Itemset Frequency Satisfiability: Complexity and Axiomatization [1]

Toon Calders [a,b]

[a] *Eindhoven University of Technology*
Den Dolech 2, HG 7.82a, P.O. Box 513,
5600 MB Eindhoven, The Netherlands

[b] *University of Antwerp, Belgium*

Abstract

Computing frequent itemsets is one of the most prominent problems in data mining. We study the following related problem, called `FREQSAT`, in depth: given some itemset-interval pairs, does there exist a database such that for every pair the frequency of the itemset falls into the interval? This problem is shown to be **NP**-complete. The problem is then further extended to include arbitrary Boolean expressions over items and conditional frequency expressions in the form of association rules. We also show that, unless **P** equals **NP**, the related function problem—find the best interval for an itemset under some frequency constraints—cannot be approximated efficiently. Furthermore, it is shown that `FREQSAT` is recursively axiomatizable, but that there cannot exist an axiomatization of finite arity.

Key words: Data Mining, Frequent Itemset, Complexity

1 Introduction

The *frequent itemset mining problem* [3] is one of the core problems in data mining. We are given a database $\mathcal{D}$ of sets, called *transactions*, and a threshold *minfreq*. The *frequency* of a set I in $\mathcal{D}$ is the number of transactions in $\mathcal{D}$ that contain all items of I divided by the total number of transactions in $\mathcal{D}$. The frequent itemset problem is to compute all sets I such that the frequency of I in $\mathcal{D}$ is at least *minfreq*. The most important application of frequent itemsets is forming the so-called association rules [3]. An association rule is an implication of the form $I \rightarrow J$, where I and J are itemsets. The strength of an association

[1] Parts of the reported results were published in the extended abstract [12].

rule is expressed by its support, i.e., the number of transactions in which I and J are both present, and its confidence, i.e., the conditional probability that a transaction contains J given that it contains I. Both support and confidence of an association rule can be obtained from the frequency of I and $I \cup J$. Association analysis has been applied and shown to be useful in many domains, such as web mining, document analysis, telecommunication alarm diagnosis, bio-informatics, etc. Association rule and frequent itemset mining form also often the basis of other algorithms for classification, regression, and clustering. For an overview of relevant references to applications of association analysis, see [43, Chapter 6].

During the last decade, many algorithms to solve this problem were developed. For an overview, see [8, 24, 29]. All these frequent itemset mining algorithms rely heavily on the monotonicity of frequency: if $I \subseteq J$, then the frequency of J is bounded from above by the frequency of I. In general, this property of frequency allows for pruning substantial parts of the search space. Besides monotonicity, also other relationships between the frequencies can be identified. For example, in the MAXMINER algorithm [7], relations of the following form are exploited: $\mathit{freq}(\{a,b,c\}) \geq \mathit{freq}(\{a,b\}) + \mathit{freq}(\{a,c\}) - \mathit{freq}(\{a\})$. There are many more relations between the frequencies of itemsets. See [13] for extensions based on the inclusion-exclusion principle. For a generalization to other measures besides frequency, see [42].

FREQSAT. The relationships between the frequencies of itemsets can be seen as consistency constraints; only configurations of frequencies that satisfy these relationships, represent valid outcomes of frequent itemset mining. In this context, we introduce the problem FREQSAT: given a collection of expressions $\mathit{freq}(I) \in [l,u]$, does there exist a transaction database that satisfies them? For example, $\{\mathit{freq}(\{a\}) \in [0,0.5], \mathit{freq}(\{a,b\}) \in [0.6,1]\}$ is not satisfiable, because of the monotonicity of frequency.

This paper concentrates on the properties of FREQSAT. The results can roughly be divided in three classes: the first type of results concerns the robustness of the FREQSAT-problem: what is the influence if we replace the intervals in the definition by single points? What if we allow arbitrary Boolean formulas or association rules instead of simple itemsets? The second type of results concerns the complexity of FREQSAT and the deduction of frequency constraints. What is the complexity of FREQSAT and related (function) problems? Is there an axiomatization for the deduction of frequencies? The third type concerns a negative approximation result.

Equivalence with pSAT. We show that FREQSAT is equivalent to *probabilistic satisfiability* (pSAT) [37]. pSAT is the problem of deciding if, given set of Boolean formulas with probabilities, there exists a probability distribution that assigns to for every given formula the given probability. The reduction from FREQSAT

to pSAT is quite straightforward; a transaction database can be considered as a probability distribution and the frequency of an itemset as the probability of the conjunction of the items in it.

The reduction from pSAT to FREQSAT, however, is more surprising, as it shows that even with simple itemsets we can express frequency constraints on arbitrary Boolean formulas. That is, in the probabilistic version of logical satisfiability, conjunctive formulas are as powerful as arbitrary Boolean formulas with negation and disjunction. Because pSAT is **NP**-complete [23], the equivalence of the two problems shows at the same time that FREQSAT is **NP**-complete as well.

Association Rules. We also show that in FREQSAT we are able to express the confidence of association rules. This equivalence links FREQSAT to *probabilistic logic programming with conditional constraints*, which was studied, e.g., by Lukasiewicz [34].

Furthermore, from the fact that we can simulate satisfiability of arbitrary Boolean formulas and conditional constraints with FREQSAT, we can easily construct sets of frequency constraints such that the interval of possible frequencies for a given itemset is either $[0, 0]$, or $[0, 0.5]$, and it is **NP**-complete to decide which one of the two is the case. Therefore, it is not possible to efficiently approximate the upper bound on the frequency of an itemset, given a set of frequency constraints (unless **P** equals **NP**). As such, the entailed interval cannot be approximated efficiently.

Axiomatization. We prove that there cannot exist a complete set of deduction rules with finite schema that axiomatizes FREQSAT. That is, there does not exist a number n such that FREQSAT can be axiomatized with rules "**if** R **then** ρ", where R contains at most n parameterized frequency constraints. Hence, there are infinitely many non-redundant relations between frequencies. We do show, however, that FREQSAT is recursively axiomatizable, and that we can always find *locally* complete axioms. That is, if we fix some sets $I_1, \ldots, I_m$, and a target set I, we can give a sound and complete axiomatization for the deduction

$$\{\mathit{freq}\,(I_1) \in [l_1, u_1], \ldots, \mathit{freq}\,(I_1) \in [l_m, u_m]\} \models \mathit{freq}\,(I) \in [l, u] \ ,$$

with $l, u, l_1, u_1, \ldots, l_m, u_m$ being parameters. For example, for the sets $\{a\}$, $\{b\}$, and the target $\{a, b\}$, a sound and complete axiom is:

$$\mathit{freq}\,(\{a\}) \in [l_a, u_a], \mathit{freq}\,(\{b\}) \in [l_b, u_b] \ \vdash \\ \mathit{freq}\,(\{a, b\}) \in [\max\{0, l_a + l_b - 1\}, \min\{u_a, u_b, 1\}] \ .$$

Organization. In Section 2, we formally introduce important notions, and we

define the problems studied in the paper. In Section 3, the equivalence with pSAT is shown, and the implications for the complexity of FREQSAT is studied. In Section 4, we show how association rules can be expressed in FREQSAT. In Section 5, the axiomatization of FREQSAT is discussed in detail. Section 6.1 describes related work, and Section 7 summarizes the most important results and concludes the paper.

2 Preliminaries

In this section we formalize the problem statement as the FREQSAT-*problem*.

2.1 Itemsets

Let $\mathcal{I}$ be a finite set, called the set of items. A *transaction* over $\mathcal{I}$ is a pair (tid, J), with tid an identifier, and J a subset of $\mathcal{I}$. A *transaction database* over $\mathcal{I}$ is a finite set of such transactions where no two transactions have the same identifier. In the following, we assume that the transaction identifiers are strictly positive integers. Hence, a transaction is a pair (tid, I), with $tid \in \{1, 2, 3, \ldots\}$, and $I \subseteq \mathcal{I}$.

Let I be some set of items. We say that the transaction (tid, J) *contains* I, denoted $I \subseteq (tid, J)$, if $I \subseteq J$.

The *support* of I in $\mathcal{D}$, denoted $supp(I, \mathcal{D})$, is the absolute number of transactions in $\mathcal{D}$ that contain I. The *frequency* of I in $\mathcal{D}$, denoted $freq\,(I, \mathcal{D})$, is $supp(I, \mathcal{D})$ divided by the number of transactions in $\mathcal{D}$. In all what follows, $\mathcal{D}$ is a transaction database over $\mathcal{I}$.

Example 1 *Consider the following transaction database, with the frequencies of some sets:*

$\mathcal{D}$ =

TID	Items
1	a, b
2	a, c
3	c
4	a, b, c

$freq\,(\{a\}) = 0.75$

$freq\,(\{b\}) = 0.5$

$freq\,(\{c\}) = 0.75$

$freq\,(\{a, b\}) = 0.5$

$freq\,(\{a, b, c\}) = 0.25$

A *Frequency Constraint* is an expression $freq\,(I) \in [l, u]$, with I an itemset, and $0 \leq l, u \leq 1$ rational numbers. We say that $\mathcal{D}$ *satisfies* this expression, denoted $\mathcal{D} \models freq\,(I) \in [l, u]$, if the frequency of I in $\mathcal{D}$ is in the interval $[l, u]$. $\mathcal{D}$ satisfies a set of frequency constraints, if it satisfies all of them.

A set of frequency constraints $\mathcal{C}$ *entails* a constraint $freq\,(I) \in [l, u]$, denoted $\mathcal{C} \models freq\,(I) \in [l, u]$, if every database $\mathcal{D}$ that satisfies $\mathcal{C}$, satisfies $freq\,(I) \in [l, u]$ as well. The entailment is said to be *tight*, denoted $\mathcal{C} \models_{tight} freq\,(I) \in [l, u]$, if for every smaller interval $[l', u'] \subset [l, u]$, $\mathcal{C}$ does not entail $freq\,(I) \in [l', u']$. That is, if $[l, u]$ is the best interval derivable for I, based on $\mathcal{C}$.

We often use $freq\,(I) = f$ to denote $freq\,(I) \in [f, f]$.

Example 2 *Consider the following set of frequency constraints:*

$$\mathcal{C} = \left\{ \begin{array}{ll} freq\,(\{a\}) \in [0.75, 1], & freq\,(\{b\}) \in [0.5, 0.75], \\ freq\,(\{c\}) = 0.75, & freq\,(\{a, b\}) = 0.5 \end{array} \right\} .$$

This set of constraints is satisfied by the database $\mathcal{D}$ given in Example 1.

The constraint $freq\,(\{a, b, c\}) = 0.5$ is not entailed by the constraints in $\mathcal{C}$. The database $\mathcal{D}$ in Example 1 is a counter example; $\mathcal{D}$ satisfies $\mathcal{C}$, but does not satisfy $freq\,(\{a, b, c\}) = 0.5$.

The constraint $freq\,(\{a, b, c\}) \in [0, 0.5]$ is entailed by $\mathcal{C}$. Indeed, because of the monotonicity of frequency, the frequency of $\{a, b, c\}$ must always be less than the frequency of $\{a, b\}$. Therefore, in any database that satisfies $freq\,(\{a, b\}) = 0.5$, the frequency of $\{a, b, c\}$ will be less than 0.5. The entailment is not tight, however, because the interval $[0, 0.5]$ can be made even smaller; in every database that satisfies $\mathcal{C}$, the frequency of $\{a, b, c\}$ must be at least 0.25. This can be seen as follows: because of the constraints $freq\,(\{c\}) = 0.75$ and $freq\,(\{a, b\}) = 0.5$, 75% of the transactions of a satisfying database for $\mathcal{C}$ contain $\{c\}$, and 50% contain $\{a, b\}$. Therefore, there must be an overlap of at least 25% transactions that contain both $\{a, b\}$ and $\{c\}$.

The entailed interval $[0.25, 0.5]$ for $\{a, b, c\}$ given $\mathcal{C}$ is tight. We can prove this by showing, with examples, that the lower and upper bound are indeed both feasible. The tightness of the lower bound follows from the database given in Example 1. For the upper bound, the following database shows the tightness:

$$\{(1, \{a, b, c\}), (2, \{a, b, c\}), (3, \{a, c\}), (4, \{b\})\} \ .$$

We are now ready to state the main problem studied in this paper: the FREQSAT-problem.

Problem FREQSAT:
Input: A set of frequency constraints $\mathcal{C} = \{freq\,(I_j) \in [l_j, u_j], j = 1 \ldots m\}$
Accept: iff there exists a database $\mathcal{D}$ over $\bigcup_{j=1}^{m} I_j$ that satisfies $\mathcal{C}$. □

Example 3 *Suppose that the following set $\mathcal{C}$ of frequency constraints $\mathcal{C}$ is given:*

$$\left\{\begin{array}{lll} freq\,(\{a,b\}) \in [3/4,1], & freq\,(\{a,c\}) \in [3/4,1], & freq\,(\{b,c\}) \in [3/4,1], \\ freq\,(\{d,e\}) \in [3/4,1], & freq\,(\{d,f\}) \in [1/2,1], & freq\,(\{e,f\}) \in [1/2,1], \\ freq\,(\{a,b,c,d,e,f\}) = 0 & & \end{array}\right\}$$

$\mathcal{C}$ is in FREQSAT*, because it is satisfiable by the following database:*

$\mathcal{D}$ =

TID	Items
1	a, b, c, d, e
2	a, b, c, d, e
3	a, b, c, d, e
4	$a, b, c, d, \quad f$

TID	Items
5	$a, b, c, \quad e, f$
6	$a, b, \quad d, e, f$
7	$a, \quad c, d, e, f$
8	b, c, d, e, f

Notice incidentally that the FREQSAT-instance in the above example illustrates that the *relative* frequencies in the definition of FREQSAT cannot be replaced straightforwardly by *absolute* support; even though all bounds on the frequencies can be written as a multiple of $1/4$, there does not exist a satisfying database with 4 transactions. To prove that such a satisfying database with 4 transactions cannot exist, it suffices to notice that from $\mathcal{C}$ it follows that $freq\,(\{a,b,c\}) \in [5/8, 1]$, and $freq\,(\{d,e,f\}) \in [3/8, 1]$. This is because in every transaction database, the following relation between the frequencies holds [16]:

$$freq\,(\{x,y,z\}) \geq \frac{freq\,(\{x,y\}) + freq\,(\{x,z\}) + freq\,(\{y,z\}) - 1}{2} \ .$$

From $freq\,(\{a,b,c,d,e,f\}) = 0$ we can conclude that $\{a,b,c\}$ and $\{d,e,f\}$ cannot be in the same transaction. All these observations combined lead to the conclusion that $freq\,(\{d,e,f\})$ must be $3/8$, and $freq\,(\{a,b,c\}) = 5/8$, as otherwise they would overlap.

3 The Computational Complexity of FREQSAT

In this section we study the complexity of FREQSAT. We start by showing that in the definition of FREQSAT, the intervals can be replaced by exact frequencies while keeping the same expressibility. Thus, this simplification of the problem does not change the properties of the problem, nor the complexity. Then we prove that in FREQSAT we can express the frequency of arbitrary Boolean formulas over items. This shows that FREQSAT is equivalent to *probabilistic satisfiability* (pSAT) [37]. The implications of this relation with pSAT are then discussed.

3.1 Replacing Intervals with Single Numbers

First of all, we show that in a FREQSAT-problem, we can replace intervals with exact frequencies. That is, we can reduce arbitrary FREQSAT-problems to FREQSAT-problems that only contain constraints of type $freq\,(I) = f$. Let $\mathcal{C} = \{freq\,(I_j) \in [l_j, u_j], j = 1, \ldots, m\}$ be a FREQSAT-instance. Let $\mathcal{I} = \bigcup_{j=1}^{m} I_j$, and let $a_1, \ldots, a_m$, $b_1, \ldots, b_m$ be $2m$ items not in $\mathcal{I}$. $\mathcal{EQ}(\mathcal{C})$ now denotes the following set of frequency constraints:

$$\mathcal{EQ}(\mathcal{C}) \;=\; \bigcup_{j=1}^{m} \{freq\,(I_j \cup \{a_j\}) = l_j, freq\,(\{b_j\}) = 1 - u_j, freq\,(I_j \cup \{b_j\}) = 0\} \;\; .$$

We are now ready to state and prove the main theorem of this subsection:

Theorem 1 *$\mathcal{C} = \{freq\,(I_j) \in [l_j, u_j], j = 1, \ldots, m\}$ is in* FREQSAT *if and only if $\mathcal{EQ}(\mathcal{C})$ is.*

Proof The proof is based on the following simple observation: if a set I that has a frequency between l and u in a database $\mathcal{D}$, then there exist at least a fraction l of transactions that contains I, and a fraction $1-u$ that doesn't. The idea is to "mark" *exactly* l of the transactions containing I by adding an item a, and $1 - u$ transactions that do not contain I with item b. The existence of such items a and b with respectively frequencies l and $1 - u$ implies therefore that the frequency of I is between l and u. There is, however, one problem: if the denominator (let's say k) of l is not a divisor of $|\mathcal{D}|$, it is not possible to add a to *exactly* a fraction l of the transactions. Luckily, this problem is easily solved by constructing a new database $\bigoplus_k \mathcal{D}$ that consists of k copies of every transaction in $\mathcal{D}$. This database has the same frequencies for its itemsets, and the number of transactions in it is a multiple of k. The full proof of this theorem can be found in Appendix A. □

3.2 Probabilistic Satisfiability

Boolean formulas, truth assignments and valuations are defined as usual: let $x_1, \ldots, x_n$ be Boolean variables. A truth assignment over $x_1, \ldots, x_n$ is a function from $\{x_1, \ldots, x_n\}$ to $\{0, 1\}$. The set of all 2^n truth assignments over $x_1, \ldots, x_n$ is denoted $\mathcal{A}(x_1, \ldots, x_n)$. The tuple of variables $(x_1, \ldots, x_n)$ is omitted when clear from the context. A *probability distribution over* $x_1, \ldots, x_n$ is a function Π that maps every assignment in $\mathcal{A}(x_1, \ldots, x_n)$ to a real number between 0 and 1 such that $\sum_{A \in \mathcal{A}} \Pi(A) = 1$.

