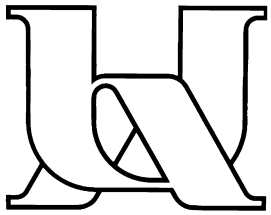


UNIVERSITEIT ANTWERPEN

Faculteit Wetenschappen

Informatica

Axiomatization and Deduction Rules for the Frequency of Itemsets

Proefschrift voorgelegd tot het behalen van de graad van
doctor in de Wetenschappen aan de Universiteit Antwerpen
te verdedigen door

Toon CALDERS

Promotor: prof. dr. Jan Paredaens

Antwerpen,
2003

Acknowledgements

I would like to thank the many people that contributed to the realization of this thesis.

First of all, I would like to thank my advisor, Jan Paredaens, for his guidance during my doctoral studies. Especially the opportunities he offered me for making contacts with other interesting researchers are greatly appreciated. Many thanks also to the other members of our research group ADReM and the departement for creating a stimulating environment.

I am very much in debt of Jef Wijsen, who, especially in the beginning of my doctoral research, was a great support. I thank him for patiently teaching me the basics of scientific research.

Another important influence for me was Bart Goethals. I enjoyed very much the many discussions we had. His questions, viewpoints, and insights helped me a lot. Collaboration with him resulted in many of the results covered in Chapter 4.

Other persons that supported me are Jan Van den Bussche, who regularly helped me with his impressive knowledge of scientific literature, and also Dirk Van Gucht, Raymond T. Ng, and Laks V.S. Lakshmanan. The many short visits to Dirk Van Gucht in Bloomington resulted in the material presented in Chapter 5. Also my three-month visit to Raymond T. Ng and Laks V.S. Lakshmanan in Vancouver was great stimulation for my scientific development. I am very grateful for this opportunity.

Also many thanks to my parents and family for their unconditional support and encouragement during the many years of my studies at the University of Antwerp.

Finally, I owe a lot to my wife An, who I would like to thank for the constant encouragement and support during my doctoral research and the writing of my thesis, day after day.

Thanks to all of you for making these four years into a positive and stimulating experience.

Contents

1

Introduction

1.1 Preliminaries

Advances in databases and technology make it possible to collect, store, and retrieve huge collections of data. Virtually every corporation and organization stores large amounts of data. However, not only the amount of data is important, but also the ability to analyze it. Indeed, for a company it is of vital importance to extract actionable knowledge and information from the data. This challenge is the motivation for *Data Mining* [42, 44], a relatively young research discipline on the edge of *Databases*, *Statistics*, and *Machine Learning*. In [44], data mining is defined as follows.

> Data Mining is the *analysis* of *large observational datasets* to find *unexpected relationships* and to *summarize* the data in novel ways that are *understandable and useful* to the data owner.

In this definition we identify the key phrases.

- *Analysis.* In data mining, one tries to identify important relations, patterns, and trends in databases in order to better understand the data. For this purpose, automatic tools are developed to help an analyst to get a better insight in the data, and to turn bulk data into useful knowledge.

- *Large observational datasets.* The datasets considered in data mining applications are usually very large. This characteristic makes that data mining algorithms must be highly efficient and scalable in order to process large datasets.

- *Unexpected relationships.* Unlike in traditional database systems, in data mining there is no such thing as an *exact* query that needs to be

answered. Ideally, the user only gives a type of relationship he or she wants to find in the data, and the data mining algorithm selects the actual patterns of that type that hold in the database.

- *Summarize.* Typically, the output of a data mining algorithm gives general characteristics of the dataset. These characteristics offer a different, more concise, view on the database.

- *Understandable and useful.* The output of a data mining algorithm is for a user only useful if it can be interpreted. This implies that models with great predictive value, but that are not understandable for humans, are not considered. We however stress that this requirement is not supported by all data mining researchers.

Frequent Itemset Problem One of the most prominent problems in data mining is the *Frequent Itemset Problem* [1]. The original context of frequent itemset mining was market basket analysis. Consider a retail store selling products from a set $\mathcal{I}$. For every customer of the store, the set of products purchased at once, called a *transaction*, is stored in a database $\mathcal{D}$. Based on this database an analyst wants to find out which sets of products are frequently purchased together. This setting is formalized in the frequent itemset problem. This problem is, given a threshold s and a database $\mathcal{D}$, find all subsets of $\mathcal{I}$, called *itemsets*, that are included in at least s of the sets in $\mathcal{D}$. This number of transactions in the database $\mathcal{D}$ in which an itemset I is included is called the *support of* I *in* $\mathcal{D}$, and is denoted by $support(I, \mathcal{D})$. The *frequency* of an itemset I in $\mathcal{D}$, denoted $freq(I, \mathcal{D})$, is the support of I in $\mathcal{D}$ divided by the number of transactions in $\mathcal{D}$. Itemsets with support higher than s are called *(s-)frequent*.

