AN INCLUSION-EXCLUSION RESULT FOR BOOLEAN POLYNOMIALS AND ITS APPLICATIONS IN DATA MINING

SZYMON JAROSZEWICZ* , DAN A. SIMOVICI*, AND IVO ROSENBERG†

Abstract. We characterize measures on free Boolean algebras and we examine the relationships that exists between measures and binary tables in relational databases. It is shown that these measures are completely defined by their values on positive conjunctions, and a formula that obtains this value is given by using the method of indicators. We also obtain Bonferroni-type inequalities that allow approximative evaluations of there measures. Finally we present a measure extending the notion of support that is well suited for tables with missing values.

Key words. free Boolean algebra, measure, Bonferroni-type inquality, exclusion-inclusion, missing values

1. Introduction. The focus of this paper is a study of measures on free Boolean algebras with a finite number of generators (abbreviated as MFBAs) who take their values in the set $\mathbb{N}$ of natural numbers. As we shall see, these measures play an important role in query optimization in relational databases, and also, in the study of frequent sets in data mining. We obtain general Bonferroni-type inequalities for sizes of arbitrary Boolean queries. The origin of our investigation resides in a series of seminal papers by H. Mannila et al. ([3, 2, 5]) in which the idea of using supports of attribute sets discovered with a data mining algorithm to obtain the size of a database query was introduced.

Let $\mathcal{B} = (B, \mathbf{0}, \mathbf{1}, \bar{}\,, \vee, \wedge)$ be a Boolean algebra, where $\mathbf{0}, \mathbf{1} \in B$ are two distinguished elements of $\mathcal{B}$, $\bar{}$ is a unary operation, and $\vee, \wedge$ are two binary associative, commutative, and idempotent operation that satisfy the usual axioms of Boolean algebras (see, for example [7]). Here $\mathbf{0}$ and $\mathbf{1}$ are the least and the largest element of the algebra, respectively.

We define

$$x^b = \begin{cases} x & \text{if } b = \mathbf{1} \\ \bar{x} & \text{if } b = \mathbf{0}, \end{cases}$$

for $x \in B$ and $b \in \{\mathbf{0}, \mathbf{1}\}$.

It is a well-known fact that a Boolean algebra $\mathcal{B} = (B, \mathbf{0}, \mathbf{1}, \bar{}\,, \vee, \wedge)$ defines a Boolean ring structure, $\mathcal{B} = (B, \mathbf{0}, \mathbf{1}, \wedge, \oplus)$, where $\wedge$ plays the role of the multiplication, and $\oplus$ the role of addition, where

$$x \oplus y = (x \wedge \bar{y}) \vee (\bar{x} \wedge y)$$

for $x, y \in B$. This ring is unitary, commutative, and has characteristic 2 (since $x \oplus x = \mathbf{0}$ for every x). Also, $\mathbf{1} \oplus x = \bar{x}$.

Let $A = \{a_1, \ldots, a_n\}$ be a set of n variables. The set $\mathsf{pol}(A)$ of *Boolean polynomials of the n variables in A* is defined inductively by:

1. $\mathbf{0}$, $\mathbf{1}$, and each a_i belong to $\mathsf{pol}(A)$ for $1 \leq i \leq n$;
2. if p, q belong to $\mathsf{pol}(A)$, then $\bar{p}$, $(p \vee q)$, and $(p \wedge q)$ belong to $\mathsf{pol}(A)$.

*University of Massachusetts at Boston, Department of Computer Science, Boston, Massachusetts 02125, USA

†Université de Montréal, C.P. 6128, succ. A, Montréal, P.Q. H3C 3J7, Canada

If $p, q \in \mathsf{pol}(A)$, then we denote by $(p \oplus q)$ the polynomial $((p \wedge \bar{q}) \vee (\bar{p} \wedge q))$.

A Boolean polynomial $(\cdots((p_1 \omega p_2)\omega p_3)\omega \cdots \omega p_n)$ is denoted by $(p_1 \omega p_2 \omega \cdots \omega p_n)$, where $\omega \in \{\vee, \wedge, \oplus\}$. Also, we denote by $\mathsf{var}(p)$ the set of variables that occur in the polynomial p.

Let $\mathcal{B} = (B, \mathbf{0}, \mathbf{1}, {}^-, \vee, \wedge)$ be a Boolean algebra and let $A = \{a_1, \ldots, a_n\}$ be a set of n attributes. The n-ary function $f_p : B^n \longrightarrow B$ generated by a polynomial $p \in \mathsf{pol}(A)$ is defined in the usual way. We write $p = q$ for $p, q \in \mathsf{pol}(A)$ if $f_p = f_q$.

Let $\vec{b} = (b_1, \ldots, b_n)$ be a sequence of elements of the set $\{\mathbf{1}, \mathbf{0}\}$. An A-minterm is a Boolean polynomial

$$p_{\vec{b}} = a_1^{b_1} \wedge \cdots \wedge a_n^{b_n},$$

The set of A-minterms is denoted by $\mathsf{mint}(A)$. Any Boolean polynomial in $\mathsf{pol}(A)$ can be uniquely written as a disjunction of some subset of A-minterms (up to the order of the disjuncts). This observation implies that the Boolean algebra $\mathsf{pol}(A)$ is isomorphic to the Boolean algebra of collections of subsets of the set A; thus, $\mathsf{pol}(A)$ has 2^{2^n} elements.