Let φ be a Boolean formula over the variables $x_1, \ldots, x_n$, and let Π be a probability distribution over these variables. The probability of φ given Π is defined as:

$$\mathrm{Prob}_\Pi(\varphi) = \sum_{A \in \mathcal{A}} A(\varphi) \cdot \Pi(A) \ .$$

That is, $\mathrm{Prob}_\Pi(\varphi)$ is the sum of $\Pi(A)$ over all assignments A that make φ true.

The probabilistic satisfiability problem (`pSAT`) [37] is defined as follows: *Consider m logical sentences $\varphi_1, \ldots, \varphi_m$ over the variables $x_1, \ldots, x_n$ with the usual Boolean operators $\neg$, $\vee$, $\wedge$. Assume (rational) probabilities $\pi_1, \ldots, \pi_m$ for these sentences to be true are given. Does there exist a probability distribution Π over $x_1, \ldots, x_n$ such that for all $j = 1 \ldots m$, $\mathrm{Prob}_\Pi(\varphi_j) = \pi_j$?*

In [23], it is proven that `pSAT` is **NP**-complete. This proof relies on the fact that if a `pSAT`-problem is satisfiable, then it is satisfiable by a distribution Π that can be represented in a succinct way as follows: for all assignments $A \in \mathcal{A}$, $\Pi(A)$ is a rational number with length polynomial in the length of the input $(\varphi_1, \ldots, \varphi_m, \pi_1, \ldots, \pi_m)$, and there are at most $m+1$ truth assignments A such that $\Pi(A) \neq 0$. Hence, the listing of those assignments A, together with their probabilities $\Pi(A)$ is a succinct certificate.

3.2.1 Reduction From `FREQSAT` *to* `pSAT`

Because of Theorem 1, we can—without loss of generality—consider only `FREQSAT`-problems where exact frequencies are given; we can always reduce $\mathcal{C}$ to $\mathcal{EQ}(\mathcal{C})$ as a first step.

There is a straightforward relation between `FREQSAT` and `pSAT`; every instance of the `FREQSAT`-problem can be seen as an instance of `pSAT` in which only conjunctions are used. Let $\mathcal{EQ}(\mathcal{C}) = \{\mathit{freq}\,(I_j) = f_j, j = 1, \ldots, m\}$ be a `FREQSAT`-problem. Let $\mathcal{I} = \bigcup_{j=1}^{m} I_j$. Associate with every $i \in \mathcal{I}$, a variable x_i. $\mathcal{PSAT}(\mathcal{C})$

denotes the following pSAT-problem:

$$\varphi_j = \left(\bigwedge_{i \in I_j} x_i \right), \quad \pi_j = f_j, \quad j = 1 \ldots m$$

Theorem 2 *$\mathcal{C}$ is in* FREQSAT *if and only if $\mathcal{PSAT}(\mathcal{C})$ is in* pSAT*.*

Proof The proof can be found in Appendix B. □

3.2.2 Reduction From pSAT *to* FREQSAT

We can extend FREQSAT to include constraints over arbitrary Boolean formulas. An *extended frequency constraint* is an expression $freq\,(\varphi) \in [l, u]$, with φ a Boolean formula over the set of items $\mathcal{I}$. We say that a transaction (tid, J) *satisfies* φ, if the truth assignment V that assigns 1 to an item i if and only if $i \in J$, makes φ true. E.g., the transaction $(tid, \{a, b, c\})$ satisfies $a \vee (b \wedge \neg c)$, but does not satisfy $a \wedge \neg c$. The *frequency* of a Boolean formula is the number of transactions satisfying it. The satisfaction and entailment of extended frequency constraint are defined in the same way as for frequency constraints. The extension of FREQSAT to arbitrary Boolean formulas gives pSAT.

Lemma 1 *Let $\Pi = (\varphi_1, \ldots, \varphi_m, \pi_1, \ldots, \pi_m)$ be a* pSAT*-problem with variables taken from the set $\mathcal{I}$. Π is satisfiable if and only if the following extended* FREQSAT*-problem is: $\{freq\,(\varphi_1) \in [\pi_1, \pi_1], \ldots, freq\,(\varphi_m) \in [\pi_m, \pi_m]\}$.*

We now show that in FREQSAT we can simulate extended frequency constraints.

Intuition behind the proof. For every subformula σ of the formulas φ_1, $\ldots$, φ_m in the extended FREQSAT problem, (also for the items), we introduce two new items, t_σ and f_σ. t_σ stands for "σ is true in this transaction", and f_σ for "σ is false in this transaction". A transaction $T = (tid, J)$ will represent the truth assignment V_T that assigns true to all items i with $t_i \in J$, and false to the items j with $f_j \in J$. We add constraints such that t_σ is in a transaction T if and only if the truth assignment V_T makes σ true. For example, suppose that we have one formula $a \vee b$. The transaction consisting of the items $\{a, c\}$ will actually be represented as $(tid, \{a, c, t_a, f_b, t_{a \vee b}\})$. The reduction will be such that there are constraints that enforce that the "special" items t_a, t_b, f_a, f_b, $t_{a \vee b}$, $f_{a \vee b}$ be consistent with the "regular" items a, b, and c. Notice that there is, e.g., no item t_c, because c does not occur as a subformula of $a \vee b$. Notice that the number of subformulas of a Boolean formula φ is linear in the size of φ, as there is one subformula for every variable and for every connector used in φ. This linearity is important in showing that the reduction is polynomial.

The main trick used in the reduction is that only half of the transactions will represent valid truth assignments. These transactions will contain the item d, the others contain item $\overline{d}$ (hence, $\overline{d}$ is in fact *not* d):

$$\mathit{freq}\left(\{d\}\right) = 0.5, \qquad \mathit{freq}\left(\{\overline{d}\}\right) = 0.5, \qquad \mathit{freq}\left(\{d, \overline{d}\}\right) = 0 \ .$$

For every subexpression σ, we add the following constraints:

$$\mathit{freq}\left(\{t_\sigma\}\right) = 0.5, \qquad \mathit{freq}\left(\{f_\sigma\}\right) = 0.5, \qquad \mathit{freq}\left(\{t_\sigma, f_\sigma\}\right) = 0 \ .$$

In this way, we make sure that every transaction contains either t_σ, or f_σ, but not both. We use the transactions containing $\overline{d}$ to compensate the fact that we cannot know (at least not without solving an **NP**-complete problem) how many transactions will have t_σ (resp. f_σ). For example, for $a \vee \neg a$, half of the transactions will contain $\{d, t_{a\vee\neg a}\}$, and the other halve contains $\{\overline{d}, f_{a\vee\neg a}\}$. Hence, even though only *half* of the transactions contain $t_{a\vee\neg a}$, *all* transactions representing *valid* truth assignments contain $t_{a\vee\neg a}$.

We still have to make sure that within the d-part of a satisfying database, the trues and falses are consistent with each other. For example, a transaction validly representing a truth assignment cannot contain $t_{a\vee b}$, f_a, and f_b at the same time. The consistency can easily be enforced by adding some simple frequency constraints. For example, for a disjunction $\sigma_1 \vee \sigma_2$ it suffices to add the following three constraints:

$$\mathit{freq}\left(\{d, t_{\sigma_1\vee\sigma_2}, f_{\sigma_1}, f_{\sigma_2}\}\right) = 0, \mathit{freq}\left(\{d, f_{\sigma_1\vee\sigma_2}, t_{\sigma_1}\right) = 0, \mathit{freq}\left(\{d, f_{\sigma_1\vee\sigma_2}, t_{\sigma_2}\right) = 0$$

Finally, for all $j = 1 \ldots m$, we add the constraint $\{\mathit{freq}\left(\{d, t_{\varphi_j}\}\right) \in [l/2, u/2]\}$.

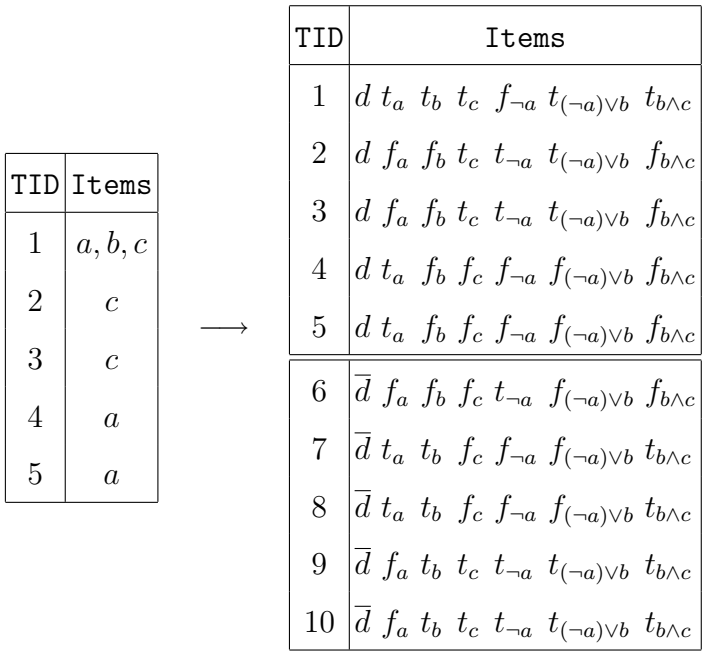
Example 4 *Consider the following set of extended frequency constraints $\mathcal{P}$:*

$$\mathcal{P} \ = \ \left\{ \begin{array}{ll} \mathit{freq}\left(a\right) \in [0.4, 0.7], & \mathit{freq}\left((\neg a) \vee b\right) = 0.6, \\ \mathit{freq}\left(b \wedge c\right) \in [0.2, 0.4], & \mathit{freq}\left(c\right) = 0.6 \end{array} \right\} \ .$$

$\mathcal{FSAT}(\mathcal{P})$ is a set of frequency constraints over the items

$$\{t_a,\ f_a,\ t_b,\ f_b,\ t_c,\ f_c,\ t_{\neg a},\ f_{\neg a},\ t_{(\neg a)\vee b},\ f_{(\neg a)\vee b},\ t_{b\wedge c},\ f_{b\wedge c},\ d,\ \overline{d}\} \ .$$

The first type of constraints in $\mathcal{FSAT}(\mathcal{P})$ makes sure that t_σ and f_σ are com-



$$\mathcal{P} \;=\; \left\{ \begin{array}{ll} \mathit{freq}\,(a) \in [0.4, 0.7], & \mathit{freq}\,((\neg a) \vee b) = 0.6, \\ \mathit{freq}\,(b \wedge c) \in [0.2, 0.4], & \mathit{freq}\,(c) = 0.6 \end{array} \right\} \;.$$

Fig. 1. A database satisfying $\mathcal{P}$ and a corresponding database for $\mathcal{FSAT}(\mathcal{P})$.

plements of each other:

$$\begin{array}{lll} \mathit{freq}\,(\{t_a, f_a\}) = 0, & \mathit{freq}\,(\{t_a\}) = 0.5, & \mathit{freq}\,(\{f_a\}) = 0.5 \\ \mathit{freq}\,(\{t_b, f_b\}) = 0, & \mathit{freq}\,(\{t_b\}) = 0.5, & \mathit{freq}\,(\{f_b\}) = 0.5 \\ & \ldots & \\ \mathit{freq}\,(\{t_{b\wedge c}, f_{b\wedge c}\}) = 0, & \mathit{freq}\,(\{t_{b\wedge c}\}) = 0.5, & \mathit{freq}\,(\{f_{b\wedge c}\}) = 0.5 \end{array}$$

The item d is in half of the transactions, and $\overline{d}$ is its complement:

$$\mathit{freq}\left(\{d, \overline{d}\}\right) = 0, \quad \mathit{freq}\,(\{d\}) = 0.5, \quad \mathit{freq}\left(\{\overline{d}\}\right) = 0.5 \;.$$

The second type of constraints makes sure that within the transactions that contain d of a satisfying database, the trues and falses are consistent:

- $\mathit{freq}\,(\{d, t_a, t_{\neg a}\}) = 0$, $\mathit{freq}\,(\{d, f_a, f_{\neg a}\}) = 0$
- $\mathit{freq}\left(\{d, t_{\neg a}, f_{(\neg a)\vee b}\}\right) = 0$, $\mathit{freq}\left(\{d, t_b, f_{(\neg a)\vee b}\}\right) = 0$, $\mathit{freq}\left(\{d, f_{\neg a}, f_b, t_{(\neg a)\vee b}\}\right) = 0$
- $\mathit{freq}\,(\{d, f_b, t_{b\wedge c}\}) = 0$, $\mathit{freq}\,(\{d, f_c, t_{b\wedge c}\}) = 0$, $\mathit{freq}\,(\{d, t_a, t_b, f_{b\wedge c}\}) = 0$

Finally, the third type of constraints translates the extended frequency con-

straints:

$$freq\,(\{d, t_a\}) \in [0.2, 0.35],\ freq\left(\{d, t_{(\neg a)\vee b}\}\right) = 0.3,$$
$$freq\,(d, t_{b\wedge c}) \in [0.1, 0.2], \quad freq\,(\{d, t_c\}) = 0.3$$

In Figure 1, two databases satisfying respectively $\mathcal{P}$ and $\mathcal{FSAT}(\mathcal{P})$ have been given.

The complete formal construction of the reduction $\mathcal{FSAT}$ can be found in appendix C.

We are now almost ready to state the main result of this section, namely that FREQSAT is equivalent to extended FREQSAT, and thus also equivalent to pSAT. We also want to relate the set of possible frequencies of an expression φ in an extended FREQSAT-problem $\mathcal{P}$, and the possible frequencies of the itemset $\{t_\varphi, d\}$ in $\mathcal{FSAT}(\mathcal{P})$. Therefore, we first introduce the *entailed frequencies*.

Definition 1 *Let $\mathcal{C}$ be a* FREQSAT*-problem, and let $\mathcal{P}$ be a* pSAT*-problem. I is an itemset, and φ is a Boolean formula.*

$$ENT_I(\mathcal{C}) \ := \ \{freq\,(I, \mathcal{D}) \mid \mathcal{D} \models \mathcal{C}\}$$
$$ENT_\varphi(\mathcal{P}) \ := \ \{\mathrm{Prob}_\Pi(\varphi) \mid \Pi \text{ is a solution of } \mathcal{P}\}$$

Theorem 3 *$\mathcal{P} = (\varphi_1, \ldots, \varphi_m, \pi_1, \ldots, \pi_m)$ is in* pSAT *if and only if $\mathcal{FSAT}(\mathcal{P})$ is in* FREQSAT*.*

Furthermore, $ENT_\varphi(\mathcal{P}) = [l, u]$, iff $ENT_{\{d,t_\varphi\}}(\mathcal{FSAT}(\mathcal{P})) = [l/2, u/2]$.

Proof The proof of this theorem can be found in Appendix C. □

3.3 Implications of the Equivalence between pSAT *and* FREQSAT

In [23], it was shown that pSAT is **NP**-complete. Therefore, the equivalence of pSAT and FREQSAT leads to the following corollary.

Corollary 1 FREQSAT *is* **NP***-complete.*

Notice that there is also a more direct proof possible of the **NP**-completeness of FREQSAT [11], along the lines of the proof in [23]. We do, however, prefer the proof via the reduction from pSAT, because of the fact that we can simulate arbitrary Boolean formulas in FREQSAT will be very important in the rest of the paper. For a direct proof, see [11].

The proof of **NP**-completeness of pSAT in [23], relies heavily on the following property. If a satisfying probability distribution for an instance of pSAT

exists, then there is one with at most $m+1$ non-zero probabilities, and with entries rational numbers with total precision $\mathcal{O}(m^2)$. (m denotes the number of Boolean formulas.) Also this result can be extended to FREQSAT.

Corollary 2 *If there exists a satisfying database for an instance $\mathcal{C}$ of the* FREQSAT- *problem, then there exists a database $\mathcal{D}$ such that $|\{J \mid (tid, J) \in \mathcal{D}\}|$ is at most $3m+1$, and the number of transactions is at most $2^{p(m)}$. ($p(m)$ is a fixed polynomial, independent of $\mathcal{C}$.)*

Proof This follows from results in [23], and the construction in Theorem 2. A satisfiable FREQSAT-instance $\mathcal{C}$ with m frequency constraints is reduced to a satisfiable pSAT-instance $\mathcal{P}(\mathcal{C})$ with $3m$ Boolean formulas (first $\mathcal{EQ}$ is applied in order to eliminate the intervals; the application of $\mathcal{EQ}$ results in a new FREQSAT-instance having $3m$ constraints.) Because the results in [23], there exists a probability distribution Π that satisfies $\mathcal{P}SAT(\mathcal{C})$ with at most $3m+1$ non-zero probabilities, and with entries rational numbers with total precision $\mathcal{O}(m^2)$. The database $\mathcal{D}_\Pi$ from the proof of Theorem 2 is the desired satisfying database for $\mathcal{C}$. □

In [34], different decision and function problems related to pSAT were introduced. We can do similarly for FREQSAT. Consider the following three entailment problems associated with FREQSAT:
(1) FREQENT($\mathcal{C}$, *freq* $(I) \in [l,u]$): Decide whether $\mathcal{C} \models$ *freq* $(I) \in [l,u]$.
(2) T-FREQENT($\mathcal{C}$, *freq* $(I) \in [l,u]$): Decide whether $\mathcal{C} \models_{tight}$ *freq* $(I) \in [l,u]$.
(3) Func T-FREQENT($\mathcal{C}$, I): Give $[l,u]$ such that $\mathcal{C} \models_{tight}$ *freq* $(I) \in [l,u]$.