The frequent itemset problem is central in many data mining algorithms, such as association rules [1], sequential patterns [3], classification [4], emerging patterns [25], etc. Since the introduction of the frequent itemset problem in [1], many different approaches and algorithms have been proposed to find them in large databases, especially in the context of association rule mining [1, 2, 43]. For overviews of the different techniques, we refer to [43, 48, 81], and [35, Ch. 2].

Despite its simple description, the frequent itemset problem is far from trivial. It was shown in [39] that given a support threshold s, a number k, and a transaction database $\mathcal{D}$, the decision problem asking whether there is a s-frequent itemset in $\mathcal{D}$ of size k is **NP**-complete.

Monotonicity Principle All algorithms for mining frequent itemsets use the following *monotonicity principle* [63].

> Let $I_1 \subseteq I_2$ be two itemsets. In every transaction database $\mathcal{D}$, the frequency of I_2 will be at most as high as the frequency of I_1.

Many times this simple *rule of deduction* has been used successfully. The best example is the well-known Apriori-algorithm [2]. To exploit monotonicity as much as possible, the Apriori-algorithm starts with counting the singleton itemsets in a single pass over the database. In a second pass over the database, only itemsets $\{i_1, i_2\}$ such that $\{i_1\}$ and $\{i_2\}$ were found s-frequent are considered. All other itemsets of size 2 are *pruned*, since, as we can derive with the monotonicity principle, they cannot be s-frequent. In a third pass over the database, the algorithm proceeds with the itemsets of size 3 that only contain subsets of size 2 that are s-frequent. Thus, itemset $\{i_1, i_2, i_3\}$ is counted only if $\{i_1, i_2\}$, $\{i_1, i_3\}$, and $\{i_2, i_3\}$ are all s-frequent. This iteration continues until no more new frequent itemsets are found. The search for frequent itemsets by the Apriori-algorithm can thus be seen as an interleaving of a *counting phase* and a *meta phase*. In the counting phase, the frequencies of some predetermined itemsets, the so-called *candidates* are counted. In the meta phase the results of the counting phase are evaluated. Based on the monotonicity principle, some itemsets are *a-priori* —that is, without counting them in the database— excluded. These observations also apply to other frequent itemset mining algorithms such as DIC [11] and FPGrowth [43]. Since the introduction of the Apriori-algorithm, many improvements have been proposed. Most optimizations try to gain performance by reorganizing the input data in a format that allows more efficient counting, or by minimizing the number of scans over the database. Little work however went into improving pruning.

1.2 Subject of the Thesis

Research Question of this Thesis Although the monotonicity of frequency is commonly used, there is little previous work that tries to extend the monotonicity rule. This thesis studies deduction rules, such as the monotonicity principle, in general and on a theoretical basis; that is, without necessarily focussing on a particular algorithm. The central research question addressed is the following:

> Given information about the frequency of some itemsets $I_1, \ldots, I_n$,

what information can be derived about the frequencies of other itemsets?

Central in our approach to this problem is the notion of a *frequency constraint*. A frequency constraint is defined as an expression $freq(I) \in [l, u]$, with I an itemset, and l, u rational numbers between 0 and 1. A database $\mathcal{D}$ is said to satisfy this constraint if $freq(I, \mathcal{D}) \in [l, u]$. The given information is now modelled as a finite set of such frequency constraints. A set of frequency constraints $\mathcal{C}$ is said to *imply* the frequency constraint $freq(I) \in [l, u]$ if every database that satisfies all constraints in $\mathcal{C}$, also satisfies $freq(I) \in [l, u]$. That is, in every situation in which $\mathcal{C}$ holds, $freq(I) \in [l, u]$ must be true as well. For example, consider the set of frequency constraints

$$\mathcal{C} \ = \ \{ \ freq(\{a\}) \in [0.8, 0.9] \ , \ freq(\{b\}) \in [0.6, 0.8] \ \}$$