For a set of polynomials $M = \{p_1, \ldots, p_n\}$ and $J = \{j_1, \ldots, j_m\} \subseteq \{1, \ldots, n\}$ we denote by p_J the conjunction $p_{j_1} \wedge \cdots \wedge p_{j_m}$. For the special case, when $J = \emptyset$ we write $p_J = \mathbf{1}$.

A *measure* on a Boolean algebra $\mathcal{B} = (B, \mathbf{0}, \mathbf{1}, {}^-, \vee, \wedge)$ is a non-negative, real-valued function $\mu : B \longrightarrow \mathbb{R}$ such that $\mu(x \vee y) = \mu(x) + \mu(y)$ for every $x, y \in B$ such that $x \wedge y = \mathbf{0}$.

2. A Representation Result for MFBAs. Let $A = \{a_1, \ldots, a_n\}$ be a set of variables. In this context, we find convenient to use the relational database terminology and we refer to the the members of A as *attributes*. We attach a set $\mathsf{Dom}(a_i)$ to each attribute a_i such that $|\mathsf{Dom}(a_i)| \geq 2$. The set $\mathsf{Dom}(a_i)$ is the *domain of* a_i.

A table is a triple $\tau = (T, A, \rho)$, where T is the name of the table, $A = \{a_1, \ldots, a_n\}$ is the heading of the table and $\rho = \{t_1, \ldots, t_m\}$ is a finite set of functions of the form $t_i : A \longrightarrow \bigcup_{a \in A} \mathsf{Dom}(a)$ such that $t_i(a) \in \mathsf{Dom}(a)$ for every $a \in A$. Following the relational database terminology we shall refer to these functions as A-tuples, or simpler, as tuples. If $\mathsf{Dom}(a_i) = \{\mathbf{0}, \mathbf{1}\}$ for $1 \leq i \leq n$, then τ is a binary table.

Let $\tau = (T, A, \rho)$ be a binary table. A *query on the table* τ is a Boolean polynomial in $\mathsf{pol}(A)$. This definition of queries is a formalization of the usual notion of query in databases.

EXAMPLE 2.1. To retrieve in SQL all tuples t of τ such that at least two of $t(a_1), t(a_2)$ and $t(a_3)$ equal $\mathbf{1}$ we write the query as

select * **from** T **where** $(a_1 = \mathbf{1}$ **and** $a_2 = \mathbf{1})$ **or**
$(a_2 = \mathbf{1}$ **and** $a_3 = \mathbf{1})$ **or** $(a_1 = \mathbf{1}$ **and** $a_3 = \mathbf{1})$;

The condition specified in this select corresponds to the polynomial $(a_1 \wedge a_2) \vee (a_2 \wedge a_3) \vee (a_1 \wedge a_3)$. □

A query p defines a table $\mathcal{Q}(p, \tau) = (T_p, A, \rho_p)$, where ρ_p is defined inductively as follows:

1. $\rho_0 = \emptyset$ and $\rho_1 = \rho$;
2. if $p = a_i$, then $\rho_p = \{t \in \rho | t(a_i) = \mathbf{1}\}$;
3. if $p = \bar{q}$, then $\rho_p = \rho - \rho_q$;
4. if $p = (q_1 \vee q_2)$, then $\rho_p = \rho_{p_1} \cup \rho_{p_2}$ and,
5. if $p = (q_1 \wedge q_2)$, then $\rho_p = \rho_{p_1} \cap \rho_{p_2}$.

It is easy to see that for a conjunction

$$p = a_{i_1}^{b_1} \wedge \cdots \wedge a_{i_m}^{b_m},$$

where $b_i \in \{\mathbf{0}, \mathbf{1}\}$ for $1 \leq i \leq m$, the set ρ_p consists of those tuples t such that $t(a_{i_\ell}) = b_\ell$ for $1 \leq \ell \leq m$.

THEOREM 2.1. *A function $\mu : \mathsf{pol}(A) \longrightarrow \mathbb{N}$ is a measure if and only if there exists a binary table $\tau = (T, A, \rho)$ such that $\mu(p) = |\rho_p|$ for all $p \in \mathsf{pol}(A)$.*

Proof. Suppose that $\tau = (T, A, \rho)$ is a table. Define the mapping $\mu_\tau : \mathsf{pol}(A) \longrightarrow \mathbb{R}$ by $\mu(p) = |\rho_p|$ for every $p \in \mathsf{pol}(A)$. Let p, q be two polynomials such that $(p \wedge q) = \mathbf{0}$. Then, $\mu_\tau(p \vee q) = |\rho_{p \vee q}| = |\rho_p \cup \rho_q|$. Since $p \wedge q = \mathbf{0}$ we have $\rho_p \cap \rho_q = \emptyset$, so $\mu_\tau(p \vee q) = \mu_\tau(p) + \mu_\tau(q)$. Thus, μ_τ is a measure on $\mathsf{pol}(A)$.

Conversely, let μ be a measure on $\mathsf{pol}(A)$, where $A = \{a_1, \ldots, a_n\}$. If $\vec{b} = (b_1, \ldots, b_n) \in \{\mathbf{0}, \mathbf{1}\}^n$, $p_{\vec{b}} = a_1^{b_1} \wedge \cdots \wedge a_n^{b_n}$ is a minterm and $\mu(p_{\vec{b}}) = k$ consider a set $\sigma_{p_{\vec{b}}}$ of k tuples $t_{\vec{b}}^1, \ldots, t_{\vec{b}}^k$, where $t_{\vec{b}}^j(a_i) = b_i$ for every i, j, $1 \leq j \leq k$, and $1 \leq i \leq n$. Define the table $\tau_\mu = (T, A, \rho_\mu)$, where $\rho = \bigcup\{\sigma_{p_{\vec{b}}} | p_{\vec{b}} \in \mathsf{mint}(A)\}$.