The complexity of these three problems is very related to the complexity of FREQSAT. Since FREQSAT and pSAT are equivalent, we can directly use the results of Lukasiewicz [34]. Hence, we obtain the following corollary:

Corollary 3 FREQENT *is* **co-NP**-*complete,* T-FREQENT *is* **DP**-*complete, and* Func T-FREQENT *is* $\mathbf{FP}^{\mathbf{NP}}$-*complete.*

Finally, it is well-known that for the pSAT-problem, the entailed sets are always intervals. This is due to the fact that a pSAT entailment problem can be restated as an optimization problem that amounts to minimizing and maximizing a linear programme. In a similar way, it can be shown that for all itemsets I, and FREQSAT-instances $\mathcal{C}$, $ENT_I(\mathcal{C})$ is an interval, with a rational lower and upper bound with precision polynomial in the sizes of $\mathcal{C}$ and I.

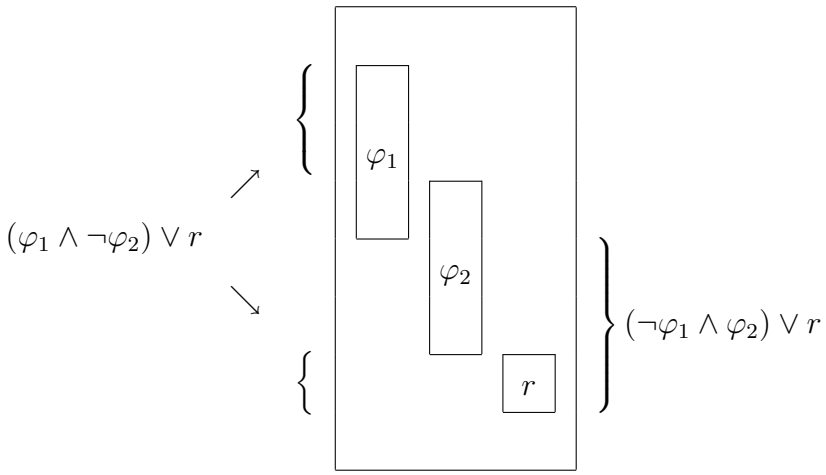


Fig. 2. Construction of $\epsilon(\varphi_1, \varphi_2)$

4 Simulating Association Rules

In this section we show that the confidence of association rules can be expressed with FREQSAT. A key construction herein is the *Multiplication Lemma*. This lemma illustrates that we can express constraints like $\mathit{freq}\,(\varphi) = 2 \cdot \mathit{freq}\,(\psi)$. From this lemma, the ability to express that a certain association rule must have confidence in a given interval is immediate.

Definition 2 *An* association constraint *is an expression* $\mathit{conf}\,(I \to J) \in [l, u]$, *with* I, J *itemsets. A database* $\mathcal{D}$ *satisfies this association constraint if and only if* $l \cdot \mathit{freq}\,(I) \leq \mathit{freq}\,(I \cup J) \leq u \cdot \mathit{freq}\,(I)$.

Notice that this definition implies that if the frequency of I is 0, then the association constraint $\mathit{conf}\,(I \to J) \in [l, u]$ is satisfied.

4.1 Multiplication Lemma

This Multiplication Lemma shows how we can construct a set of constraints such that a new item m is forced to have exactly d times the frequency of a given itemset I, for a given d.

One of the main constructions in this section is the following expression $\epsilon(\varphi_1, \varphi_2)$, that enforces that two Boolean formulas φ_1 and φ_2 have the ex-

act same frequency (ϵ of equals):

$$\begin{aligned}\epsilon(\varphi_1, \varphi_2) \;:=\; \{\, &\mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2 \wedge r) = 0,\, \mathit{freq}\,((\varphi_1 \wedge \neg\varphi_2) \vee r) = 0.5, \\ &\mathit{freq}\,(\neg\varphi_1 \wedge \varphi_2 \wedge r) = 0,\, \mathit{freq}\,((\neg\varphi_1 \wedge \varphi_2) \vee r) = 0.5 \;\}\end{aligned}$$

This construction of ϵ is illustrated in Figure 2. It is based on the fact that φ_1 and φ_2 have the same frequency if and only if $\varphi_1 \wedge \neg\varphi_2$ and $\neg\varphi_1 \wedge \varphi_2$ have the exact same frequency. Furthermore, because these two expressions cannot be true at the same time, if they have the same frequency, it is at most 0.5. Therefore, we can add a new item r such that r is in no transaction that satisfies either $\varphi_1 \wedge \neg\varphi_2$, or $\neg\varphi_1 \wedge \varphi_2$, and the frequency of r is exactly the difference between 0.5 and the frequencies of $\varphi_1 \wedge \neg\varphi_2$ and $\neg\varphi_1 \wedge \varphi_2$.

We often use more than one ϵ-expression at the same time. We assume implicitly that for each use of ϵ, a new item is substituted for r. That is, if we use, e.g., the set of constraints $\mathcal{C} \cup \epsilon(\varphi_1, \varphi_2) \cup \epsilon(\varphi_3, \varphi_4)$, we implicitly assume that the item r in $\epsilon(\varphi_1, \varphi_2)$ differs from the one used in $\epsilon(\varphi_3, \varphi_4)$.

Using multiple ϵ-expressions, we construct the following expression $\delta(\varphi_1, \varphi_2)$ enforcing that the frequency of φ_2 is exactly twice the frequency of φ_1 (δ stands for double):

$$\delta(\varphi_1, \varphi_2) \;:=\; \epsilon(\varphi_1, k_1) \cup \epsilon(\varphi_1, k_2) \cup \{\mathit{freq}\,(k_1 \wedge k_2) = 0, \epsilon(\varphi_2, k_1 \vee k_2)\} \;.$$

This δ-expression creates two disjoint items k_1 and k_2 that have the same frequency as φ_1, and sets the frequency of φ_2 equal to the frequency of $k_1 \vee k_2$. As k_1 and k_2 never occur in the same transaction, the frequency of φ_1 must hence be twice the frequency of φ_1.

Again, the same remark as with ϵ applies: if we use multiple δ-expressions simultaneously, we implicitly assume that the items k_1, k_2 are replaced by unique, new items.

Obviously, we can also multiply by $3, 4, \ldots$, by making enough disjoint copies of φ_1 with ϵ, and setting φ_2 equal to $k_1 \vee k_2 \vee \ldots$ This method, however, has one big disadvantage: the formulas to multiply with n would be exponentially large in the size of the *representation* of n. This can easily be solved though, by iterative doubling and adding. Let n be a positive integer with binary representation $b_\ell \ldots b_0$. That is, $n = \sum_{j=0}^{\ell} b_j 2^j$. We introduce the expression

$\mathcal{MULT}_n(\varphi_1, \varphi_2)$ as follows:

$$\begin{aligned}\mathcal{MULT}_n(\varphi_1, \varphi_2) \;=\; & \epsilon(\varphi_1, b_0) \cup \delta(b_1, b_0) \cup \ldots \cup \delta(b_\ell, b_{\ell-1}) \\ & \cup \{\mathit{freq}\,(b_i \wedge b_j) = 0 \mid 0 \leq i < j \leq \ell \;\wedge\; b_i = b_j = 1\} \\ & \cup \epsilon(\varphi_2, \bigvee_{\substack{b_j=1 \\ 0\leq j\leq \ell}} b_j)\end{aligned}$$

This construction of ϵ is illustrated in Figure 3. The first line of this expression makes sure that for all $i = 1 \ldots \ell$, b_i has frequency $2^i \cdot \mathit{freq}\,(\varphi_1)$. b_i corresponds hence to the ith bit in the binary representation of n. The second line ensures that no two items, that represent 1-bits, occur in the same transaction. Finally, the last line sets φ_2 equal to the disjunction of the 1-bits b_j. Because the frequency of b_j is 2^j times the frequency of φ_1, and all items representing 1-bits are in separate transactions, the frequency of this disjunction is exactly n times the frequency of φ_1.

The following *Multiplication Lemma* now states the correctness of the above constructions:

Lemma 2 (Multiplication Lemma) *If $\mathcal{D}$ satisfies $\mathcal{MULT}_{n_1}(\varphi_1^1, \varphi_2^1) \cup \ldots \cup \mathcal{MULT}_{n_\ell}(\varphi_1^\ell, \varphi_2^\ell)$, then for all $j = 1 \ldots \ell$, $n_j \cdot \mathit{freq}\left(\varphi_1^j, \mathcal{D}\right) = \mathit{freq}\left(\varphi_2^j, \mathcal{D}\right)$.*

There exists a database $\mathcal{D}$ that satisfies $\mathcal{C}$ and with for all $j = 1 \ldots \ell$, $n_j \cdot \mathit{freq}\left(\varphi_1^j, \mathcal{D}\right) = \mathit{freq}\left(\varphi_2^j, \mathcal{D}\right)$ if and only if there exists a database $\mathcal{D}$ that satisfies $\mathcal{C} \cup \mathcal{MULT}_{n_1}(\varphi_1^1, \varphi_2^1) \cup \ldots \cup \mathcal{MULT}_{n_\ell}(\varphi_1^\ell, \varphi_2^\ell)$.

Proof The proof of this lemma can be found in Appendix D. □

Example 5 *In Figure 3, the construction of $\mathcal{MULT}_{11}(\varphi_1, \varphi_2)$ is illustrated. The binary representation of* 11 *is* 1011*. The expression for $\mathcal{MULT}_{11}(\varphi_1, \varphi_2)$ is thus:*

$$\begin{aligned}& \epsilon(\varphi_1, b_0) \cup \delta(b_1, b_0) \cup \delta(b_2, b_1) \cup \delta(b_3, b_2) \\ & \cup \{\mathit{freq}\,(b_0 \wedge b_1) = 0, \mathit{freq}\,(b_0 \wedge b_3) = 0, \mathit{freq}\,(b_1 \wedge b_3) = 0\} \\ & \cup \{\epsilon(\varphi_2, b_0 \vee b_1 \vee b_3)\}\end{aligned}$$

The first line ensures that for $i = 0 \ldots 3$, $\mathit{freq}\,(b_i) = 2^i \cdot \mathit{freq}\,(\varphi_1)$. The second line expresses that no transaction contains more than one of b_0, b_1, and b_3. Therefore,

$$\mathit{freq}\,(b_0 \vee b_1 \vee b_3) \;=\; \mathit{freq}\,(b_0) + \mathit{freq}\,(b_1) + \mathit{freq}\,(b_3) \;=\; 11 \cdot \mathit{freq}\,(\varphi_1) \quad .$$

Finally, the last line states that φ_2 is in exactly those transactions that satisfy $\mathit{freq}\,(b_0 \vee b_1 \vee b_3)$. Therefore, $\mathit{freq}\,(\varphi_2) = 11 \cdot \mathit{freq}\,(\varphi_1)$, as well.

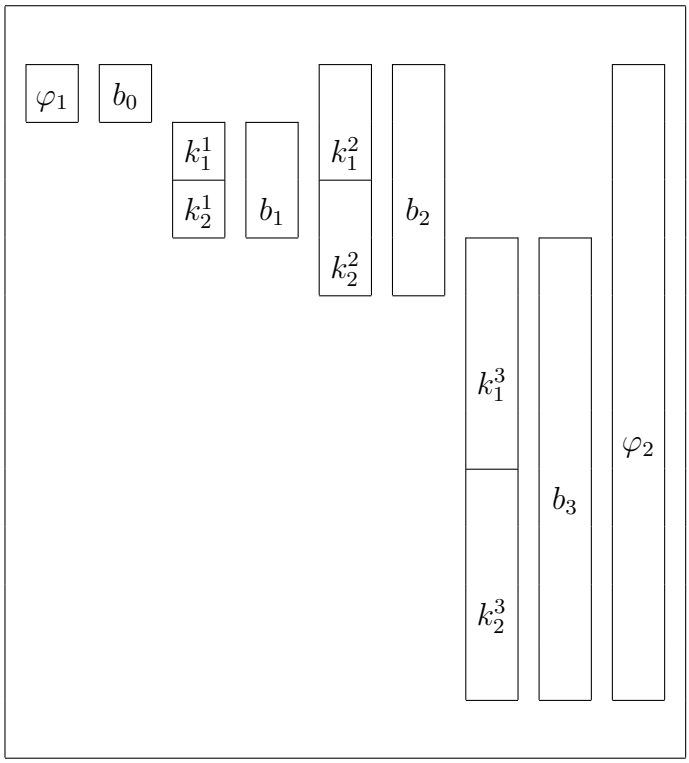


Fig. 3. Construction of $\mathcal{MULT}_{11}(\varphi_1, \varphi_2)$

4.2 Expressing Association Rules in `FREQSAT`

Assume that besides frequency constraints $\mathcal{C}$, also a set of association constraints $\mathcal{A}$ has been given. We show that there exists an extended `FREQSAT`-instance $\mathcal{EFSAT}(\mathcal{C} \cup \mathcal{A})$ that is equivalent to $\mathcal{C} \cup \mathcal{A}$. Thus, because of the equivalence between `FREQSAT` and extended `FREQSAT` proven in Section 3, this implies that $\mathcal{C} \cup \mathcal{A}$ is equivalent to $\mathcal{FSAT}(\mathcal{EFSAT}(\mathcal{C} \cup \mathcal{A}))$.

Consider the association constraint $\mathit{conf}(\varphi \rightarrow \psi) \in [l, u]$. This constraint holds in a database $\mathcal{D}$ if and only if

$$l \cdot \mathit{freq}(\varphi, \mathcal{D}) \leq \mathit{freq}(\varphi \wedge \psi, \mathcal{D}) \leq u \cdot \mathit{freq}(\varphi, \mathcal{D}) \quad .$$

Let $L = N \cdot l$, and $U = N \cdot u$. Then, The association constraint holds if:

$$L \cdot \mathit{freq}(\varphi, \mathcal{D}) \leq N \cdot \mathit{freq}(\varphi \wedge \psi, \mathcal{D}) \leq U \cdot \mathit{freq}(\varphi, \mathcal{D}) \quad .$$

The translation now seems easy: we introduce new items α, β, and γ, and

using the multiplication lemma, we enforce that:

$$\begin{aligned} \mathit{freq}(\alpha) &= L \cdot \mathit{freq}(\varphi) & \mathit{freq}(\alpha \wedge \neg\gamma) &= 0 \\ \mathit{freq}(\beta) &= U \cdot \mathit{freq}(\varphi) & \mathit{freq}(\neg\beta \wedge \gamma) &= 0 \\ \mathit{freq}(\gamma) &= N \cdot \mathit{freq}(\varphi \wedge \psi) \end{aligned}$$

That is, we create three new items α, β, γ with frequencies such that the association constraint is satisfied if $\mathit{freq}(\alpha) \leq \mathit{freq}(\gamma) \leq \mathit{freq}(\beta)$. This relation between the frequencies of the three items is enforced by the two constraints of the right that require that every α occurs together with γ, and every γ occurs together with a β.

However, it is very well possible that for example $U \cdot \mathit{freq}(\varphi)$ is larger than 1, and hence the multiplications cannot be carried out. To resolve this problem, we will embed the database that satisfies $\mathcal{C} \cup \mathcal{A}$ into a larger database. Let $N_{\mathcal{A}}$ be the smallest integer such that for all association constraints $\mathit{conf}(\varphi \rightarrow \psi) \in [l, u]$, $N_{\mathcal{A}} \cdot l$ and $N_{\mathcal{A}} \cdot u$ are integers. That is, $N_{\mathcal{A}}$ is the smallest common multiple of the denominators of the bounds on the association constraints. The larger database, in which the satisfying database $\mathcal{D}$ of $\mathcal{C} \cup \mathcal{A}$ will be embedded will have $N_{\mathcal{A}} \cdot |\mathcal{D}|$ transactions. To indicate which transactions belong to the database $\mathcal{D}$, a new item d is introduced. The transactions containing d will form a database satisfying $\mathcal{C} \cup \mathcal{A}$. The other transactions are there to create the space to do the multiplications. Since we maximally multiply with $N_{\mathcal{A}}$, there will always be enough space to be able to apply the multiplication lemma.

