in the row o belongs to the set ω. An example of the conditional 4ft association rule:

$$A_1(1,3,4) \wedge A_3(5,6) \quad \Leftrightarrow_{0.9} \quad A_5(8,12) \wedge A_7(11,12,14) \wedge A_8(2) \;/\; A_9(4,5,6,7,8)\ .$$

4.4 Input of 4ft-Miner

An input of 4ft-Miner is given by:

i) The analysed data matrix
ii) 4ft quantifier
iii) Simple definition of all antecedents to be automatically generated. It consists of:
- A list of all columns of data matrix, from which basic Boolean attributes of antecedent will be automatically generated.
- Simple definition of the set of all basic Boolean attributes to be automatically generated from each particular column.
- Minimal and maximal number of basic Boolean attributes in each generated antecedent.

iv) Analogous definitions of all succedents and of all conditions to be automatically generated.

The set of all basic Boolean attributes to be generated from a particular column is given by a *type of subsets* and by the minimal and the maximal number of particular values in the subset. There are five types of subsets to be generated: *all subsets*, *intervals*, *left cuts*, *right cuts* and *cuts*.

Examples of definition of the set of basic Boolean attributes for column A with possible values {1,2,3,4,5}:

1. *All subsets with 2-3 values* defines basic Boolean attributes $A(1,2)$, $A(1,3)$, $\ldots$, $A(4,5)$, $A(1,2,3)$, $A(1,2,4)$, $\ldots$, $A(3,4,5)$;
2. *Intervals with 2-3 values* defines basic Boolean attributes $A(1,2)$, $A(2,3)$, $A(3,4)$, $A(4,5)$, $A(1,2,3)$, $A(2,3,4)$ and $A(3,4,5)$;
3. *Left cuts with most 3 values* defines basic Boolean attributes $A(1)$, $A(1,2)$, $A(1,2,3)$.
4. *Right cuts with most 3 values* defines basic Boolean attributes $A(5)$, $A(4,5)$, $A(3,4,5)$.
5. *Cuts with most 3 values* defines basic Boolean attributes $A(1)$, $A(1,2)$, $A(3,4,5)$, $A(5)$, $A(4,5)$, $A(3,4,5)$.

4.5 Output of 4ft-Miner

4ft-Miner automatically generates all 4ft association rules or all conditional 4ft association rules given by the conditions (ii) - (iv) (usually 10^5 - 10^7) *and verifies them* in data matrix given by (i). Output of 4ft-Miner is the set of all 4ft association rules (all conditional 4ft association rules) true in data matrix given by (i).

Usual output of 4ft-Miner consists of tens or hundreds of true 4ft association rules (true conditional 4ft association rules). There are strong tools for dealing with output of 4ft-Miner. It is possible to sort output 4ft association rules by various criterions. Flexible conditions can be used to define subsets of output 4ft association rules. It is also possible to export defined subsets in several formats.

4.6 Some Further Features of 4ft-Miner

4ft-Miner works under WINDOWS, analysed data matrix can be stored in a database (ODBC is applied). New values can be defined for particular columns (e.g. intervals or groups of original values). New columns of data matrix can be also defined (SQL - like) and used in conditions (iii) - (v).

4ft-Miner works very fast. Usual task (data matrix with 10^4 rows, several millions of 4ft association rules to be generated and verified) requires only several minutes at PC with Pentium II and with 128 MB of operational memory. Several optimisation techniques and deep theoretical results are used, e.g. bit strings for representation of analysed data matrix [7], deduction rules (see subsection 3.4) and tables of critical frequencies mentioned in subsection 3.5.

References

1. Aggraval, R. et al: Fast Discovery of Association Rules. In Fayyad, U. M. et al.: Advances in Knowledge Discovery and Data Mining. AAAI Press , 1996. 307–328
2. Berka, P. - Rauch, J.: Data Mining using GUHA and KEX. In proceedings of World Multiconference on Systemics, Cybernetics and Informatics. Red. Callaos, N. - Yang, T. - Aguilar, J. Orlando Florida 1998, pp. 238 - 244
3. Hájek, P. - Havránek T.: Mechanising Hypothesis Formation - Mathematical Foundations for a General Theory. Springer-Verlag, 1978, 396 p.

4. Hájek, P. - Havránek, T. - Chytil M.: Metoda GUHA. Praha, Academia, 1983, 314 p. (In Czech)
5. Hájek, P. - Sochorová, A. - Zvárová, J.: GUHA for personal computers. Computational Statistics & Data Analysis 19, (1995) pp 149 - 153
6. Havránek, T. - Chyba, M. - Pokorn D.: Processing of sociological data by the GUHA method - an example International Journal of Man-Machine Studies, 9, 439 – 447
7. Rauch J.: Some Remarks on Computer Realisations of GUHA Procedures. International Journal of Man-Machine Studies, 10, (1978) 23–28
8. Rauch, J.: GUHA as a Data Mining Tool. In: Practical Aspects of Knowledge Management. Schweizer Informatiker Gesellshaft Basel, 1996,
9. Rauch, J.: Logical Calculi for Knowledge Discovery in Databases. In Principles of Data Mining and Knowledge Discovery, (J. Komorowski, J. Zytkow, eds.), Springer-Verlag, 47-57, 1997.
10. Rauch, J.: Classes of Four-Fold Table Quantifiers. In Principles of Data Mining and Knowledge Discovery, (J. Zytkow, M. Quafafou, eds.), Springer-Verlag, 203-211, 1998.
11. Rauch, J.: Four-Fold Table Calculi and Missing Information. In JCIS'98 Proceedings, (Paul P. Wang, editor), Association for Intelligent Machinery, 375-378, 1998.
12. Rauch, J.: Contribution to Logical Foundations of KDD (in Czech). Assoc. Prof. Thesis, Faculty of Informatocs and Statistics, University of Economics Prague, 1998, 137 p.
13. Rauch, J. - Šimůnek M.: Mining for 4ft Association Rules. In Discovery Science 2000. Red. Arikawa, S. - Morishita S. Springer Verlag 2000, pp. 268 - 272
14. Rauch, J.: Mining for Statistical Association Rules. In The Fifth Pacific-Asia Conference on Knowledge Discovery and Data Mining Industrial Track and Workshop Proceeding Red. Joseph Fong ang Michael Ng Hong Kong 2001, pp. 149 –158
15. Vlček, V.: An application of the GUHA method to chemical engineering GUHA. International Journal of Man-Machine Studies, 15, 299 – 307
16. Zembowicz R. - Zytkow J.: From Contingency Tables to Various Forms of Knowledge in Databases. in Fayyad, U. M. et al.: Advances in Knowledge Discovery and Data Mining. AAAI Press/ The MIT Press, 329 - 349, 1996.

This work is supported by grant EU No IST-1999-11495: Data Mining and decision support for business competitiveness: Solomon European Virtual Enterprise.