We claim that $\mu(p) = |\rho_p|$ for every polynomial $p \in \mathsf{pol}(A)$. Suppose that p can be expressed as a disjunction of minterms $p = p_{\vec{b}_1} \vee \cdots \vee p_{\vec{b}_k}$, where $\vec{b}_1, \ldots, \vec{b}_k \in \{\mathbf{0}, \mathbf{1}\}^n$. Then, $\mu(p) = \sum_{j=1}^k \mu(p_{\vec{b}_j})$, because $p_{\vec{b}_l} \wedge p_{\vec{b}_h} = \mathbf{0}$ when $l \neq h$. On the other hand, $|\rho_p| = |\bigcup_{j=1}^k \rho_{p_{\vec{b}_j}}| = \sum_{j=1}^k |\rho_{p_{\vec{b}_j}}|$, so $\mu(p) = |\rho_p|$. □

We shall refer to μ_τ as the *measure induced by the table* τ on $\mathsf{pol}(A)$.

In the next section we regard the set of minterms $\mathsf{mint}(A)$ as a sample space and each polynomial $p \in \mathsf{pol}(A)$ as an event on this sample space. The event p occurs in $p_{\vec{b}}$ if $p_{\vec{b}} \leq p$. Thus, if μ is a measure on $\mathsf{pol}(A)$, then the mapping $P_\mu : \mathsf{pol}(A) \longrightarrow \mathbb{R}$ given by $P_\mu(p) = \frac{\mu(p)}{\mu(\mathbf{1})}$ is a probability on $\mathsf{pol}(A)$.

3. An Exclusion-Inclusion Principle for MFBAs. Let p be a polynomial in $\mathsf{pol}(A)$. It is known that p can be uniquely written as

$$p = \sum_{(i_1, \ldots, i_m)}^{\oplus} c_{(i_1, \ldots, i_m)} \wedge a_{i_1} \wedge \cdots a_{i_m},$$

where the summation $\sum^{\oplus}$ involves the "exclusive or" operation $\oplus$ and is extended to all subsets $\{i_1, \ldots, i_m\}$ of $\{1, \ldots, n\}$. The coefficients $c_{(i_1, \ldots, i_m)}$ belong to the set $\{\mathbf{0}, \mathbf{1}\}$. Thus, for a measure μ on $\mathsf{pol}(A)$ it is interesting to evaluate $\mu(p_1 \oplus p_2 \oplus \cdots \oplus p_m)$, where $p_1, \ldots, p_m$ are polynomials in $\mathsf{pol}(A)$.

The *indicator random variable* of a polynomial p is the variable I_p defined by

$$I_p(p_{\vec{b}}) = \begin{cases} 1 & \text{if } p_{\vec{b}} \leq p \\ 0 & \text{otherwise.} \end{cases}$$

for $p_{\vec{b}} \in \mathsf{mint}(A)$.

If $M = \{p_1, \ldots, p_n\}$ and $J = \{j_1, \ldots, j_m\} \subseteq \{1, \ldots, n\}$, then $p_J = p_{j_1} \wedge \ldots \wedge p_{j_m}$, and $I_{p_J} = I_{p_{j_1}} \cdots I_{p_{j_m}}$.

Note that the expected value $E[I_p]$ of I_p equals $P_\mu(p)$.

For a set of polynomials M denote by $S_{M,k}^\mu$ the probability that exactly k events in M hold:

$$S_{M,k}^\mu = \sum\{P_\mu(p_K) \mid |K| = k\}.$$

The number of k-subsets K of M such that p_K holds is given by the random variable $\sum\{I_{p_K} \mid |K| = k\}$. By the previous observation

$$S^{\mu}_{M,k} = \sum\{E(I_{p_K}) \mid |K| = k\} = E\left[\sum\{I_{p_K} \mid |K| = k\}\right].$$

Let ν_M be the random variable on $\mathsf{mint}(A)$ such that $\nu_M(p_{\vec{b}}) = |\{p_i \in M \mid p_{\vec{b}} \leq p_i\}|$. Note that ν_M gives the number of events in M that hold and, therefore, the random variable $\binom{\nu_M}{k}$ gives the number of k-subsets Q of M such that p_Q holds, which means that $\binom{\nu_M}{k} = \sum\{I_{p_K} \mid |K| = k\}$, and

$$S^{\mu}_{M,k} = E\left[\binom{\nu_M}{k}\right]. \tag{3.1}$$

The equality (3.1) is the basis of the method of indicators, that is a method of proving probabilistic identities by taking expectations of their non-probabilistic counterparts, see [1] for details.

THEOREM 3.1. *Let $\mu : \mathsf{pol}(A) \longrightarrow \mathbb{N}$ be a measure on the free Boolean algebra $\mathsf{pol}(A)$, where $A = \{a_1, \ldots, a_n\}$. If $P = \{p_1, \ldots, p_m\}$ is a set of m polynomials of $\mathsf{pol}(A)$, then*

$$\mu(p_1 \oplus \cdots \oplus p_m) = \sum_{k=1}^{m} (-2)^{k-1} \cdot \sum\{\mu(p_K) \mid K \subseteq \{1, \ldots, m\}, |K| = k\}. \tag{3.2}$$