Hence, we get the following constraints. Let

$$\begin{aligned} \mathcal{C} &= \{\mathit{freq}(\varphi_j) \in [l_j, u_j], j = 1 \ldots m\} \\ \mathcal{A} &= \{\mathit{conf}(\varphi_k \rightarrow \psi_k) \in [L_k/N_{\mathcal{A}}, U_k/N_{\mathcal{A}}], k = 1 \ldots \ell\} \end{aligned}$$

Then, we define the extended FREQSAT-instance $\mathcal{E}FSAT(\mathcal{C} \cup \mathcal{A})$ as

$$\begin{aligned} &\{\mathit{freq}(\varphi_j \wedge \{d\}) \in [l_j/N_{\mathcal{A}}, u_j/N_{\mathcal{A}}], j = 1 \ldots m\} \cup \{\mathit{freq}(d) = 1/N_{\mathcal{A}}\} \\ &\textstyle\bigcup_{k=1}^{\ell} \ (\mathcal{M}ULT_{L_k}(\varphi_k, \alpha_k) \ \cup \ \mathcal{M}ULT_{U_k}(\varphi_k, \beta_k) \ \cup \ \mathcal{M}ULT_{N_{\mathcal{A}}}(\varphi_k \wedge \psi_k, \gamma_k)) \\ &\cup \{\mathit{freq}(\alpha_k \wedge \neg\gamma_k) = 0, k = 1 \ldots \ell\} \ \cup \ \{\mathit{freq}(\neg\beta_k \wedge \gamma_k) = 0, k = 1 \ldots \ell\} \end{aligned}$$

Theorem 4 *$\mathcal{C} \cup \mathcal{A}$ is satisfiable if and only if the extended* FREQSAT*-instance $\mathcal{E}FSAT(\mathcal{C} \cup \mathcal{A})$ is satisfiable. Furthermore, $ENT_I(\mathcal{C} \cup \mathcal{A}) = [l, u]$ if and only if $ENT_I(\mathcal{E}FSAT(\mathcal{C} \cup \mathcal{A})) = [l/N_{\mathcal{A}}, u/N_{\mathcal{A}}]$.*

Proof This is a direct consequence of the multiplication lemma. □

Example 6 *Let $\mathcal{C} \cup \mathcal{A}$ be the following set:*

$$\{freq\,(a) = 1/2, freq\,(a \vee b) = 3/4\} \cup \{conf(a \rightarrow b) \in [1/2, 1]\} \ .$$

N here equals 2. The association constraint $conf(a \rightarrow b) \in [1/2, 1]$ holds if and only if

$$freq\,(a) \leq 2 \cdot freq\,(a \wedge b) \leq 2 \cdot freq\,(a) \ .$$

Hence, the items α and β will have frequency equal to respectively $freq\,(a)$ and $2 \cdot freq\,(a)$. γ will have frequency $2 \cdot freq\,(a \wedge b)$. The association constraint is enforced by requiring that every transaction with α also contains γ, and every one with γ also has β.

The following databases satisfy respectively $\mathcal{C} \cup \mathcal{A}$ and $\mathcal{E}FSAT(\mathcal{C} \cup \mathcal{A})$:

TID	Items
1	a
2	a, b
3	b
4	

$\longrightarrow$

TID	Items
1	d, a
2	d, a, b
3	d, b
4	d

TID	Items
5	α, γ, β
6	α, γ, β
7	β
8	β

4.3 Implications for Approximation Results

In this subsection we discuss the approximation of the entailment version of `FREQSAT`. Based on the ability to express association rules, it is not too hard to prove that `FREQSAT` cannot be approximated. More concretely, we show that the **NP**-complete satisfiability problem $\mathcal{C}$ can be reduced to the function problem $Func(\mathcal{C})$ in such a way that if $\mathcal{C}$ is satisfiable, then $ENT_{\{d,t_i\}}(Func(\mathcal{C})) = [0, 0.5]$, otherwise $ENT_{\{d,t_i\}}(Func(\mathcal{C})) = [0, 0]$. Therefore, unless **P** equals **NP**, there cannot exist an approximation algorithm that approximates the upper bound of the interval with an absolute error less than 0.25, because otherwise we would have a deterministic polynomial procedure to decide `FREQSAT`. Thus, for any polynomial time approximation, the relative error on the upper bound is unbounded.

Let $\mathcal{C}$ be $\{freq\,(I_1) \in [l_1, u_1], \ldots, freq\,(I_m) \in [l_m, u_m]\}$, and let i and a be items in none of the I_j's. $Func(\mathcal{C})$ denotes the following `FREQSAT`-problem:

$$\mathcal{F}SAT\left(\mathcal{E}FSAT\left(\begin{array}{l}\{conf(\{a\} \rightarrow I_1) \in [l_1, u_1], \ldots, \\ conf(\{a\} \rightarrow I_m) \in [l_m, u_m]\}\end{array}\right) \ \cup \ \mathcal{M}ULT_N(\{a\}, \{i\})\right)$$

with N being the least common multiplier of the denominators of l_1, u_1, ..., l_m, u_m.

Theorem 5 *Let $\mathcal{C}$ be a set of frequency constraints. If $\mathcal{C}$ is satisfiable then $ENT_{\{d,t_i\}}(Func(\mathcal{C})) = [0, 0.5]$, otherwise $ENT_{\{d,t_i\}}(Func(\mathcal{C})) = [0, 0]$.*

Proof If $\mathcal{C}$ is not satisfiable, then the only way to satisfy the following set of association constraints is by a transaction database with $freq\,(\{a\}) = 0$:

$$\{conf(\{a\} \rightarrow I_1) \in [l_1, u_1], \ldots, conf(\{a\} \rightarrow I_1) \in [l_k, u_k]\}$$

On the other hand, if $\mathcal{C}$ is satisfiable, then we can satisfy these association constraints by adding the item a to every transaction of a database satisfying $\mathcal{C}$. Therefore, if $\mathcal{C}$ is not satisfiable, $freq\,(\{a\})$ is 0, otherwise $freq\,(\{a\})$ can be any number in the interval $[0, 1]$. Because of Theorem 4, it follows that the entailed frequency interval for $freq\,(\{a\})$ in the *extended* `FREQSAT`-instance

$$\mathcal{E} := \mathcal{E}FSAT(\{conf(\{a\} \rightarrow I_1) \in [l_1, u_1], \ldots, conf(\{a\} \rightarrow I_m) \in [l_m, u_m]\})$$

is $[0, 0]$ if $\mathcal{C}$ is not satisfiable, and $[0, 1/N]$ otherwise. Because of the constraint $\mathcal{M}ULT_N(\{a\}, \{i\})$, and the Multiplication Lemma 2, the entailed interval for $\{i\}$ given the *extended* `FREQSAT`-instance

$$\mathcal{E} \cup \{\mathcal{M}ULT_N(\{a\}, \{i\})\}$$

is $[0, 0]$ and $[0, 1]$ in these respective cases. Finally, because of Theorem 3, we get that for the `FREQSAT`-instance

$$\mathcal{F}SAT(\mathcal{E} \cup \{\mathcal{M}ULT_N(\{a\}, \{i\})\}) \ ,$$

the entailed interval for $\{d, t_i\}$ is $[0, 0]$ if $\mathcal{C}$ is not satisfiable, and is $[0, 0.5]$ otherwise, as, going from *extended* `FREQSAT` to `FREQSAT`, we have to take into account a factor of $1/2$ on the bounds. □

5 Axiomatization of `FREQSAT`

In this section we show that `FREQSAT` does not have an axiomatization of finite arity. We use the same notations and approach to axiomatizations as in [1]. We consider a countable infinite set of items $\mathcal{I}$. The set of all frequency constraints $freq\,(I) \in [l, u]$, with I a finite subset of $\mathcal{I}$ is denoted $\mathcal{S}$. It is clear that any instance of the `FREQSAT`-problem can be mapped to the satisfiability of a finite subset of $\mathcal{S}$.

A *ground inference rule* is an expression of the form (**if** S **then** s), where $S \subseteq \mathcal{S}$, and $s \in \mathcal{S}$. This rule is said to be *sound*, if $S \models s$. A set of ground

inference rules $\mathcal{R}$ is sound is each rule in $\mathcal{R}$ is sound.

Let $\Sigma \cup \{\sigma\} \subseteq \mathcal{S}$ be a set of frequency constraints. A *proof* of σ from Σ using $\mathcal{R}$ is a sequence $\sigma_1 \dots \sigma_n = \sigma$ such that for every $i = 1 \dots n$, either σ_i is in Σ, or there is a rule (**if** S **then** s) in $\mathcal{R}$, such that $S \subseteq \{\sigma_1, \dots, \sigma_{i-1}\}$, and $\sigma_i = s$. We write $\Sigma \vdash_{\mathcal{R}} \sigma$ if there is a proof of σ from Σ using $\mathcal{R}$.

$\mathcal{R}$ is called *complete* if for each pair (Σ, σ), $\Sigma \models \sigma$ implies $\Sigma \vdash_{\mathcal{R}} \sigma$. A set of rules $\mathcal{R}$ is an *axiomatization* if it is sound and complete.

Intuitively, an axiomatization is called *finite* if there is an axiomatization $\mathcal{R}$, such that there exists a finite set of *inference rule schemas* that can be instantiated to form $\mathcal{R}$. Instead of formalizing the somewhat fuzzy notion of inference rule schemas, however, we concentrate on axiomatizations of ground inference rules, and prove properties of them.

An implication rule (**if** S **then** s) is said to be k-*ary* for some $k \geq 0$, if $|S| = k$. An axiomatization $\mathcal{R}$ is k-ary if each rule in it is l-ary with $l \leq k$. We show next that `FREQSAT` is not finitely axiomatizable by an axiomatization of finite arity. Thus, this implies that every axiomatization for `FREQSAT`, must include inference rules of arbitrary large arity.

We furthermore show in this section that, nevertheless, there exists a recursive axiomatization of `FREQSAT`, and when we fix the itemsets that can occur in frequency constraints, we can always obtain a finite, and locally complete axiomatization.

The most important property of an axiomatization system is that it provides an effective procedure to reason about frequency constraints. Also, an axiomatization systems reveals the actual structure of the problem by giving all tools to solve the problem. Therefore, if possible, having a complete axiomatization system is very desirable. In this context, the axiomatization results in this section address these issues. The importance of the negative results concerning the axiomatization is that it shows that any local deduction procedure is necessarily incomplete. Any complete method must, at one point, take into account all given frequency constraints at the same time. Notice that this is different than, e.g., the Armstrong axioms for functional dependencies. As such, divide and conquer techniques are, inevitably, incomplete.

5.1 Any Axiomatization of `FREQSAT` *Has Infinite Arity*

We first give a theorem that provides a set of axioms that are sound and complete in the special case that for every subset I of a finite set $\mathcal{I}$, a frequency constraints $freq\,(I) = f_I$ is given. The number of axioms depends on the set $\mathcal{I}$,

and the axioms are only complete in this very special case. In Theorem 7, we then show that in general, no axiomatization of finite arity exists for FREQSAT.

Theorem 6 ([11]) *Let for all $I \subseteq \mathcal{I}$, f_I be a rational number. There exists a transaction database $\mathcal{D}$ such that for all $I \subseteq \mathcal{I}$, $freq(I, \mathcal{D}) = f_I$ if and only if, for all $I \subseteq \mathcal{I}$, the following rule holds:*

$$\mathcal{R}_{\mathcal{I}}(I) \qquad \sigma_{\mathcal{I}}(I) \;=\; \sum_{I \subseteq K \subseteq \mathcal{I}} (-1)^{|K-I|} f_K \geq 0$$

Theorem 7 *Every axiomatization for* FREQSAT *that does not include an axiom that involves the frequency of all nonempty itemsets is incomplete. Therefore, for no k does there exist a k-ary sound and complete axiomatization for* FREQSAT.

Proof Let n be an arbitrary number. We construct a FREQSAT problem $\mathcal{C}$ over the set $\mathcal{I} = \{i_1, \ldots, i_n\}$, such that (a) $\mathcal{C}$ is not satisfiable, but, (b) every strict subset of $\mathcal{C}$ is satisfiable. Furthermore, $\mathcal{C}$ contains one expression $freq(I) = f_I$ for every $I \subseteq \mathcal{I}$.

From (a) and (b) it follows then that an axiomatization for FREQSAT must contain at least one axiom that involves every frequency constraint in the input. Indeed; suppose that the axioms $A_1, A_2, \ldots, A_m$ are sound and complete for FREQSAT, but none of the axioms A_i involves all frequencies. Because $\mathcal{C}$ is not satisfiable, there must be at least one axiom A that is not satisfied by $\mathcal{C}$. This is so because $\mathcal{C}$ contains an expression $freq(I) = f_I$, for every subset I of $\mathcal{I}$. Hence, every expression $freq(I) \in [l, u]$ entailed by $\mathcal{C}$ is either in contradiction with $freq(I) = f_i$, or is less expressive. Therefore, if it can be derived by the axioms that $\mathcal{C}$ is not satisfiable, then this can be derived in one step. Suppose that this unsatisfied axiom A does not involve itemset I, and c is the constraint in $\mathcal{C}$ involving I. Then we have the following contradiction: $\mathcal{C} \setminus \{c\}$ is satisfiable, but violates A.

The full proof can be found in Appendix E. □

5.2 Recursive Axiomatization of FREQSAT

Theorem 8 FREQSAT *is recursively axiomatizable. That is, it is decidable if a given rule* (**if** S **then** s) *is sound.*

Proof From Theorem 6, it follows that the set of frequency constraints

$$\mathcal{C} \;=\; \{freq(I_1) \in [l_1, u_1], \ldots, freq(I_m) \in [l_m, u_m]\}$$

is satisfiable if and only if the following system of linear inequalities $Prog(\mathcal{C})$ has a solution: Let, for every subset I of $\mathcal{I} = \bigcup_{i=1}^{m} I_m$, x_I be a variable.

$$\begin{cases} \sum_{I \subseteq K \subseteq \mathcal{I}} (-1)^{|K-I|} x_K \geq 0 & \forall I \subseteq \mathcal{I} \\ x_{I_j} \geq l_j & \forall j = 1 \ldots m \\ x_{I_j} \leq u_j & \forall j = 1 \ldots m \end{cases}$$

Via linear programming we can now compute the minimal and maximal frequency of any given set I, by minimizing/maximizing the variable x_I given the linear program $Prog(\mathcal{C})$. Hence, given $\mathcal{C}$, for any set I we can compute the tightly entailed interval $[l, u]$, and therefore also check the soundness of a given rule (**if** S **then** s). □

5.3 Locally Complete Axioms

We show how we can construct an axiomatization for the case where the input sets are fixed. This generic construction is based on the Fourier-Motzkin elimination method [19, p. 84] for linear systems of inequalities.

Construction of Axioms. Suppose that we want to make axioms for the specific case that we know bounds on the sets $\{a\}$, $\{b\}$, and $\{a, b, c\}$. We denote the hypothetical bounds on a set I by $[l_I, u_I]$. We can state the existence of a satisfying database with a linear program, in the same way as in the proof of Theorem 8:

$$\begin{cases} x_{abc} \geq 0 & x_{abc} \geq x_{ab} + x_{ac} - x_a \\ x_{ab} \geq x_{abc} & x_{abc} \geq x_{ab} + x_{bc} - x_b \\ x_{ac} \geq x_{abc} & x_{abc} \geq x_{ac} + x_{bc} - x_c \\ x_{bc} \geq x_{abc} & x_{ab} + x_{ac} + x_{bc} - x_a - x_b - x_c + 1 \geq x_{abc} \\ x_a \geq l_a & u_a \geq x_a \\ x_b \geq l_b & u_b \geq x_a \\ x_{abc} \geq l_{abc} & u_{abc} \geq x_{abc} \end{cases}$$

Thus, given bounds on the frequency of $\{a\}$, $\{b\}$, and $\{a, b, c\}$, there exists a database that satisfies them if and only if the above system has a solution. It would, however, be nicer if we had existence conditions that did not involve the variables x_I. For this, we can use the Fourier-Motzkin elimination method. This elimination method allows to remove variables from linear systems of inequalities, without affecting the satisfiability of the system.

First we eliminate x_a. This is done as follows. In all the inequalities that involve x_a, we isolate x_a:

$$\begin{array}{rl} l_a \leq x_a & x_a \leq u_a \\ x_{ab} + x_{ac} - x_{abc} \leq x_a & x_a \leq x_{ab} + x_{ac} + x_{bc} - x_b - x_c + 1 - x_{abc} \end{array}$$

We can eliminate x_a by replacing these inequalities with all inequalities $lin_1 \leq lin_2$ such that $lin_1 \leq x_a$ and $x_a \leq lin_2$ was in the original system. On the one hand, it is easy to see that in every solution to the original system these new inequalities are fulfilled. Hence, if the original system had a solution, then the new system has a solution as well. On the other hand, if we have a solution for the new system, the new inequalities ensure that there does exist an x_a. Indeed; suppose for the sake of contradiction that no such x_a exists. Then, there must exist inequalities $lin_1 \leq x_a$ and $x_a \leq lin_2$ such that the value of lin_1 is larger than the value of lin_2 in the solution. This is however in contradiction with the fact that in the new system, the inequality $lin_1 \leq lin_2$ is satisfied.

In our example, eliminating x_a results in replacing the inequalities in (5.3) with the following equivalent inequalities that no longer involve variable x_a.

$$\begin{cases} u_a \geq l_a & x_{ab} + x_{ac} + x_{bc} - x_b - x_c + 1 - x_{abc} \geq l_a \\ u_a \geq x_{ab} + x_{ac} - x_{abc} & x_{bc} - x_b - x_c + 1 \geq 0 \end{cases}$$

We then continue eliminating all other variables x_I one by one. The final result of all eliminations is:

$$\begin{cases} 0 \leq u_a & l_a \leq 1 & l_a \leq u_a & l_{abc} \leq u_a \\ 0 \leq u_b & l_b \leq 1 & l_b \leq u_b & l_{abc} \leq u_b \\ 0 \leq u_{abc} & l_{abc} \leq 1 & l_{abc} \leq u_{abc} & \end{cases}$$

The leftmost 3 columns just state that the intervals $[l, u]$ must contain at least one possible frequency; i.e., $[l, u] \cap [0, 1] \neq \{\}$. This translates to the conditions $l \leq u$, $l \leq 1$, $u \geq 0$. The rightmost two conditions state the monotonicity rules; the lower bound on $\{a, b, c\}$ must always be smaller than the upper bounds of $\{a\}$ and $\{b\}$. Thus, these conditions together with the implicit assumption that $[l, u]$ is a non-empty subinterval of $[0, 1]$ for all bounds, gives the following 5 axioms for the special case in which bounds on $\{a\}$, $\{b\}$, and $\{a, b, c\}$ have been given:

$$\{l_a \leq u_a \qquad l_b \leq u_b \qquad l_{abc} \leq u_{abc} \qquad l_{abc} \leq u_a \qquad l_{abc} \leq u_b$$

Entailment. With a slight variation on the elimination method, we can find a complete set of deduction rules for the entailment problem as well.

Suppose that we want to entail formulas that give tight bounds on the frequency of $\{a, b, c\}$ in the case that $\{freq(\{a\}) \in [l_a, u_a], freq(\{b\}) \in [l_b, u_b]\}$ has been given. We construct a similar system as in last section:

$$\begin{cases} x_{abc} \geq 0 & x_{ab} + x_{ac} + x_{bc} - x_a - x_b - x_c + 1 \geq x_{abc} \\ x_{ab} \geq x_{abc} & x_{abc} \geq x_{ab} + x_{ac} - x_a \\ x_{ac} \geq x_{abc} & x_{abc} \geq x_{ab} + x_{bc} - x_b \\ x_{bc} \geq x_{abc} & x_{abc} \geq x_{ac} + x_{bc} - x_c \\ x_a \geq l_a & u_b \geq x_a \\ u_a \geq x_a & x_b \geq l_b \end{cases}$$

In this system we eliminate all x_I's except for x_{abc}. This gives the following, equivalent system:

$$\begin{cases} l_a \leq 1 & l_a \leq u_a & x_{abc} \leq 1 & x_{abc} \leq u_b \\ l_b \leq 1 & l_b \leq u_b & x_{abc} \leq u_a & 0 \leq x_{abc} \end{cases}$$

The two leftmost columns of conditions are again existence conditions. The rightmost 4 conditions show that

$$\{freq(\{a\}) \in [l_a, u_a], freq(\{b\}) \in [l_b, u_b]\} \models_{tight} freq(\{a, b, c\}) \in [0, \min\{1, u_a, u_b\}] \ .$$

6 Related Work and Applications

6.1 Related Work

Deduction of frequencies. Before the systematic study of the FREQSAT-problem, several special instances have been studied. In [16], a complete axiomatization is given and the complexity is studied of the case where only lower bounds on the frequencies are known. An example of a deduction that can be made with the axioms in [16], is that from $freq(I) \geq 60\%$ and $freq(J) \geq 60\%$, it follows that $freq(I \cup J) \geq 20\%$. FREQSAT, however, is much more difficult and complex. An example of a deduction that cannot be made using the axioms of [16] is the following: from $freq(I) \geq 60\%$, $freq(J) \geq 60\%$, and $freq(I \cap J) = 80\%$ it follows that $freq(I \cup J) \geq 40\%$. The relative simplicity

of only considering lower bounds is also clear from the fact that the complete deduction for the case studied in [16] can be performed in polynomial time, while FREQSAT is **NP**-complete.

In [13, 14, 15], another special case is studied. In this case, for a given itemset I, the frequency of all its strict subsets are known exactly. For this case, deduction rules are given to derive tight bounds on the frequency of I. This deduction can be done in polynomial time; on the one hand, the number of deduction rules is exponential in the size of I, but, on the other hand, also the size of the input, i.e., the frequency for every strict subset of I, is exponential in the size of I. Based on these deduction rules, the non-derivable itemsets are introduced as a condensed representation for itemsets. Again the FREQSAT-problem is much more general, as it allows any collection of itemsets as input, and it allows for intervals instead of exact frequencies.

In [44], *Tatti* studies the complexity of a set of boolean query problems, most of which are in fact a special case of the FREQSAT-problem, in the sense that not every collection of frequency constraints are allowed, but only anti-monotonic collections, and exact frequencies. In this context, Tatti showed that deciding consistency is still **NP**-complete, and deciding if it is possible that a certain target itemset B has a frequency of at least b given an anti-monotonic, exact set of frequency constraints remains **NP**-complete, even when the given set of constraints is consistent. Tatti also studies a variant based on maximal entropy that is provably more complex (given **NP** does not equal **PP**.)

In [45], *Tatti* studies the same entailment problem as studied in this paper and [12] for frequencies of itemsets. Conditions are given for which the linear programming problem that needs to be solved in order to determine tight bounds can be simplified. In this context, the notion of a safe set is proposed; a safe set is one on which the linear program can be "projected" without changing the solutions of the program. In this way, safe sets can provide a more time-efficient solution for the entailment problem in specific cases.

Probabilistic logics. In artificial intelligence literature, probabilistic logic [26] and reasoning about uncertainty and belief [38] is studied intensively. The link with this paper is that the frequency of an itemset I can be seen as the probability that a randomly chosen transaction from the transaction database satisfies I; i.e., we can consider the transaction database as an underlying probability structure. Some examples of probabilistic logics include the pSAT-problem introduced by *Nilsson* [37], and extensions to intervals, conditional constraints, etc. [22, 27, 28, 33, 34], Another interesting probabilistic language is formed by the weight formulas of *Fagin et al.* [21]. A *basic weight formula* is an expression $a_1 w(\phi_1) + \ldots + a_k w(\phi_k) \geq c$, where $a_1, \ldots, a_k$ and c are integers and $\phi_1, \ldots, \phi_k$ are propositional formulas, meaning that the sum of all a_i times the *weight* of ϕ_i is greater than or equal to c. A *weight formula* is a

boolean combination of basic weight formulas. The main contribution of [21] is the description of a sound and complete axiomatization for this probabilistic logic. All types of frequency constraints studied in this paper can be expressed in this probabilistic logic.

Closely related to our work on axiomatizing FREQSAT are [30, 33]. In [33], *Lukasiewicz* gives a locally complete rule for the inference of the conditional probability of $P(A|C)$, given intervals on the probabilities $P(A|B)$, $P(B|A)$, $P(C|B)$, and $P(B|C)$, and a taxonomy on the premises. *Jaeger* [30] develops a method for automatic derivation of probabilistic inference rules for conditional probabilities comparable to the method we propose to get a locally complete axiomatization. Given parameterized bounds on some input conditional probabilities, a parameterized optimal bound for a target output conditional probability is calculated. This parameterized solution is then the rule. Notice that the problem studied by Jaeger is strictly more general than the problem studied here. As such, the methods applied by Jaeger also apply to our problem. More specifically, Jaeger does not use the relatively simple Fourier-Motzkin elimination methods as we do, but instead analyzes the list of the parameterized vertices of the polytope $V(\mathcal{C})$ consisting of the instantiations that satisfy the input constraints. As such, Jaeger's method might result in less redundant rules, although essentially, both methods must result in the same, or at least equivalent, results, although for large numbers of variables the method proposed by Jaeger [30] will outperform our method.

6.2 Applications

Privacy preserving data mining. Data Mining can be a serious threat to the privacy[4, 32]. In this context, methods have been developed that aim at changing databases in such a way that still meaningful data mining results can be produced from it, but the individual data is randomized [4]. Notice, however, that in the approach given in [4], privacy might still be compromised. A popular way to quantify anonymity of a released dataset is the notion of k-anonymity [41]. A dataset is k-anonymous if every tuple in the published data corresponds to at least k individuals. Another setting is that multiple parties do not want to share their data, but nevertheless want to build data mining models over the union of their databases. In this setting, cryptographic techniques can be used to guarantee privacy [32].

It is, however, conceivable that the mining is done by a trusted party. In that case, there is no risk of disclosure based on the original data. Even though, the results of the mining themselves can disclose more of the original data than is desirable [5]. Closely related to this concern is the research in *statistical disclosure control* [20]. There it is studied how to prevent that users can retrieve

information of individuals from a statistical database by subsequently asking queries. A popular technique here is to perturbate the output data. We can make the connection between FREQSAT and statistical disclosure control, by considering the class of queries asking for the frequency of itemsets. The disclosure question then becomes: "How much can be inferred from the original database by knowing the frequencies of a collection of published itemsets?"

The process of trying to reconstruct parts of the original database from data mining results is called *inverse data mining* [36]. The FREQSAT-problem, its various variants and the entailment problems can be situated in this context. The results of a frequent set mining operation can be represented as an instance of FREQSAT. Inverse data mining would then amount to deriving the frequencies of other itemsets, not in the result set. In this context, the high complexities of the problems studied in this paper are bad news: suppose that we want to publish some itemsets with their frequencies, but first we want to assess how much these frequencies disclose of the original dataset. This problem can be stated as one of the variants of FREQSAT. The high complexity of the FREQSAT-problems in this paper, however, shows that there is little hope that it is effectively possible to assess the degree of disclosure. On the bright side, the high complexity means also that it is potentially very hard to break the privacy. However, the situation is different from that of, for example, public key encryption. In inverse mining, partial information can be derived with incomplete methods, whereas, in general, in public key encryption, the code cannot be *partially* broken. Hence, in inverse mining, the more computing power one has, the more one can derived. Therefore, unless one has superior computing power over potentially malicious parties, the results of mining cannot be guaranteed to be safe.

In [46], the following problem of approximate inverse frequent itemset mining is studied. Given some itemsets with their absolute support, does there exist a database such that these support constraints are *approximately* satisfied, in the sense that a difference proportional to the number of constraints given is allowed. This problem is shown to be **NP**-complete. Also an approximate algorithm to determine information leakage is given. In [18, 47], heuristic methods for generating a database (approximately) satisfying given frequency constraints are given. The idea behind this database generation is to, instead of publishing a confidential database, generate a new database with the same frequency information that can be published for analysis purposes. The feasibility of these approaches depends highly on the assumption that many of the items are (conditionally) independent.

Condensed Representations. Another application is the construction of condensed representations [35]. A condensed representation of a dataset is a summary of the dataset that allows to answer a certain target class of queries more efficiently than based on the complete dataset. In this context, in [35],

the example is given of the collection of frequent itemsets that can serve as a condensed representation to answer frequency queries for arbitrary Boolean expressions over the attributes in a binary-valued dataset.

In many applications, however, even the collection of frequent itemsets is too large to enumerate. In this context, condensed representations for the collection of frequent itemsets have been studied. A condensed representation of the frequent sets is a summary of the data that allows to derive, for every itemset, whether or not it is frequent in the database, and if it is frequent, its actual frequency. As such, in the research on condensed representations of the frequent itemsets, the query class is fixed to frequency queries for conjunctions only, i.e., for itemsets. Often, condensed representations for itemsets are a subset of the complete collection of itemsets that allow to derive or approximate the frequency information of the other frequent sets. Some examples of exact representations are the closed sets [39], the free sets [9, 10], and the non-derivable itemsets [14, 15]. Examples of approximate representations include the δ-free sets [9, 10]. Pavlov et al. [40] study how probabilistic models such as the maximum entropy model can be used to approximate answers to queries posed to large sparse binary data sets. Sometimes, one is not interested in the exact frequencies, but only in the frequent sets themselves. Afrati et at. [2] show an approximate solution to how k sets can be selected that cover the complete collection of frequent sets as good as possible. This work is then further extended by Yan et al. [48] to a profile-based approach that is not only good at summarizing the patterns themselves, but also at integrating their supports.

For an overview of exact condensed representations for the itemset domain, see [17]. In such condensed representations typically only non-redundant information is stored. Entailment of frequencies as in the `FREQSAT`-problem allows for derivation of frequencies. The stronger the deduction mechanism, the more redundancy in the set of frequencies can be found. The complexity results in this paper indicate that complete deduction in the most general context is infeasible, and hence, incomplete, yet tractable methods are more appropriate. Also for association rules condensed representations are of great interest [25, 31]. Because of the simulation of association rules with itemsets as shown in Section 4, any condensed representation for frequent itemsets has direct implications for condensed representations on collections of association rules.

Frequent Itemset Mining Algorithms. A third application is improving the pruning of frequent itemset mining algorithms. All frequent set mining algorithms use the monotonicity rule to prune substantial parts of the search space. This monotonicity rule can be seen as a very simple example of deduction. Based on partial frequency information of some itemsets, bounds on the frequencies of yet to be counted sets are derived. If these bounds establish that a certain set must be certainly frequent or certainly infrequent, the count-

ing of it can be omitted in some cases. In the context of FREQSAT, frequency constraints can be used to model the frequency information gathered in previous scans over the database. The deduction can then be used to identify sets that are certainly frequent/infrequent. In [6, 7, 14], in some form, deduction rules are used in order to improve pruning and speed up frequent set mining algorithms.

7 Summary and Conclusion

In this paper, we discussed the FREQSAT-problem. This problem was shown to be **NP**-complete. It was also shown that restricting to exact frequencies $freq(I) = f$, instead of intervals $freq(I) \in [l, u]$ does not change the problem significantly. Furthermore, extension to arbitrary Boolean formulas instead of itemsets and to constraints on the confidence of association rules did not result in higher complexity, as they can all be simulated in FREQSAT. A result of this quite unexpected expressive power of FREQSAT is that the bounds on the frequency of itemsets implied by some given frequency constraints cannot be approximated efficiently.

Another indication of the complexity of FREQSAT comes from the fact that any axiomatization of FREQSAT must have infinite arity. It was shown, however, that there exists a recursive axiomatization, and a method to construct locally complete axioms was given.

Finally, it was discussed that the study of the FREQSAT problem has applications in privacy preserving data mining, condensed representations, and frequent itemset mining algorithms in general.

Acknowledgement The author would like to thank Dirk Van Gucht for his helpful comments and insights, especially for the material presented in Section 5.3.

This work has been partially funded by the EU contract IQ FP6-516169.

References

[1] S. Abiteboul, R. Hull, and V. Vianu. *Foundations of Databases*. Addison-Wesley, 1995.

[2] F. Afrati, A. Gionis, and H. Mannila. Approximating a collection of frequent sets. In *Proc. KDD Int. Conf. Knowledge Discovery in Databases*, pages 12–19, 2004.

[3] R. Agrawal, T. Imilienski, and A. Swami. Mining association rules between sets of items in large databases. In *Proc. ACM SIGMOD Int. Conf. Management of Data*, pages 207–216, 1993.

[4] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proc. ACM SIGMOD Int. Conf. Management of Data*, pages 439–450, 2000.

[5] M. Atzori, F. Bonchi, F. Giannotti, and D. Pedreschi. Blocking anonymity threats raised by frequent itemset mining. In *Proc. IEEE Int. Conf. on Data Mining*, pages 561–564, 2005.

[6] Y. Bastide, R. Taouil, N. Pasquier, G. Stumme, and L. Lakhal. Mining frequent patterns with counting inference. *SIGKDD Explorations*, 2(2):66–75, 2000.

[7] R. J. Bayardo. Efficiently mining long patterns from databases. In *Proc. ACM SIGMOD Int. Conf. Management of Data*, pages 85–93, 1998.

[8] R. J. Bayardo, B. Goethals, and M.J. Zaki, editors. *Proc. of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations (FIMI 2004)*. CEUR-WS.org, 2004.

[9] J.-F. Boulicaut, A. Bykowski, and C. Rigotti. Approximation of frequency queries by means of free-sets. In *Proc. PKDD Int. Conf. Principles of Data Mining and Knowledge Discovery*, pages 75–85, 2000.

[10] J.-F. Boulicaut, A. Bykowski, and C. Rigotti. Free-sets: A condensed representation of boolean data for the approximation of frequency queries. *Data Min. Knowl. Discov.*, 7(1):5–22, 2003.

[11] T. Calders. *Axiomatization and Deduction Rules for the Frequency of Itemsets*. PhD thesis, University of Antwerp, Belgium, 2003.

[12] T. Calders. Computational complexity of itemset frequency satisfiability. In *Proc. PODS Int. Conf. Principles of Database Systems*, pages 143–154, 2004.

[13] T. Calders. Deducing bounds on the support of itemsets. In *Database Technologies for Data Mining*, pages 214–233. Springer LNAI 2682, 2004.

[14] T. Calders and B. Goethals. Mining all non-derivable frequent itemsets. In *Proc. PKDD Int. Conf. Principles of Data Mining and Knowledge Discovery*, pages 74–85. Springer, 2002.

[15] T. Calders and B. Goethals. Non-derivable itemset mining. *Data Mining and Knowledge Discovery*, 14(1):171–206, 2007.

[16] T. Calders and J. Paredaens. Axiomatization of frequent itemsets. *Theoretical Computer Science*, 290(1):669–693, 2003.

[17] T. Calders, C. Rigotti, and J-F. Boulicaut. A survey on condensed representations for frequent sets. In J-F. Boulicaut, L. de Raedt, and H. Mannila, editors, *Constraint-based mining and inductive databases*, volume 3848 of *LNCS*. Springer, 2005.

[18] X. Chen and M. E. Orlowska. A further study on inverse frequent set mining. In *Proc. ADMA Int. Conf. Advanced Data Mining and Applications*, pages 753–760, 2005.

[19] G.B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, 1963.

[20] C. Dwork. Ask a better question, get a better answer a new approach to private data analysis. In *Proc. ICDT Int. Conf. Database Theory*, pages 18–27, 2007.

[21] R. Fagin, J. Halpern, and N. Megiddo. A logic for reasoning about probabilities. *Information and Computation*, 87(1,2):78–128, 1990.