Proof. Let $a \in \mathbb{N}$, note that $(-1)^a = \sum_{k=0}^{a} (-2)^k \binom{a}{k}$, which yields, after elementary transformations:

$$\sum_{k=1}^{a} (-2)^{k-1} \binom{a}{k} = (-1)^a - 1 = \begin{cases} 0 & \text{if } a \text{ is even} \\ 1 & \text{if } a \text{ is odd.} \end{cases}$$

This implies

$$\sum_{k=1}^{\nu_M} (-2)^{k-1} \binom{\nu_m}{k} = \sum_{k=1}^{|M|} (-2)^{k-1} \binom{\nu_m}{k} = \begin{cases} 0 & \text{, if } \nu_n \text{ is even} \\ 1 & \text{, if } \nu_n \text{ is odd.} \end{cases}$$

By taking expectations of both sides, and using equality (3.1) we get

$$E\left[\sum_{k=1}^{|M|} (-2)^{k-1} \binom{\nu_m}{k}\right]$$
$$= \sum_{k=1}^{|M|} (-2)^{k-1} E\left[\binom{\nu_m}{k}\right] = \sum_{k=1}^{|M|} (-2)^{k-1} S^{\mu}_{M,k} = P_{\mu}(\nu_m \text{ is odd}) = P_{\mu}(p_1 \oplus \ldots \oplus p_m).$$

Since

$$S^{\mu}_{M,k} = \frac{\sum\{\mu(p_K) \mid K \subseteq \{1, \ldots, m\}, |K| = k\}}{\mu(\mathbf{1})},$$

we obtain the desired equality. □

COROLLARY 3.2. *Let $\mu, \mu' : \mathsf{pol}(A) \longrightarrow \mathbb{N}$ be two measures on the free Boolean algebra $\mathsf{pol}(A)$, where $A = \{a_1, \ldots, a_n\}$. If $\mu(p) = \mu'(p)$ for every conjunction p of the form $p = a_{i_1} \wedge \cdots \wedge a_{i_m}$, then $\mu = \mu'$.*

Proof. The result follows immediately from Theorem 3.1. □

EXAMPLE 3.1. Consider the "majority polynomial" $p_{maj} = (a_1 \wedge a_2) \vee (a_2 \wedge a_3) \vee (a_1 \wedge a_3)$. For $f_{p_{maj}}$ we have $f_{p_{maj}}(x_1, x_2, x_3) = 1$ if and only if at least two of its arguments are equal to 1. Note that

$$p_{maj} = (a_1 \wedge a_2) \oplus (a_2 \wedge a_3) \oplus (a_1 \wedge a_3).$$

Theorem 3.1 allows us to write

$$\begin{aligned}\mu(p_{maj}) = {} & \mu(a_1 \wedge a_2) + \mu(a_2 \wedge a_3) + \mu(a_1 \wedge a_3) \\ & -2\mu((a_1 \wedge a_2) \wedge (a_2 \wedge a_3)) - 2\mu((a_1 \wedge a_2) \wedge (a_1 \wedge a_3)) \\ & -2\mu((a_2 \wedge a_3) \wedge (a_1 \wedge a_3)) + 4\mu((a_1 \wedge a_2) \wedge (a_2 \wedge a_3) \wedge (a_1 \wedge a_2)) \\ = {} & \mu(a_1 \wedge a_2) + \mu(a_2 \wedge a_3) + \mu(a_1 \wedge a_3) - 2\mu(a_1 \wedge a_2 \wedge a_3).\end{aligned}$$

□

Corollary 3.2 shows that the values of a measure on $\mathsf{pol}(A)$ is completely determined by its values on positive conjunctions of the form a_I for $I \subseteq \{1, \ldots, n\}$. Note that the contribution of every tuple of a table $\tau = (T, A, \rho)$ of the form $(b_1, \ldots, b_n)$ to the value of $\mu_\tau(I)$ equals 1 for every set I such that $I \subseteq \{i \in \{1, \ldots n\} \mid b_i = \mathbf{1}\}$.

Next, we obtain Bonferroni type inequalities [1] that give bounds on the value of $\mu(p_1 \oplus \ldots \oplus p_m)$. To this end we need the following technical result:

Define W_b^a for $a, b \in \mathbb{N}$ and $b \leq a$ as

$$W_b^a = \sum_{k=b}^{a} (-2)^{k-1} \binom{a}{k}$$

Alternatively, W_b^a can be written as

$$W_b^a = (-2)^{b-1} \sum_{k=b}^{a} (-2)^{k-b} \binom{a}{k} = (-2)^{b-1} \sum_{\ell=0}^{a-b} (-2)^{\ell} \binom{a}{b+\ell}.$$

LEMMA 3.3. *The signs of the members of the sequence $(W_b^a, W_{b+1}^a, \ldots, W_a^a)$ are alternating.*

THEOREM 3.4. *For any $r, s \in \mathbb{N}$ we have:*

$$\mu(\mathbf{1}) \cdot \sum_{k=1}^{2r} (-2)^{k-1} S_k^\mu \leq \mu(p_1 \oplus \ldots \oplus p_m) \leq \mu(\mathbf{1}) \cdot \sum_{k=1}^{2s+1} (-2)^{k-1} S_k^\mu.$$

Proof. By equality (3.1) and Lemma 3.3 we get that for any $r, s \in \mathbb{N}$

$$\sum_{k=1}^{2r} (-2)^{k-1} \binom{a}{k} \leq \sum_{k=1}^{a} (-2)^{k-1} \binom{a}{k} \leq \sum_{k=1}^{2s+1} (-2)^{k-1} \binom{a}{k},$$

implying

$$\sum_{k=1}^{2r} (-2)^{k-1} \binom{\nu_M}{k} \leq \sum_{k=1}^{|M|} (-2)^{k-1} \binom{\nu_M}{k} \leq \sum_{k=1}^{2s+1} (-2)^{k-1} \binom{\nu_M}{k}.$$