[22] A. M. Frisch and P. Haddawy. Anytime deduction for probabilistic logic. *Artificial Intelligence*, 69(1,2):93–112, 1994.

[23] G. Georgakopoulos, D. Kavvadias, and C. H. Papadimitriou. Probabilistic satisfiability. *Journal of Complexity*, 4:1–11, 1988.

[24] B. Goethals. Frequent set mining. In *The Data Mining and Knowledge Discovery Handbook*, chapter 17, pages 377–397. Springer, 2005.

[25] B. Goethals, J. Muhonen, and H. Toivonen. Mining non-derivable association rules. In *Proc. SIAM Int. Conf. on Data Mining*, 2005.

[26] T. Hailperin. *Sentential Probability Logic*. Lehigh University Press, 1996.

[27] P. Hansen and B. Jaumard. Probabilistic satisfiability. Les Cahiers du GERAD G-96-31, GERAD, 1996.

[28] P. Hansen, B. Jaumard, G.-B. D. Nguets, and M. P. de Aragäo. Models and algorithms for probabilistic and bayesian logic. In *Proc. IJCAI Int. Joint Conf. Artificial Intelligence*, pages 1862–1868, Montreal, Canada, 1995.

[29] J. Hipp, U. Güntzer, and G. Nakhaeizadeh. Algorithms for association rule mining - a general survey and comparison. *SIGKDD Explorations*, 2(1):58–64, 2000.

[30] M. Jaeger. Automatic derivation of probabilistic inference rules. *Journal of Approximate Reasoning*, 28(1):1–22, 2001.

[31] S. Jaroszewicz and D. A. Simivici. Pruning redundant association rules using maximum entropy principle. In *Proc. PaKDD Pacific-Asia Conf. on Knowledge Discovery and Data Mining*, pages 135–147, 2002.

[32] Y. Lindell and B. Pinkas. Privacy Preserving Data Mining. *Journal of Cryptology*, 15(3):177–206, 2002.

[33] T. Lukasiewicz. Local probabilistic deduction from taxonomic and probabilistic knowledge-bases over conjunctive events. *Journal of Approximate Reasoning*, 21:23–61, 1999.

[34] T. Lukasiewicz. Probabilistic logic programming with conditional constraints. *ACM Transactions on Computational Logic*, 2(3):289–339, 2001.

[35] H. Mannila and H. Toivonen. Multiple uses of frequent sets and condensed representations. In *Proc. KDD Int. Conf. Knowledge Discovery in Databases*, 1996.

[36] T. Mielikäinen. On inverse frequent set mining. In *2nd IEEE ICDM Workshop on Privacy Preserving Data Mining (PPDM)*, pages 18–23. IEEE, 2003.

[37] N. Nilsson. Probabilistic logic. *Artificial Intelligence*, 28:71–87, 1986.

[38] J. B. Paris. *The Uncertain Reasoner's Companion*. Tracts in Theoretical Computer Science 39. Cambridge University Press, 1994.

[39] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Discovering frequent

closed itemsets for association rules. In *Proc. ICDT Int. Conf. Database Theory*, pages 398–416, 1999.

[40] D. Pavlov, H. Mannila, and P. Smyth. Beyond independence: Probabilistic models for query approximation on binary transaction data. *IEEE Trans. on Knowledge and Data Engineering*, 15(6):1409–1421, 2003.

[41] P. Samarati. Protecting respondents' identities in microdata release. *IEEE Trans. on Knowledge and Data Engineering*, 13(6):1010–1027, 2001.

[42] B. Sayrafi and D. Van Gucht. Differential constraints. In *Proc. PODS Int. Conf. Principles of Database Systems*, 2005.

[43] P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*. Addison-Wesley, 2005.

[44] N. Tatti. Computational complexity of queries based on itemsets. *Information Processing Letters*, 98(5):183–187, 2006.

[45] N. Tatti. Safe projections of binary data sets. *Acta Informatica*, 42(8-9):617–638, 2006.

[46] Y. Wang and X. Wu. Approximate inverse frequent itemset mining: Privacy, complexity, and approximation. In *Proc. IEEE Int. Conf. on Data Mining*, 2005.

[47] X. Wu, Y. Wu, Y. Wang, and Y. Li. Privacy aware market basket data set generation: A feasible approach for inverse frequent set mining. In *Proc. SIAM Int. Conf. on Data Mining*, 2005.

[48] X. Yan, H. Cheng, J. Han, and D. Xin. Summarizing itemset patterns: a profile-based approach. In *Proc. KDD Int. Conf. Knowledge Discovery in Databases*, pages 314–323, 2005.

A Proof of Theorem 1

Definition 3 *Let $\mathcal{D}_1$ and $\mathcal{D}_1$ be two transaction database, and let k be a positive integer. Let M be* $\max\{tid \mid (tid, J) \in \mathcal{D}_1\}$. *$\mathcal{D}_1 \bigoplus \mathcal{D}_2$ denotes the following transaction database: $\mathcal{D}_1 \bigoplus \mathcal{D}_2 := \mathcal{D}_1 \cup \{(M + tid, J) \mid (tid, J) \in \mathcal{D}_2\}$. Hence, $\mathcal{D}_1 \bigoplus \mathcal{D}_2$ is the transaction database that consists of both the transactions of $\mathcal{D}_1$ and $\mathcal{D}_2$.*

Let $\mathcal{D}$ be a transaction database. Then $\bigoplus_k \mathcal{D} := \overbrace{((\ldots(\mathcal{D} \bigoplus \mathcal{D}) \bigoplus \ldots) \bigoplus \mathcal{D}}^{k \text{ times}}$. Hence, $\bigoplus_k \mathcal{D}$ is the transaction database that consists of k copies of $\mathcal{D}$.

Lemma 3 *For all itemsets I, databases $\mathcal{D}_1, \mathcal{D}_2$, and integers $k \geq 1$,*

$$freq\left(I, \mathcal{D}_1 \bigoplus \mathcal{D}_2\right) = \frac{|\mathcal{D}_1|}{|\mathcal{D}_1 + \mathcal{D}_2|} \cdot freq\left(I, \mathcal{D}_1\right) + \frac{|\mathcal{D}_2|}{|\mathcal{D}_1 + \mathcal{D}_2|} \cdot freq\left(I, \mathcal{D}_2\right) \quad ,$$
$$freq\left(I, \mathcal{D}_1\right) = freq\left(I, \bigoplus_k \mathcal{D}_1\right) \quad .$$

Hence, if $\mathcal{D}_1$ and $\mathcal{D}_2$ satisfy $\mathcal{C}$, then does $\mathcal{D}_1 \bigoplus \mathcal{D}_2$, and $\mathcal{D}_1$ satisfies $\mathcal{C}$ if and only if $\bigoplus_k \mathcal{D}_1$ does.

Theorem 1 *Let $\mathcal{C} = \{freq\left(I_j\right) \in [l_j, u_j], j = 1, \ldots, m\}$. $\mathcal{C}$ is in* FREQSAT *if and only if $\mathcal{EQ}(\mathcal{C})$ is.*

Proof Only if: Let $\mathcal{D}$ be a database that satisfies $\mathcal{C}$. Let k be the least common multiplier of the denominators of the l_j's and u_j's. Hence, k is an integer, and for all j, there exist integers L_j, U_j such that $l_j = \frac{L_j}{k}$, and $u_j = \frac{U_j}{k}$.

Let now, for all $j = 1 \ldots m$, $\mathcal{A}_j$ and $\mathcal{B}_j$ be subsets of $\bigoplus_k \mathcal{D}$ such that:

- $|\mathcal{A}_j| = L_j \cdot |\mathcal{D}|$, and all transactions in $\mathcal{A}_j$ contain I_j;
- $|\mathcal{B}_j| = (k - U_j) \cdot |\mathcal{D}|$, and none of the transactions in $\mathcal{B}_j$ contain I_j.

Such sets exists, since

$$\begin{aligned} |\{(tid, J) \in \bigoplus_k \mathcal{D} \mid I_j \subseteq J\}| &= k \cdot |\{(tid, J) \in \mathcal{D} \mid I_j \subseteq J\}| \\ &= k \cdot |\mathcal{D}| \cdot freq\left(I_j, \mathcal{D}\right) \\ &\in [k \cdot |\mathcal{D}| \cdot l_j, k \cdot |\mathcal{D}| \cdot u_j] \\ &= [L_j \cdot |\mathcal{D}|, U_j \cdot |\mathcal{D}|] \end{aligned}$$

We will now construct a database $\mathcal{D}'$ that satisfies $\mathcal{EQ}(\mathcal{C})$. This database is formed as follows: we start with $\bigoplus_k \mathcal{D}$, and to each transaction in $\mathcal{A}_j$, we add

a_j, and to each transaction in $\mathcal{B}_j$, we add b_j. Hence, to a transaction (tid, J), the items $A[(tid, J)] = \{a_j \mid (tid, J) \in \mathcal{A}_j, j = 1 \ldots m\}$, and $B[(tid, J)] = \{b_j \mid (tid, J) \in \mathcal{B}_j, j = 1 \ldots m\}$ are added. Thus, $\mathcal{D}'$ is the following database:

$$\{(tid, J \cup A[(tid, J)] \cup B[(tid, J)]) \mid (tid, J) \in \bigoplus_k \mathcal{D}\} \ .$$

$\mathcal{D}'$ satisfies $\mathcal{EQ}(\mathcal{C})$.

If: Suppose $\mathcal{D}$ satisfies $\mathcal{EQ}(\mathcal{C})$. Then $\mathcal{D}$ also satisfies $\mathcal{C}$. Indeed: because of the monotonicity principle, $l \leq \mathit{freq}\,(I_j \cup \{a_j\}, \mathcal{D}) \leq \mathit{freq}\,(I_j, \mathcal{D})$, and, since $(1 - u_j) \cdot |\mathcal{D}|$ transactions contain b_j ($\mathit{freq}\,(\{b_j\}, \mathcal{D}) = 1 - u_j$), and none of the transactions contains both I_j and b_j ($\mathit{freq}\,(I_j \cup \{b_j\}, \mathcal{D}) = 0$), at most $|\mathcal{D}| - (1 - u_j) \cdot |\mathcal{D}| = u_j \cdot |\mathcal{D}|$ transactions contain I_j. Thus, also $\mathit{freq}\,(I_j, \mathcal{D}) \leq u_j$. Restricting $\mathcal{D}$ to the items in $\bigcup_{j=1}^{m} I_j$ does not affect the frequency of the sets I_j. Hence there exists a database that satisfies $\mathcal{C}$, and thus $\mathcal{C}$ is in FREQSAT. □

B Proof of Theorem 2

Theorem 2 *$\mathcal{C}$ is in* FREQSAT *if and only if $\mathcal{PSAT}(\mathcal{C})$ is in* pSAT.
Proof Because of Theorem 1, it suffices to show that $\mathcal{EQ}(\mathcal{C})$ is in FREQSAT if and only if $\mathcal{PSAT}(\mathcal{EQ}(\mathcal{C}))$ is in pSAT.

Only if: Let $\mathcal{D}$ be a database that satisfies $\mathcal{EQ}(\mathcal{C})$. Let, for all subsets $I \subseteq \mathcal{I}$,

$$\phi_I(\mathcal{D}) \ := \ \frac{|\{(tid, J) \in \mathcal{D} \mid J = I\}|}{|\mathcal{D}|} \ .$$

Hence, $\phi_I(\mathcal{D})$ denotes the fraction of transactions of $\mathcal{D}$ that are of the form (tid, I). It is easy to see that $\mathit{freq}\,(I, \mathcal{D}) = \sum_{I \subseteq J \subseteq \mathcal{I}} \phi_J(\mathcal{D})$.

We associate with every truth assignment A over $x_i, i \in \mathcal{I}$, a set of items $I(A)$ as follows: $I(A) = \{i \in \mathcal{I} \mid A(x_i) = 1\}$. Consider now the following distribution over the truth assignments over $\{x_i \mid i \in \mathcal{I}\}$: for all $A \in \mathcal{A}$, $\Pi(A) = \phi_{I(A)}$. We claim that Π satisfies $\mathcal{PSAT}(\mathcal{EQ}(\mathcal{C}))$: for all $j = 1 \ldots m$,

$$\begin{aligned}
\mathrm{Prob}_\Pi(\varphi_j) &= \sum_{A \in \mathcal{A}} A(\varphi_j) \cdot \Pi(A) \ = \ \sum_{A \in \mathcal{A}} A\left(\bigwedge_{i \in I_j} x_i\right) \cdot \Pi(A) \\
&= \sum \{\Pi(A) \mid A \in \mathcal{A}, \forall i \in I_j : A(x_i) = 1\} \\
&= \sum \{\phi_{I(A)}(\mathcal{D}) \mid A \in \mathcal{A}, I_j \subseteq I(A)\} \\
&= \sum \{\phi_J(\mathcal{D}) \mid J \subseteq \mathcal{I}, I_j \subseteq J\} \\
&= \mathit{freq}\,(I_j, \mathcal{D}) \ = \ f_j \ = \ \pi_j
\end{aligned}$$

If: Suppose that $\mathcal{P}(\mathcal{EQ}(\mathcal{C}))$ has a solution, then there exists a solution in which all $\Pi(A)$ are rational numbers [23]. Let now D be the least common multiplier of the denominators of all $\Pi(A)$. That is, every $\Pi(A)$ can be written as $\frac{N_A}{D}$, with N_A an integer. We will construct a database $\mathcal{D}$ with D transactions that satisfies $\mathcal{EQ}(\mathcal{C})$. The database $\mathcal{D}$ consists of N_A transactions of the form $(tid, I(A))$, for every assignment A. The number of transactions adds up to D, since the $\Pi(A)$ add up to 1. Hence,

$$\mathcal{D} \;=\; \bigoplus_{A \in \mathcal{A}} \{(tid, I(A)) \mid tid = 1 \ldots N_A\} \;\; .$$

For the definition of $\oplus$ we refer to Appendix A.

Let for every itemset I, $A(I)$ be the assignment that assigns 1 to x_i if and only if $i \in I$. $\mathcal{EQ}(\mathcal{C})$ is satisfied by $\mathcal{D}$, since for all $j = 1 \ldots m$, it holds that:

$$\begin{aligned}
\mathit{freq}\,(I_j, \mathcal{D}) &= \frac{|\{(tid, J) \in \mathcal{D} \mid I \subseteq J\}|}{|\mathcal{D}|} \\
&= \frac{\sum\{N_A \mid A \in \mathcal{A}, I_j \subseteq I(A)\}}{D} \\
&= \frac{\sum\{D \cdot \Pi(A) \mid A \in \mathcal{A}, \forall i \in I_j : A(x_i) = 1\}}{D} \\
&= \sum \left\{ \Pi(A) \mid A \in \mathcal{A}, A\left(\bigwedge_{i \in I_j} x_i\right) = 1 \right\} \\
&= \mathrm{Prob}_\Pi \left(\bigwedge_{i \in I_j} x_i \right) \;=\; \pi_j \;=\; f_j
\end{aligned}$$

□

C Proof of Theorem 3

In this section wet give the full formal construction of the reduction $\mathcal{FSAT}$ followed by the proof of Theorem 3.

Formal Construction Let

$$\mathcal{P} = \{\mathit{freq}\,(\varphi_1) \in [l_1, u_1], \ldots, \mathit{freq}\,(\varphi_m) \in [l_m, u_m]\}$$

be a set of m *extended* frequency constraints with $\varphi_1, \ldots, \varphi_m$ Boolean formulas over the set of items $\{i_1, \ldots, i_n\}$. $SF(\varphi)$ denotes the set of all subformulas of φ. For example, $SF(i_1 \wedge \neg i_2) = \{i_1, i_2, \neg i_2, i_1 \wedge \neg i_2\}$ The set of items $\mathcal{I}$ over which we construct the `FREQSAT`-instance will be $\{t_\sigma, f_\sigma \mid \sigma \in SF(\varphi_j), j = 1 \ldots m\} \cup \{d\}$.

We define the reduction $\mathcal{F}SAT(\mathcal{P})$ in four steps:

(1) The constraints TF, that will allow for expressing negations. t_σ stands for "σ is true", and f_σ for "σ is false."

$$TF(\varphi_1, \ldots, \varphi_m) := \bigcup_{\substack{\sigma \in SF(\varphi_j) \\ j=1\ldots m}} \{freq(\{t_\sigma\}) = 0.5, freq(\{f_\sigma\}) = 0.5, freq(\{t_\sigma, f_\sigma\}) = 0\}$$

(2) Recursive definition of the *consistency constraints* $Cons$. Within the part of the database that contains item d, the truth values for the sub-formulas must be consistent.

$$\begin{aligned}
Confs(\varphi_1, \ldots, \varphi_m) &:= Cons(\varphi_1) \cup \ldots \cup Cons(\varphi_m) \\
Confs(\sigma \vee \psi) &:= freq(\{d, t_{\sigma\vee\psi}, f_\sigma, f_\psi\}) = 0, freq(\{d, f_{\sigma\vee\psi}, t_\sigma) = 0, \\
&\quad freq(\{d, f_{\sigma\vee\psi}, t_\psi\}) = 0\} \cup Cons(\sigma) \cup Cons(\psi) \\
Confs(\sigma \wedge \psi) &:= freq(\{d, f_{\sigma\vee\psi}, t_\sigma, t_\psi\}) = 0, freq(\{d, t_{\sigma\vee\psi}, f_\sigma) = 0, \\
&\quad freq(\{d, t_{\sigma\vee\psi}, f_\psi\}) = 0\} \cup Cons(\sigma) \cup Cons(\psi) \\
Confs(\neg\sigma) &:= freq(\{d, t_{\neg\sigma}, t_\sigma\}) = 0, freq(\{d, f_{\neg\sigma}, f_\sigma\}) = 0\} \\
&\quad \cup Cons(\sigma) \\
Confs(i) &= \{\}
\end{aligned}$$

(3) The constraints $Freq$ that express that $freq(\varphi_j)$ must be in $[l_j, u_j]$.

$$Freq(\{freq(\varphi_j) \in [l_j, u_j] \mid j = 1 \ldots n\}) := \left\{ freq\left(\{d, t_{\varphi_j}\}\right) \in \left[\frac{l_j}{2}, \frac{u_j}{2}\right] \mid j = 1 \ldots m \right\}$$

(4) The reduction $\mathcal{F}$ itself:

$$\begin{aligned}\mathcal{F}SAT(\{\mathit{freq}\,(\varphi_j) \in [l_j, u_j] \mid j = 1 \ldots n\}) \;:=\; & \\ & \{\mathit{freq}\,(\{d\}) = 0.5\} \cup TF(\varphi_1, \ldots, \varphi_m) \\ & \cup\, Cons(\varphi_1, \ldots, \varphi_m) \\ & \cup\, Freq(\varphi_1, \ldots, \varphi_m, \pi_1, \ldots, \pi_m)\end{aligned}$$

Lemma 4 *Let $\mathcal{D}$ be a database that satisfies $TF(\varphi) \cup Cons(\varphi)$. Then, for all transactions (tid, J) such that $d \in J$, there exists a unique truth assignment A over the variables of φ such that t_σ is in J if and only if $A(\sigma) = 1$, and f_σ is in J if and only if $A(\sigma) = 0$.*

Proof Because of $TF(\varphi)$, for every subformula σ, the frequency constraints $\mathit{freq}\,(\{t_\sigma\}, \mathcal{D}) = 0.5$, $\mathit{freq}\,(\{f_\sigma\}, \mathcal{D}) = 0.5$, and $\mathit{freq}\,(\{t_\sigma, f_sigma\}, \mathcal{D}) = 0.5$ hold. Therefore, every transaction of $\mathcal{D}$ contains either t_σ, or f_σ, but not both.

Let now $T = (tid, J)$ be a transaction that contains d. Since the variables $i_1, \ldots, i_n$ of φ are subformulas of φ themselves, for every $j = 1 \ldots n$, J contains either t_{i_i}, or f_{i_i}, but not both. Let A_J now be the following truth assignment: $A_J(i_i) = 1$ if and only if $t_{i_i} \in J$, for all $i = 1 \ldots n$. Clearly, if there exists a truth assignment A that is consistent with $t_\sigma \in J$ if and only if $A(\sigma) = 1$, it can only be A_J. Therefore, the assignment associated with T will be unique.

We still need to show that A_J is consistent with the other subformulas σ. Hence, for every $\sigma \in SF(\varphi)$, we need to show that $t_\sigma \in J$ if and only if $A_J(\sigma) = 1$. Since $t_\sigma \in J$ and $f_\sigma \in J$ are mutual exclusive, it follows then that if $A_J(\sigma) = 0$, f_σ must be in J. We show this claim by induction on the structure of the subformula σ. The base case is trivially true; A_J was defined such that $A_J(i_i) = 1$ if and only if $t_{i_i} \in J$. The general case is split in three parts:

$\sigma = \neg\sigma_1$**:** By induction, we can assume that $A_J(\sigma_1) = 1$ if and only if $t_{\sigma_1} \in J$. Assume $A_J(\neg\sigma_1) = 0$. Then $A_J(\sigma_1) = 1$, and hence $t_{\sigma_1} \in J$. We need to show that J does not contain $t_{\neg\sigma}$. This requirement is indeed enforced by the following constraint in $Cons(\varphi)$: $\mathit{freq}\,(\{d, t_{\sigma_1}, t_{\neg\sigma_1}\}) = 0$. Hence, $\mathcal{D}$ cannot have a transaction that simultaneously contains d, t_{σ_1}, and $t_{\neg\sigma_1}$. Since T already contains d and t_{σ_1}, it therefore does not contain $t_{\neg\sigma_1}$. The case $A_J(\sigma_1) = 0$ can be proven in a similar fashion, using $\mathit{freq}\,(\{d, f_{\sigma_1}, f_{\neg\sigma_1}\}) = 0$.

$\sigma = \sigma_1 \wedge \sigma_2$**:** By induction, we can assume that $A_J(\sigma_i) = 1$ if and only if $t_{\sigma_i} \in J$, $i = 1, 2$. Assume that $A_J(\sigma_1 \wedge \sigma_2) = 0$. Then, at least one of σ_i, $A_J(\sigma_i) = 0$. We assume without loss of generality that $A_J(\sigma_1) = 0$ (the argument applies for σ_2 as well). Therefore, $t_{\sigma_1} \notin J$, and thus $f_{\sigma_1} \in J$. We need to show that $t_{\sigma_1 \wedge \sigma_2} \notin J$. This requirement is fulfilled by the following constraint in $Cons(\varphi)$: $\mathit{freq}\,(\{d, f_{\sigma_1}, t_{\sigma_1 \wedge \sigma_2}\}) = 0$. Since T already contains

d and f_{σ_1}, T cannot contain $t_{\sigma_1 \wedge \sigma_2}$.

The case $A_J(\sigma_1 \wedge \sigma_2) = 1$ is proved in a similar fashion, using *freq* $(\{d, t_{\sigma_1}, t_{\sigma_2}, f_{\sigma_1 \wedge \sigma_2}\}) = 0$.

$\sigma = \sigma_1 \vee \sigma_2$**:** Similar to $\sigma_1 \wedge \sigma_2$, using *freq* $(\{d, t_{\sigma_1}, f_{\sigma_1 \vee \sigma_2}\}) = 0$, and *freq* $(\{d, f_{\sigma_1}, f_{\sigma_2}, t_{\sigma_1 \vee \sigma_2}\}) = 0$.

□

Corollary 4 *Let $\mathcal{D}$ be a database with $TF(\varphi_1, \ldots, \varphi_m) \cup Cons(\varphi_1, \ldots, \varphi_m)$ satisfied. Then, for all transactions (tid, J) such that $d \in J$, there exists a unique truth assignment A_J over the variables of $\varphi_1, \ldots, \varphi_m$ such that t_σ is in J if and only if $A_J(\sigma) = 1$, and f_σ is in J if and only if $A_J(\sigma) = 0$.*

Proof Since $\mathcal{D}$ satisfies $TF(\varphi_1, \ldots, \varphi_m) \cup Cons(\varphi_1, \ldots, \varphi_m)$, $\mathcal{D}$ also satisfies $TF(\varphi_j) \cup Cons(\varphi_j)$, $j = 1 \ldots m$. Therefore, for every transaction $T = (tid, J)$ with $d \in J$, there exist truth assignments $A_1, \ldots, A_m$, such that for all $j = 1 \ldots m$, for every subformula σ of φ_j, $A_j(\sigma) = 1$ if and only if $t_\sigma \in J$. Since these truth assignments A_J are uniquely determined by the presence or absence of $t_{i_1}, \ldots, t_{i_n}$ in J, the truth assignments must agree on the common variables and subexpressions. Hence, there exists one unique assignment A_J such that t_σ is in J if and only if $A_J(\sigma) = 1$, and f_σ is in J if and only if $A_J(\sigma) = 0$. □

Definition 4 *Let $\mathcal{D}$ be a transaction database, and let d be an item.*

- *$\mathcal{D}^{-d}$ denotes the following transaction database:*

$$\mathcal{D}^{-d} := \{(tid, J \setminus \{d\}) \mid (tid, J) \in \mathcal{D}, d \in J\} .$$

- *$\mathcal{D}^{+d}$ denotes the following transaction database:*

$$\mathcal{D}^{+d} := \{(tid, J \cup \{d\}) \mid (tid, J) \in \mathcal{D}\} .$$

- *$\sigma_d \mathcal{D}$ denotes the following transaction database:*

$$\sigma_d \mathcal{D} := \{(tid, J) \in \mathcal{D} \mid d \in J\} .$$

Theorem 3 *$\mathcal{P} = (\varphi_1, \ldots, \varphi_m, \pi_1, \ldots, \pi_m)$ is in* `pSAT` *if and only if $\mathcal{FSAT}(\mathcal{P})$ is in* `FREQSAT`.

Furthermore, $ENT_\varphi(\mathcal{P}) = [l, u]$, iff $ENT_{\{d, t_\varphi\}}(\mathcal{FSAT}(\mathcal{P})) = [l/2, u/2]$.

Proof We assume that the variables used in $\varphi_1, \ldots, \varphi_m$ are $i_1, \ldots, i_n$. Let SF be the set of subformulas of $\varphi_1, \ldots, \varphi_m$. We need to show that there exists a database $\mathcal{D}$ such that $\mathcal{D}$ satisfies $\mathcal{FSAT}(\varphi_1, \ldots, \varphi_m, \pi_1, \ldots, \pi_m)$ if and only if there exists a probability distribution Π over the variables of $\varphi_1, \ldots, \varphi_m$ such that $\text{Prob}(\varphi_j) = \pi_j$, for $j = 1 \ldots m$. Furthermore, in the constructions we will use to show this result, it will always be the case that *freq* $(\{d, t_\varphi\}, \mathcal{D}) =$

$\text{Prob}_\Pi(\varphi)/2$, for any Boolean formula over $i_1, \ldots, i_n$, and hence, $ENT_\varphi(\mathcal{P}) = [l, u]$, if and only if $ENT_{\{d,t_\varphi\}}(\mathcal{FSAT}(\mathcal{P})) = [l/2, u/2]$.

If: Let Π be a probability distribution that satisfies $(\varphi_1, \ldots, \varphi_m, \pi_1, \ldots, \pi_m)$. We construct a database $\mathcal{D}_\Pi$ that satisfies $\mathcal{FSAT}(\varphi_1, \ldots, \varphi_m, \pi_1, \ldots, \pi_m)$.

Associate with every truth assignment A over $i_1, \ldots, i_n$, a set of items $I(A)$ as follows:

$$I(A) \;=\; \{t_\sigma \mid A(\sigma) = 1, \sigma \in SF\} \cup \{f_\sigma \mid A(\sigma) = 0, \sigma \in SF\} \;.$$

Let D be the least common multiplier of the denominators of $\{\Pi(A) \mid A \in \mathcal{A}\}$. Hence, for all assignments A, $N_A = D \cdot \Pi(A)$ is a positive integer. Let $\mathcal{D}$ now be the following database:

$$\mathcal{D} \;=\; \bigoplus_{A \in \mathcal{A}} \{(tid, I(A)) \mid tid = 1 \ldots N_A\} \;.$$

Thus, $\mathcal{D}$ consists of N_A transactions with set of items $I(A)$, for every truth assignment A. Notice incidently that in $\mathcal{D}$, $A_{I(A)} = A$. $\mathcal{D}$ has D transactions.

For every set of items I, let $\overline{I}$ be the smallest set of items that contains t_σ, if and only if I contains f_σ, and contains t_σ if and only if I contains t_σ. That is,

$$\overline{I} \;=\; \{t_\sigma \mid f_\sigma \in I\} \cup \{f_\sigma \mid t_\sigma \in I\} \;.$$

Let $\overline{\mathcal{D}}$ be the following transaction database:

$$\overline{\mathcal{D}} \;=\; \{(tid, \overline{J}) \mid (tid, J) \in \mathcal{D}\} \;.$$

The following database $\mathcal{D}_\Pi$ satisfies $\mathcal{FSAT}(\varphi_1, \ldots, \varphi_m, \pi_1, \ldots, \pi_m)$:

$$\mathcal{D}_\Pi \;=\; \mathcal{D}^{+d} \oplus \overline{\mathcal{D}} \;.$$

Notice that $\mathcal{D}_\Pi$ has $2 \cdot D$ transactions.

(1) $\mathcal{D}_\Pi$ satisfies $TF(\varphi_1, \ldots, \varphi_m)$: Every transaction $T = (tid, J)$ of $\mathcal{D}^{+d}$ contains t_σ if and only if $A_T(\sigma) = 1$, and f_σ if and only if $A_T(\sigma) = 0$. Therefore, for every $\sigma \in SF$, T contains either t_σ, or f_σ, but not both. The same is true for $\overline{\mathcal{D}}$, since a transaction $(tid, \overline{I})$ in $\overline{\mathcal{D}}$ that contains both t_σ and f_σ would imply that there is a transaction (tid, I) in $\mathcal{D}$ that contains both t_σ and f_σ as well. Therefore, for every $\sigma \in SF$, $\mathcal{D}_\Pi$ satisfies $\mathit{freq}\,(\{t_\sigma, f_\sigma\}) = 0$. Because of the way $\mathcal{D}$ and $\overline{\mathcal{D}}$ are constructed, for all $\sigma \in SF$, $\mathit{freq}\,(t_\sigma, \mathcal{D}_1) = \mathit{freq}\left(f_\sigma, \overline{\mathcal{D}_1}\right)$, and $\mathit{freq}\,(f_\sigma, \mathcal{D}) = \mathit{freq}\left(t_\sigma, \overline{\mathcal{D}}\right)$. Since every transaction of $\mathcal{D}$ and $\overline{\mathcal{D}}$ contain t_σ, or f_σ, but not both, we have as well $\mathit{freq}\,(\{t_\sigma\}, \mathcal{D}) + \mathit{freq}\,(\{f_\sigma\}, \mathcal{D}) = 1$, and $\mathit{freq}\left(\{t_\sigma\}, \overline{\mathcal{D}}\right) + \mathit{freq}\left(\{f_\sigma\}, \overline{\mathcal{D}}\right) = 1$. Henceforth,

$$\begin{aligned} \mathit{freq}\left(\{t_\sigma\}, \mathcal{D}_\Pi\right) &= \frac{|\mathcal{D}|}{|\mathcal{D}_\Pi|}\mathit{freq}\left(\{t_\sigma\}, \mathcal{D}\right) + \frac{|\overline{\mathcal{D}}|}{|\mathcal{D}_\Pi|}\mathit{freq}\left(\{t_\sigma\}, \overline{\mathcal{D}}\right) \\ &= \frac{1}{2}\mathit{freq}\left(\{t_\sigma\}, \mathcal{D}\right) + \frac{1}{2}\mathit{freq}\left(\{f_\sigma\}, \mathcal{D}\right) \;=\; 1/2 \end{aligned}$$

We can show in a similar fashion that $\mathit{freq}\left(\{t_\sigma\}, \mathcal{D}_\Pi\right) = 0.5$. Hence, $\mathcal{D}_\Pi$ satisfies $\{\mathit{freq}\left(\{t_\sigma\}\right) = 0.5, \mathit{freq}\left(\{f_\sigma\}\right) = 0.5\}$ for every $\sigma \in SF$.

(2) $\mathcal{D}$ satisfies $Cons(\varphi_1, \ldots, \varphi_m)$: This follows directly from the fact that the transactions that contain d have as set of items $I(A) \cup \{d\}$, with A a truth assignment. Indeed, t_σ and $t_{\neg\sigma}$ can never occur together in a transaction T of $\mathcal{D}$, since this would imply that $A_T(\sigma) = 1$ and $A_T(\neg\sigma) = 1$ at the same time.

(3) $\mathcal{D}$ satisfies $\mathit{freq}\left(\{d\}\right) = 0.5$: $\mathit{freq}\left(\{d\}, \mathcal{D}_\Pi\right) = \frac{|\mathcal{D}|}{|\mathcal{D}_\Pi|} = 0.5$.

(4) $\mathcal{D}$ satisfies $\mathit{freq}\left(\{d, \varphi_j\}\right) = \frac{\pi_j}{2}$, for all $j = 1 \ldots m$:

$$\begin{aligned} \mathit{freq}\left(\{d, t_{\varphi_j}\}, \mathcal{D}_\Pi\right) &= \frac{|\mathcal{D}|}{|\mathcal{D}_\Pi|}\mathit{freq}\left(\{d, t_{\varphi_j}\}, \mathcal{D}\right) \\ &= \frac{1}{2}\mathit{freq}\left(\{t_{\varphi_j}\}, \mathcal{D}\right) \\ &= \frac{1}{2}\frac{|\{(tid, J) \in \mathcal{D} \mid t_{\varphi_j} \in J\}|}{|\mathcal{D}|} \\ &= \frac{|\bigoplus_{A\in\mathcal{A}}\{(tid, I(A)) \mid tid = 1 \ldots N_A, t_{\varphi_j} \in I(A)\}|}{2 \cdot D} \\ &= \sum_{\substack{A\in\mathcal{A} \\ A(\varphi_j)=1}} \frac{D \cdot \Pi(A)}{2 \cdot D} \\ &= \frac{1}{2}\sum_{\substack{A\in\mathcal{A} \\ A(\varphi_j)=1}} (\Pi(A)) \;=\; \frac{\mathrm{Prob}_\Pi(\varphi_j)}{2} \;=\; \frac{\pi_j}{2} \end{aligned}$$

Hence, $\mathcal{D}$ satisfies $\mathcal{F}SAT(\varphi_1, \ldots, \varphi_m, \pi_1, \ldots, \pi_m)$.