By applying expectations and using equality (3.1) we get the desired result. □

EXAMPLE 3.2. Consider a table τ given below

a_1	a_2	a_3
0	0	0
0	1	0
1	0	0
0	0	0
0	1	0
1	0	1
0	1	1
1	1	0
1	1	0
0	1	1
1	0	1
1	1	0
1	1	1

and the majority polynomial p_{maj} from Example 3.1. We have $\mu(a_1 \wedge a_2) = 4$, $\mu(a_1 \wedge a_3) = 3$, $\mu(a_2 \wedge a_3) = 3$, giving $\mu(p_{maj}) \leq 10$. Also $\mu((a_1 \wedge a_2) \wedge (a_1 \wedge a_3)) = \mu((a_1 \wedge a_2) \wedge (a_2 \wedge a_3)) = \mu((a_1 \wedge a_3) \wedge (a_2 \wedge a_3)) = 1$ giving $\mu(p_{maj}) \geq 4$. The true value of $\mu(p_{maj})$ is 8. □

Let $\mathcal{V} : \{\mathbf{0}, \mathbf{1}\} \longrightarrow \{0, 1\}$ be the bijection defined by $\mathcal{V}(\mathbf{0}) = 0$ and $\mathcal{V}(\mathbf{1}) = 1$, where $0, 1 \in \mathbb{R}$. Note that

$$\mathcal{V}(a \vee b) = \mathcal{V}(a) + \mathcal{V}(b) - \mathcal{V}(a)\mathcal{V}(b), \tag{3.3}$$

$$\mathcal{V}(a \wedge b) = \mathcal{V}(a)\mathcal{V}(b), \tag{3.4}$$

$$\mathcal{V}(\bar{a}) = 1 - \mathcal{V}(a), \tag{3.5}$$

for every $a, b \in \{\mathbf{0}, \mathbf{1}\}$.

For a Boolean function $f : \{\mathbf{0}, \mathbf{1}\}^n \longrightarrow \{\mathbf{0}, \mathbf{1}\}$ define the real-valued function $\phi_f : \{0, 1\}^n \longrightarrow \{0, 1\}$ by

$$\phi_f(\xi_1, \ldots, \xi_n) = \mathcal{V}(f(\mathcal{V}^{-1}(\xi_1), \ldots, \mathcal{V}^{-1}(\xi_n)))$$

for every $\xi_1, \ldots, \xi_n \in \{0, 1\}$.

EXAMPLE 3.3. It is easy to verify that if p is the majority polynomial considered in Example 3.1, then for the numerical function ϕ_{f_p} we can write:

$$\phi_{f_p}(\xi_1, \xi_2, \xi_3) = \xi_1\xi_2 + \xi_2\xi_3 + \xi_1\xi_3 - 2\xi_1\xi_2\xi_3$$

for every $\xi_1, \xi_2, \xi_3 \in \{0, 1\}$. Note that the coefficients are the same as the ones in Example 3.1. □

The remark contained in the above Example is not a coincidence. Next, we prove that for every polynomial p, the numerical function ϕ_{f_p} can be expressed as a sum of monomials multiplied by the coefficients that occur in the expansion of $\mu(p)$ given in Theorem 3.1.

THEOREM 3.5. *Let $A = \{a_1, \ldots, a_n\}$ and $p \in \mathsf{pol}(A)$. Suppose that*

$$\mu(p) = \sum_{I \in \mathcal{I}} c_I \mu(a_I),$$

where $\mathcal{I}$ is a family of subsets of $\{1, \ldots, n\}$. Then, we have:

$$\phi_{f_p}(\xi_1, \ldots, \xi_n) = \sum_{I \in \mathcal{I}} c_I \xi_I,$$

where ξ_I is the monomial $\xi_I = \xi_{i_1} \cdots \xi_{i_m}$.

Proof. Let $p, q, r \in \mathsf{pol}(A)$ such that $r = (p \vee q)$. We have:

$$\begin{aligned}\phi_{f_r}(\xi_1, \ldots, \xi_n) &= \mathcal{V}(f_r(\mathcal{V}^{-1}(\xi_1, \ldots, \xi_n))) \\ &= \mathcal{V}(f_p(\mathcal{V}^{-1}(\xi_1, \ldots, \xi_n)) \vee f_q(\mathcal{V}^{-1}(\xi_1, \ldots, \xi_n)) \\ &= \mathcal{V}(f_p(\mathcal{V}^{-1}(\xi_1, \ldots, \xi_n)) + f_q(\mathcal{V}^{-1}(\xi_1, \ldots, \xi_n)) \\ &\quad - \mathcal{V}(f_p(\mathcal{V}^{-1}(\xi_1, \ldots, \xi_n))) \cdot \mathcal{V}(f_q(\mathcal{V}^{-1}(\xi_1, \ldots, \xi_n))),\end{aligned}$$

for $(x_1, \ldots, x_n) \in \{\mathbf{0}, \mathbf{1}\}^n$, by equality(3.3). Thus,

$$\phi_{f_r}(\xi_1, \ldots, \xi_n) = \phi_{f_p}(\xi_1, \ldots, \xi_n) + \phi_{f_q}(\xi_1, \ldots, \xi_n) - \phi_{f_p}(\xi_1, \ldots, \xi_n)\phi_{f_q}(\xi_1, \ldots, \xi_n)$$

for $(\xi_1, \ldots, \xi_n) \in \{0, 1\}^n$. Since $\phi_{f_{p \wedge q}}(\xi_1, \ldots, \xi_n) = \phi_{f_p}(\xi_1, \ldots, \xi_n)\phi_{f_q}(\xi_1, \ldots, \xi_n)$, $p \wedge q = \mathbf{0}$ implies

$$\phi_{f_r}(\xi_1, \ldots, \xi_n) = \phi_{f_p}(\xi_1, \ldots, \xi_n) + \phi_{f_q}(\xi_1, \ldots, \xi_n)$$

for every $(\xi_1, \ldots, \xi_n) \in \{0, 1\}^n$. This shows that for a every $\xi = (\xi_1, \ldots, \xi_n)$, the mapping $\mu : \mathsf{pol}(A) \longrightarrow \mathbb{R}$ defined by $\mu_\xi(p) = \phi_{f_p}(\xi)$ is a measure on $\mathsf{pol}(A)$, so Theorem 3.1 is applicable and we can write:

$$\phi_{f_p}(\xi_1, \ldots, \xi_n) = \sum_{I \in \mathcal{I}} c_I \xi_I,$$

for every $(\xi_1, \ldots, \xi_n) \in \{0, 1\}^n$. □

4. Applications in Data Mining and Database Query Optimization.

4.1. Accuracy of Inclusion-Exclusion Principle. In database query optimization and in data mining, it is often necessary to estimate the number of rows in a database table satisfying a given query. Unfortunately, in most cases, the exact number of rows satisfying a query cannot be computed exactly and has to be estimated (usually using the assumption of statistical independence between attributes).

Let $\tau = (T, A, \rho)$ be a binary table and let K be a set of attributes, $K \subseteq A$. The support of the set K relative to the table τ is defined as the number:

$$\mathtt{supp}_\tau(K) = \{t \in \rho \mid t(a) = \mathbf{1} \text{ for all } a \in K\}.$$

Thus, $\mathtt{supp}_\tau(K) = \mu_\tau(a_{k_1} \wedge \ldots \wedge a_{k_m})$, where $K = \{a_{k_1}, \ldots, a_{k_m}\}$. In other words, the support of an attribute set K in the table τ can we viewed as the value of the measure induced by the table on the Boolean polynomial that describes the attribute set. By extension, we can regard the number $\mu_\tau(q)$ as the support of the query q and we denote this number by $\mathtt{supp}(q)$. Indeed, if $q \in \mathsf{pol}(A)$ is a query involving a table $\tau = (T, A, \rho)$ such that q can be written as

$$q = c \oplus \overset{\oplus}{\sum_{I \in \mathcal{I}}} a_I,$$

where $c \in \{\mathbf{0}, \mathbf{1}\}$ and $\mathcal{I}$ is a collection of subsets of $\{1, \ldots, n\}$, then $\mathtt{supp}(q)$ can be obtained from Theorem 3.1 using the numbers $\mathtt{supp}_\tau(a_I)$. Methods that obtain approximative estimations of query sizes been proposed [2], including the use of Maximum Entropy Principle. An open problem raised was estimating the quality of such an approximation.

The computation of the size of the query using Theorem 3.1 can be often simplified if there is a known maximal number of **1** components in the tuples of the table. For example, in a store that sells 1000 items (corresponding to 1000 attributes in a table that contains the records of purchases) it is often the case that we can use an empirical limit of, say, 8 items per tuple. In this case, conjunctions that contain more than 8 conjuncts can be discarded and the estimation is considerably simplified. Even, if such an upper bound cannot be imposed apriori, it is often the case that we can discard large conjunctions (which have low support). However, there are some risks when approximations of this nature are performed due to the the large values of coefficients that multiply the supports for large conjunctions.

Indeed, consider the tables $\tau^n_{odd} = (T_o, A, \rho_{odd})$, $\tau^n_{even} = (T_e, A, \rho_{even})$, where

$$\rho_{odd} = \{t \in \mathsf{Dom}(A) : n_1(t) \text{ is odd}\}, \rho_{even} = \{t \in \mathsf{Dom}(A) : n_1(t) \text{ is even}\},$$

where $n_1(t)$ denotes the number of attributes equal to **1** in tuple t and $|A| = n$.

Note that for proper subset K of A, we have $\mathtt{supp}_{\tau_{odd}}(K) = \mathtt{supp}_{\tau_{even}}(K)$, while

$$\mathtt{supp}_{\tau^n_{odd}}(A) = \begin{cases} 1 & \text{if } n \text{ is odd} \\ 0 & \text{otherwise,} \end{cases} \text{ and } \mathtt{supp}_{\tau^n_{even}}(A) = \begin{cases} 1 & \text{if } n \text{ is even} \\ 0 & \text{otherwise.} \end{cases}$$

Thus, from the point of view of the supports of any proper subset of the attribute set the tables τ^n_{odd} and τ^n_{even} are indiscernible. However, the support of certain queries can be vastly different on these tables. For example, consider the polynomial $p = a_1 \oplus a_2 \oplus \ldots \oplus a_n$. We have $\mu_{\tau^n_{odd}}(p) = |\rho_{odd}| = 2^{n-1}$ and $\mu_{\tau^n_{even}}(p) = |\rho_{even}| = 0$. So, ignoring the term that corresponds to the support for a single attribute set (note that this is also the attribute set with the smallest possible support) has a huge impact on $\mu(p)$. Note that the result is consistent with Theorem (3.1) which gives the set of attributes A a coefficient 2^{n-1}. We stress however that the negative result above does not rule out practical applicability of approximating the values of μ_τ since the parity function query used above is by no means a typical database query.