Only If: Let $\mathcal{D}$ be a database that satisfies $\mathcal{F}SAT(\varphi_1, \ldots, \varphi_m, \pi_1, \ldots, \pi_m)$. We will construct a probability distribution Π over $i_1, \ldots, i_n$ such that $\mathrm{Prob}_\Pi(\varphi_j) = \pi_j$, for all $j = 1 \ldots m$. Let Π be defined as follows:

$$\forall A \in \mathcal{A} : \Pi(A) = \phi_{I(A)}(\sigma_d\mathcal{D}) \;\; .$$

(Recall that $\phi_I(\mathcal{D})$ denotes $\frac{|\{tid \mid (tid,I)\in\mathcal{D}\}|}{|\mathcal{D}|}$)

We show that Π is (1) well-defined (i.e., the probabilities sum to 1), and (2) has the desired properties (i.e., $\mathrm{Prob}_\Pi(\varphi_j) = \pi_j$).

(1) Π is well-defined: This amounts to showing that $\sum_{A\in\mathcal{A}} \Pi(A) = 1$. Because of Corollary 4, every transaction T of $\sigma_d\mathcal{D}$ can be written as $I(A_T)$, and thus, we have:

$$\begin{aligned}\sum_{A\in\mathcal{A}} \Pi(A) &= \sum_{A\in\mathcal{A}} \phi_{I(A)}(\sigma_d\mathcal{D}) \\ &= \sum_{A\in\mathcal{A}} \frac{|\{(tid,J)\in\sigma_d\mathcal{D} \mid J = I(A)\}|}{|\sigma_d\mathcal{D}|} \\ &= \frac{\sum_{A\in\mathcal{A}} |\{(tid,J)\in\sigma_d\mathcal{D} \mid J = I(A)\}|}{|\sigma_d\mathcal{D}|} \\ &= \frac{|\sigma_d\mathcal{D}|}{|\sigma_d\mathcal{D}|} \;=\; 1\end{aligned}$$

(2) $\text{Prob}_\Pi(\varphi_j) = \pi_j$, $j = 1 \ldots m$: because of Corollary 4, for every transaction T in $\sigma_d\mathcal{D}$, it holds that t_σ is in T, if and only if $A_T(\sigma) = 1$. Hence,

$$\begin{aligned}\text{Prob}_\Pi(\varphi_j) &= \sum_{\substack{A\in\mathcal{A}\\ A(\varphi_j)=1}} \Pi(A) \;=\; \sum_{\substack{A\in\mathcal{A}\\ A(\varphi_j)=1}} \phi_{I(A)}(\sigma_d\mathcal{D}) \\ &= \sum_{\substack{A\in\mathcal{A}\\ t_{\varphi_j}\in I(A)}} \phi_{I(A)}(\sigma_d\mathcal{D}) \;=\; \mathit{freq}\left(\{t_{\varphi_j}\},\sigma_d\mathcal{D}\right) \\ &= \frac{\mathit{freq}\left(\{t_{\varphi_j}\},\mathcal{D}\right)}{\mathit{freq}\left(\{d\},\mathcal{D}\right)} \;=\; 2\cdot\frac{\pi_j}{2} \;=\; \pi_j\end{aligned}$$

□

D Proof of the Multiplication Lemma

Multiplication Lemma *If a database $\mathcal{D}$ satisfies $\mathcal{MULT}_{n_1}(\varphi_1^1,\varphi_2^1)\cup\ldots\cup\mathcal{MULT}_{n_\ell}(\varphi_1^\ell,\varphi_2^\ell)$, then for all $j = 1\ldots\ell$, $n_j\cdot\mathit{freq}\left(\varphi_1^j,\mathcal{D}\right) = \mathit{freq}\left(\varphi_2^j,\mathcal{D}\right)$.*

There exists a database $\mathcal{D}$ that satisfies $\mathcal{C}$ and with for all $j = 1\ldots\ell$, $n_j\cdot\mathit{freq}\left(\varphi_1^j,\mathcal{D}\right) = \mathit{freq}\left(\varphi_2^j,\mathcal{D}\right)$ if and only if there exists a database $\mathcal{D}$ that satisfies $\mathcal{C}\cup\mathcal{MULT}_{n_1}(\varphi_1^1,\varphi_2^1)\cup\ldots\cup\mathcal{MULT}_{n_\ell}(\varphi_1^\ell,\varphi_2^\ell)$.

The proof of this important lemma is divided into a couple of lemma's.

Lemma 5 *If a database $\mathcal{D}$ satisfies $\epsilon(\varphi_1,\varphi_2)$, then $\mathit{freq}\,(\varphi_1,\mathcal{D}) = \mathit{freq}\,(\varphi_2,\mathcal{D})$.*

There exists a database $\mathcal{D}$ that satisfies $\mathcal{C}$ and with $\mathit{freq}\,(\varphi_1,\mathcal{D}) = \mathit{freq}\,(\varphi_2,\mathcal{D})$ if and only if there exists a database $\mathcal{D}$ that satisfies $\mathcal{C}\cup\epsilon(\varphi_1,\varphi_2)$.

Proof Let $\mathcal{D}$ be a database that satisfies $\epsilon(\varphi_1,\varphi_2)$. Because for $i,j = 1,2$,

$$\mathit{freq}\,(\varphi_i,\mathcal{D}) \;=\; \mathit{freq}\,(\varphi_i\wedge\varphi_j,\mathcal{D}) + \mathit{freq}\,(\varphi_i\wedge\neg\varphi_j,\mathcal{D}) \quad ,$$

$\mathit{freq}\,(\varphi_1,\mathcal{D}) = \mathit{freq}\,(\varphi_2,\mathcal{D})$ if and only if $\mathit{freq}\,(\varphi_1\wedge\neg\varphi_2,\mathcal{D}) = \mathit{freq}\,(\varphi_2\wedge\neg\varphi_1,\mathcal{D})$.

Since $\mathcal{D}$ satisfies $\epsilon(\varphi_1, \varphi_2)$, $\mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2 \wedge r, \mathcal{D}) = 0$, and thus, there are no transactions in $\mathcal{D}$ that simultaneously satisfy $\varphi_1 \wedge \neg\varphi_2$ and contain r. Therefore, $\mathit{freq}\,((\varphi_1 \wedge \neg\varphi_2) \vee r, \mathcal{D}) = \mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2, \mathcal{D}) + \mathit{freq}\,(r, \mathcal{D})$. Together with $\mathit{freq}\,((\varphi_1 \wedge \neg\varphi_2) \vee r, \mathcal{D}) = 0.5$, this gives

$$\mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2, \mathcal{D}) = 0.5 - \mathit{freq}\,(r, \mathcal{D}) \quad .$$

We can similarly show that $\mathit{freq}\,(\varphi_2 \wedge \neg\varphi_1, \mathcal{D}) = 0.5 - \mathit{freq}\,(r, \mathcal{D})$, and hence,

$$\mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2, \mathcal{D}) = \mathit{freq}\,(\varphi_2 \wedge \neg\varphi_1, \mathcal{D}) \quad .$$

For the second claim, the if-direction follows trivially from the first claim. For the only-if direction, assume that $\mathcal{D}$ satisfies $\mathcal{C}$ and has $\mathit{freq}\,(\varphi_1, \mathcal{D}) = \mathit{freq}\,(\varphi_2, \mathcal{D})$. We assume without loss of generality that $|\mathcal{D}|$ is even (If $|\mathcal{D}|$ is odd, we can switch to $\oplus_2\mathcal{D}$.)

Because $\mathit{freq}\,(\varphi_1) = \mathit{freq}\,(\varphi_1 \wedge \varphi_2) + \mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2)$,

$$\mathit{freq}\,(\varphi_2) = \mathit{freq}\,(\varphi_1 \wedge \varphi_2) + \mathit{freq}\,(\neg\varphi_1 \wedge \varphi_2), \text{ and}$$

$$\mathit{freq}\,(\varphi_1) = \mathit{freq}\,(\varphi_2) \quad ,$$

$\mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2)$ must equal $\mathit{freq}\,(\varphi_1 \wedge \varphi_2) + \mathit{freq}\,(\neg\varphi_1 \wedge \varphi_2)$. Furthermore, because no transaction can simultaneously satisfy $\varphi_1 \wedge \neg\varphi_2$ and $\varphi_2 \wedge \neg\varphi_1$,

$$\mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2, \mathcal{D}) = \mathit{freq}\,(\varphi_2 \wedge \neg\varphi_1, \mathcal{D}) \leq 0.5 \quad .$$

Let fr be $0.5 - \mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2, \mathcal{D})$. It now suffices to add the item r to $fr \cdot |\mathcal{D}|$ transactions that do neither satisfy $\varphi_1 \wedge \neg\varphi_2$, nor $\neg\varphi_1 \wedge \varphi_2$. This addition is possible: first of all, $fr \cdot |\mathcal{D}| = (1/2 - \mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2, \mathcal{D})) \cdot |\mathcal{D}|$ is a positive integer, because $|\mathcal{D}|$ is even, and $\mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2, \mathcal{D}) \cdot |\mathcal{D}|$ equals the number of transactions that satisfy $\varphi_1 \wedge \neg\varphi_2$. Secondly, there are $fr \cdot |\mathcal{D}|$ transactions that do not satisfy any of $\varphi_1 \wedge \neg\varphi_2$ and $\neg\varphi_1 \wedge \varphi_2$;

$$\begin{aligned} & fr + \mathit{freq}\,(\varphi_1 \wedge \neg\varphi_2, \mathcal{D}) + \mathit{freq}\,(\varphi_2 \wedge \neg\varphi_1, \mathcal{D}) \\ &= \; 0.5 + \mathit{freq}\,(\varphi_2 \wedge \neg\varphi_1, \mathcal{D}) \\ &\leq \; 1 \quad . \end{aligned}$$

The database resulting from this addition of r satisfies $\epsilon(\varphi_1, \varphi_2)$. □

Lemma 6 *If a database $\mathcal{D}$ satisfies $\epsilon(\varphi_1^1, \varphi_2^1) \cup \ldots \cup \epsilon(\varphi_1^\ell, \varphi_2^\ell)$, then for all $j = 1 \ldots \ell$, $\mathit{freq}\left(\varphi_1^j, \mathcal{D}\right) = \mathit{freq}\left(\varphi_2^j, \mathcal{D}\right)$.*

There exists a database $\mathcal{D}$ satisfying $\mathcal{C}$ and for all $j = 1 \ldots \ell$, $\mathit{freq}\left(\varphi_1^j, \mathcal{D}\right) = \mathit{freq}\left(\varphi_2^j, \mathcal{D}\right)$ if and only if there exists a database $\mathcal{D}$ that satisfies $\mathcal{C} \cup \epsilon(\varphi_1^1, \varphi_2^1) \cup$

$\ldots \cup \epsilon(\varphi_1^\ell, \varphi_2^\ell)$.

Proof This lemma follows easily from the way in which the databases in the proof of Lemma 5 are constructed. □

Lemma 7 *If a database $\mathcal{D}$ satisfies $\delta(\varphi_1, \varphi_2)$, then $2 \cdot freq(\varphi_1, \mathcal{D}) = freq(\varphi_2, \mathcal{D})$.*

There exists a database $\mathcal{D}$ that satisfies $\mathcal{C}$ and with $2 \cdot freq(\varphi_1, \mathcal{D}) = freq(\varphi_2, \mathcal{D})$ if and only if there exists a database $\mathcal{D}$ that satisfies $\mathcal{C} \cup \delta(\varphi_1, \varphi_2)$.

Proof Let $\mathcal{D}$ be a database that satisfies $\delta(\varphi_1, \varphi_2)$. Because of Lemma 6, this implies that $freq(k_1, \mathcal{D}) = freq(k_2, \mathcal{D}) = freq(\varphi_1)$. Because of $freq(k_1 \wedge k_2) = 0$, there are no transactions in $\mathcal{D}$ that contain both k_1 and k_2, and thus, $freq(k_1 \vee k_2, \mathcal{D}) = freq(k_1, \mathcal{D}) + freq(k_2, \mathcal{D}) = 2 \cdot freq(\varphi_1, \mathcal{D})$. Finally, $\varphi_2 = k_1 \wedge k_2$ makes sure that $freq(\varphi_2, \mathcal{D}) = freq(k_1 \vee k_2, \mathcal{D}) = 2 \cdot freq(\varphi_1, \mathcal{D})$.

The if-part of the second claim follows trivially from the first claim. For the only-if part: assume that $\mathcal{D}$ is a database that satisfies $\mathcal{C}$ and with $2 \cdot freq(\varphi_1, \mathcal{D}) = freq(\varphi_2, \mathcal{D})$. Select $\mathcal{S}_1, \mathcal{S}_2 \subseteq \mathcal{D}$ such that $\mathcal{S}_1$ and $\mathcal{S}_2$ are disjoint, $|\mathcal{S}_1| = |\mathcal{S}_2|$, and $\mathcal{S}_1 \cup \mathcal{S}_2$ is exactly the set of transactions that satisfy φ_2.

Let now $\mathcal{D}'$ be the database that is formed, starting from $\mathcal{D}$, and adding k_1 to the transactions in $\mathcal{S}_1$, and k_2 to the transactions in $\mathcal{S}_2$. $\mathcal{D}'$ satisfies $\mathcal{C} \cup \delta(\varphi_1, \varphi_2)$. □

Lemma 8 *If a database $\mathcal{D}$ satisfies $\delta(\varphi_1^1, \varphi_2^1) \cup \ldots \cup \delta(\varphi_1^\ell, \varphi_2^\ell)$, then for all $j = 1 \ldots \ell$, $2 \cdot freq\left(\varphi_1^j, \mathcal{D}\right) = freq\left(\varphi_2^j, \mathcal{D}\right)$.*

There exists a database $\mathcal{D}$ that satisfies $\mathcal{C}$ and with for all $j = 1 \ldots \ell$, $2 \cdot freq\left(\varphi_1^j, \mathcal{D}\right) = freq\left(\varphi_2^j, \mathcal{D}\right)$ if and only if there exists a database $\mathcal{D}$ that satisfies $\mathcal{C} \cup \delta(\varphi_1^1, \varphi_2^1) \cup \ldots \cup \delta(\varphi_1^\ell, \varphi_2^\ell)$.

Proof This lemma follows directly from the way in which the databases in the proof of Lemma 7 are constructed. □

The multiplication lemma now follows directly from Lemma 8.

E Proof of Theorem 7

Theorem 7 *Every axiomatization for* `FREQSAT` *that does not include an axiom that involves the frequency of all nonempty itemsets is incomplete. Therefore, for no k does there exist a k-ary sound and complete axiomatization for* `FREQSAT`.

Proof Let n be an arbitrary number. We construct a `FREQSAT` problem $\mathcal{C}$ over

the set $\mathcal{I} = \{i_1, \ldots, i_n\}$, such that (a) $\mathcal{C}$ is not satisfiable, but, (b) every strict subset of $\mathcal{C}$ is satisfiable. Furthermore, $\mathcal{C}$ contains one expression $\mathit{freq}\,(I) = f_I$ for every $I \subseteq \mathcal{I}$.

We assume that n is even. (a similar system can be found for n odd) Let $\mathcal{C}$ be

$$\left\{ \mathit{freq}\,(I) = \frac{2^{(n-|I|)}}{(2^n)-1} \;\middle|\; \emptyset \subset I \subset \mathcal{I} \right\} \bigcup \{\mathit{freq}\,(\mathcal{I}) = 0\}$$

For all $I \neq \emptyset$, we have:

$$\begin{aligned}
\sigma_{\mathcal{I}}(I) &= \sum_{I \subseteq K \subset \mathcal{I}} (-1)^{|K-I|} \frac{2^{(n-|K|)}}{(2^n)-1} \\
&= \sum_{k=|I|}^{n-1} (-1)^{k-|I|} \binom{n-|I|}{k-|I|} \frac{2^{(n-k)}}{(2^n)-1} \\
&= \frac{1}{(2^n)-1} \sum_{k=0}^{n-|I|-1} (-1)^k \binom{n-|I|}{k} 2^{((n-|I|)-k)} \\
&= \frac{1}{(2^n)-1} (1-(-1)^{n-|I|})
\end{aligned}$$

Hence, for all $I \neq \emptyset$, $\sigma_I\mathcal{I}$ equals 0 if $|I|$ is even, and 2 if $|I|$ is odd. For $I = \emptyset$, we get:

$$\begin{aligned}
\sigma_{\mathcal{I}}(\emptyset) &= \sum_{K \subset \mathcal{I}} (-1)^{|K|} \frac{2^{(n-|K|)}}{(2^n)-1} \\
&= \left(\sum_{k=1}^{n-1} (-1)^k \binom{n-|I|}{k} \frac{2^{(n-k)}}{(2^n)-1} \right) + 1 \\
&= \left(\frac{1}{(2^n)} - \frac{2^n}{(2^n)-1} - \frac{(-1)^n}{(2^n)-1} \right) + 1 \\
&= -\frac{1}{(2^n)-1}
\end{aligned}$$

Thus, $\mathcal{C}$ is not satisfiable. However, for every nonempty set I, if we remove the expression with I from $\mathcal{C}$, the resulting system $\mathcal{C}'$ is satisfiable. Let I be odd: $\mathcal{C}' \cup \{\mathit{freq}\,(I) = \frac{2^{(n-|I|)}-1}{(2^n)}\}$ is satisfiable, if $I \neq \mathcal{I}$ is even, $\mathcal{C}' \cup \{\mathit{freq}\,(I) = \frac{2^{(n-|I|)}+1}{(2^n)}\}$ is satisfiable, and for $I = \mathcal{I}$, $\mathcal{C}' \cup \{\mathit{freq}\,(\mathcal{I}) = \frac{1}{(2^n)}\}$ is satisfiable. These claims can easily be proved by checking the changes in the sums $\sigma_{\mathcal{I}}(I)$ given above. □