4.2. Support in tables with missing values. Many real world datasets contain missing values, and it is important to adequately address this issue. Here we present a generalization of the notion of support which takes missing values into account. The idea is related to the *hot deck imputation* of missing values where each missing value is replaced by a value randomly drawn from some distribution.

Suppose that $\tau = (T, A, \rho)$ is a table such that $A = \{a_1, \ldots, a_n\}$ and $\mathsf{Dom}(a_i) = \{\mathbf{0}, \mathbf{u}, \mathbf{1}\}$ for $1 \leq i \leq n$. The symbol $\mathbf{u}$ represents *null values*, that is, values that are missing or undefined.

With every attribute $a_i \in A$ we associate a real number $\alpha_i \in [0, 1]$. Intuitively, this number corresponds to the probability of $a_i = 1$, and can be obtained using the non-missing values for the attribute or based on background knowledge.

Let $\mu^{\mathbf{u}} : \mathsf{pol}(A) \longrightarrow \mathbb{R}$ be defined as follows. For a minterm $a_1^{b_1} \ldots a_n^{b_n}$ let

$$\mu^{\mathbf{u}}(a_1^{b_1} \wedge \cdots \wedge a_n^{b_n}) = \sum \left\{ \prod_{i=1}^{n} \alpha_i^{\langle b_i, c_i \rangle} \cdot \mathtt{supp}_\tau \left(\bigwedge_{j=1}^{n} (a_j = b_j^{(c_j)}) \right) \mid (c_1, \ldots, c_n) \in \{0,1\}^n \right\},$$

where

$$\alpha^{\langle b,c\rangle} = \begin{cases} \alpha & \text{, if } b = \mathbf{1} \text{ and } c = 0 \\ 1-\alpha & \text{, if } b = \mathbf{0} \text{ and } c = 0 \\ 1 & \text{, if } c = 1, \end{cases} \qquad b^{(c)} = \begin{cases} b & \text{, if } c = 1 \\ \mathbf{u} & \text{, if } c = 0, \end{cases}$$

where $b \in \{\mathbf{1}, \mathbf{0}\}$, and $c \in \{0, 1\}$.

For an arbitrary boolean polynomial p define

$$\mu^{\mathbf{u}}(p) = \sum_{p_{\vec{d}} \in \mathsf{mint}_p} \mu^{\mathbf{u}}(p_{\vec{d}})$$

where mint_p is the set of minterm implicants of p.

THEOREM 4.1. *$\mu^{\mathbf{u}}$ is a measure on $\mathsf{pol}(A)$.*

Proof. Since $\mu^{\mathbf{u}}$ is clearly non-negative, it remains to be shown that $\mu^{\mathbf{u}}(p_1 \vee p_2) = \mu^{\mathbf{u}}(p_1) + \mu^{\mathbf{u}}(p_2)$ for every $p_1, p_2 \in \mathsf{pol}(A)$ such that $p_1 \wedge p_2 = \mathbf{0}$. Note that if $p_1 \wedge p_2 = \mathbf{0}$ then $\mathsf{mint}_{p_1} \cap \mathsf{mint}_{p_2} = \emptyset$, and

$$\mu^{\mathbf{u}}(p_1 \vee p_2) = \sum_{p_{\vec{d}} \in \mathsf{mint}_{p_1}} \mu^{\mathbf{u}}(p_{\vec{d}}) + \sum_{p_{\vec{d}} \in \mathsf{mint}_{p_2}} \mu^{\mathbf{u}}(p_{\vec{d}}) = \mu^{\mathbf{u}}(p_1) + \mu^{\mathbf{u}}(p_2).$$

□

EXAMPLE 4.1. Let $n = 2$, we have

$$\begin{aligned} &\mu^{\mathbf{u}}(a_1 \oplus a_2) \\ &= \mu^{\mathbf{u}}(\bar{a_1} \wedge a_2) + \mu^{\mathbf{u}}(a_1 \wedge \bar{a_2}) \\ &= \mathtt{supp}_\tau(a_1 = \mathbf{0} \wedge a_2 = \mathbf{1}) + (1-\alpha_1)\mathtt{supp}_\tau(a_1 = \mathbf{u} \wedge a_2 = \mathbf{1}) \\ &+ \alpha_2 \mathtt{supp}_\tau(a_1 = \mathbf{0} \wedge a_2 = \mathbf{u}) + (1-\alpha_1)\alpha_2 \mathtt{supp}_\tau(a_1 = a_2 = \mathbf{u}) \\ &+ \mathtt{supp}_\tau(a_1 = \mathbf{1} \wedge a_2 = \mathbf{0}) + \alpha_1 \mathtt{supp}_\tau(a_1 = \mathbf{u} \wedge a_2 = \mathbf{0}) \\ &+ (1-\alpha_2)\mathtt{supp}_\tau(a_1 = \mathbf{1} \wedge a_2 = \mathbf{u}) + \alpha_1(1-\alpha_2)\mathtt{supp}_\tau(a_1 = a_2 = \mathbf{u}). \end{aligned}$$

□

The benefit of using arbitrary measures instead of probabilities or supports in previous sections is that results on inclusion-exclusion principle automatically apply to $\mu^{\mathbf{u}}$. Also, the fact that $\mu^{\mathbf{u}}$ is a measure make the proof of the following theorem straightforward.

THEOREM 4.2. *For every table $\tau = (T, A, \rho)$ such that $A = \{a_1, \ldots, a_n\}$ and $\mathsf{Dom}(a_i) = \{\mathbf{0}, \mathbf{u}, \mathbf{1}\}$ for $1 \le i \le n$, and every collection of sets of attributes $\mathcal{A} = \{a_{I_1}, \ldots, a_{I_r} \mid I_j \subseteq \{1, \ldots, n\}\}$ there is a probability distribution P over A such that for every $a_{I_r} \in \mathcal{A}$, $P\{\bigwedge_{j \in I_r}(a_j = \mathbf{1})\} = \mu^{\mathbf{u}}(\bigwedge_{j \in I_r}(a_j = \mathbf{1}))/|\rho|$.*

Proof. We will prove the theorem by showing that $\mu^{\mathbf{u}}/|\rho|$ is a probability distribution. Since $\mu^{\mathbf{u}}$ is a measure, it suffices to show that $\mu^{\mathbf{u}}(\mathbf{1}) = |\rho|$. For any $a_i \in A$ we have

$$\begin{aligned} \mu^{\mathbf{u}}(\mathbf{1}) = \mu^{\mathbf{u}}(a_i \vee \bar{a_i}) &= \mu^{\mathbf{u}}(a_i) + \mu^{\mathbf{u}}(\bar{a_i}) \\ &= \mathtt{supp}_\tau(a_i = \mathbf{1}) + \alpha_i \mathtt{supp}_\tau(a_i = \mathbf{u}) \\ &\quad + \mathtt{supp}_\tau(a_i = \mathbf{0}) + (1-\alpha_i)\mathtt{supp}_\tau(a_i = \mathbf{u}) \\ &= \mathtt{supp}_\tau(a_i = \mathbf{1}) + \mathtt{supp}_\tau(a_i = \mathbf{0}) + \mathtt{supp}_\tau(a_i = \mathbf{u}) = |\rho|. \end{aligned}$$

□

The importance of the above theorem is that if we use some datamining algorithm (e.g. Apriori) to find $\mu^{\mathbf{u}}$ for a collection of sets of attributes, then their values of $\mu^{\mathbf{u}}$ are probabilistically consistent.

The previous approaches to frequent itemset mining can be found in [6, 4]. However, both these approaches can produce probabilistically inconsistent results. Specifically, the technique used in [6] is to count the support of an itemset only on the portion of the table where it is valid. For example, consider the table

a_1	a_2
1	**1**
1	**u**
0	**u**
0	**u**

Using the method from [6] the support of attribute a_2 is counted only in the first row, giving $\mathtt{supp}(a_2) = 100\%$. Similarly $\mathtt{supp}(a_1) = 50\%$, and $\mathtt{supp}(a_1a_2) = 100\%$, but this means $\mathtt{supp}(a_1a_2) > \mathtt{supp}(a_1)$, which is impossible. In the method proposed in [4] the probability for each attribute is estimated from the part of the data where the attribute is defined. When computing how much support does a row with a missing value contribute for an itemset, this probabilities are summed for each attribute (see [4] for details). In the table above this will give $\mathtt{supp}(a_1) = 50\%$, $\mathtt{supp}(a_1) = 100\%$, and $\mathtt{supp}(a_1a_2) = [(0.5 \cdot 1 + 0.5 \cdot 1) + (0.5 \cdot 1 + 0.5 \cdot 1) + 2(0.5 \cdot 0 + 0.5 \cdot 1)]/4 = 75\%$, and $\mathtt{supp}(a_1a_2) > \mathtt{supp}(a_2)$. Using our $\mu^{\mathbf{u}}$ measure gives consistent values of $\mathtt{supp}(a_1) = 100\%$, $\mathtt{supp}(a_2) = 50\%$, and $\mathtt{supp}(a_1a_2) = 50\%$.

5. Conclusions and Open Problems. We studied properties of measures defined on free Boolean algebras arising naturally in the evaluation of sizes of queries applied to binary tables in relational databases. We intend to investigate Bonferroni-type inequalities for these measure which will allow evaluations of these measures when information on supports is available only for certain families of itemsets. Another open problem is to ameliorate the bounds already obtained.

REFERENCES

[1] J. Galambos and I. Simonelli, *Bonferroni-type Inequalities with Applications*, Springer, 1996.

[2] H. Manilla, *Combining discrete algorithms and probabilistic approaches in data mining*, in Principles of Data Mining and Knowledge Discovery, L. DeRaedt and A. Siebes, eds., vol. 2168 of Lecture Notes in Artificial Intelligence, Springer-Verlag, Berlin, 2001, p. 493.

[3] H. Mannila and H. Toivonen, *Multiple uses of frequent sets and condensed representations*, in Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD'96), Portland, Oregon, 1996, pp. 189–194.

[4] J. R. Nayak and D. J. Cook, *Approximate association rule mining*, in Proceedings of the Florida Artificial Intelligence Research Symposium, 2001.

[5] D. Pavlov, H. Manilla, and P. Smyth, *Beyond independence: Probabilistic models for query approximation on binary transaction data*, ICS TR-01-09, University of California, Irvine, 2001.

[6] A. Ragel and B. Crémilleux, *Treatment of missing values for association rules*, in Proceedings of the 2nd Pacific-Asia Conference on Research and Development in Knowledge Discovery and Data Mining (PAKDD-98), X. Wu, R. Kotagiri, and K. B. Korb, eds., vol. 1394 of LNAI, Berlin, Apr. 15–17 1998, Springer, pp. 258–270.

[7] S. Rudeanu, *Boolean Functions and Equation*, North-Holland, Amsterdam, 1974.