Because of the monotonicity principle, the frequency of $\{a, b\}$ can never be larger than the frequency of $\{b\}$. Since the frequency of $\{b\}$ is at most 0.8, $freq(\{a, b\}) \in [0, 0.8]$ is implied by $\mathcal{C}$. Another important notion is *tight* implication, that expresses that the interval $[l, u]$ is the best interval we can find for I, based on $\mathcal{C}$. The best interval here means that for every smaller interval $[l', u']$, it is no longer true that $freq(I) \in [l', u']$ is implied by $\mathcal{C}$. Consider again the set of frequency constraints $\mathcal{C}$ given above. Although $freq(\{a, b\}) \in [0, 0.8]$ is implied by $\mathcal{C}$, this implication is not tight. At least a fraction 0.8 of the transactions contains item a, and a fraction of 0.6 contains b. Therefore, there is an overlap of at least 0.4 between the transactions containing a, and the transactions containing b. Hence, the frequency of $\{a, b\}$ must be in the interval $[0.4, 0.8]$. It can be shown that this interval is tight by giving two databases $\mathcal{D}_1$, and $\mathcal{D}_2$, that both satisfy $\mathcal{C}$, and with $freq(\{a, b\}, \mathcal{D}_1) = 0.4$, and $freq(\{a, b\}, \mathcal{D}_2) = 0.8$. The following are examples of such databases.

$\mathcal{D}_1 \ =$

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

$\mathcal{D}_2 \ =$

TID	Items
1	a, b
2	a, b
3	a, b
4	a, b
5	

Suppose now for example that 0.4 is not a tight lower bound on the frequency of $\{a, b\}$. Then there exists a number l, strictly larger than 0.4, such that in every database that satisfies $\mathcal{C}$, the frequency of $\{a, b\}$ is at least l. This is however in contradiction with $freq(\{a, b\}, \mathcal{D}_1) = 0.4$. Actually, $\mathcal{D}_1$ is a

counterexample for all l strictly larger than 0.4. Databases such as $\mathcal{D}_1$ and $\mathcal{D}_2$ will be called *proof-databases*. They will play a very important role in the theory we develop.

A central problem studied in this thesis is the FREQSAT problem. It is defined as the problem of deciding whether a set of frequency constraints is satisfiable. We show that this problem is **NP**-complete.

Special Cases Because of the high complexity of FREQSAT, its usefulness in practice is limited. Therefore, we study special cases that have lower complexity, but that are still interesting from a practical, algorithmic point of view. The following cases are discussed.

- *Lower Bounds.* Only lower bounds on the frequency of the itemsets are considered; that is, we only use constraints of the form $freq(I) \in [l, 1]$. A set of such frequency constraints will be called a *system of frequent sets*. A system is said to be *complete* if all information in it is tight. For systems of frequent sets, FREQSAT is always satisfiable. We show that completeness of a system of frequent sets can be decided in polynomial time. We also describe three axioms, $\boldsymbol{\mathcal{F}_1}$, $\boldsymbol{\mathcal{F}_2}$, and $\boldsymbol{\mathcal{F}_3}$ for complete systems of frequent sets.

- *Upper Bounds.* We only allow constraints of the form $freq(I) \in [0, u]$. Again, FREQSAT is always satisfiable. Although this case seems very similar to the previous, it is much simpler. For example, the completeness of the system can be decided using only logarithmic space, and only two simple axioms, $\boldsymbol{\mathcal{IF}_1}$, and $\boldsymbol{\mathcal{IF}_2}$ are required.

- *Exact Frequencies, All Subsets.* This is the most interesting case. In this case, only bounds are derived on the frequency of an itemset of which the *exact* frequency of *all* its subsets is known. In that case the derivation of bounds can be done in polynomial time. This case is very interesting because the assumed information is exactly that information the Apriori-algorithm has for the candidate itemsets. Based on the deduction rules in this case, *derivable itemsets* are introduced. A set I is called *derivable in a database* $\mathcal{D}$ if its frequency is uniquely determined by the frequencies of its subsets. An algorithm is developed to find all non-derivable itemsets efficiently.

For each of the cases the complexity of the satisfiability problem and a complete axiomatization is studied. We also show for each case how to calculate the bounds on the frequency of the target itemsets.

Generic Technique We also give a generic technique that allows for deriving a complete set of axioms for specific cases. This method is based on the elimination algorithm for systems of linear inequalities of *Fourier* and *Motzkin* [67]. We show how a FREQSAT problem can be translated to a system of linear inequalities. In this system we then eliminate some of the variables. The resulting system will contain the axiomatization. For example, suppose that we want to derive bounds on the frequency of the set $\{b\}$, based on the information that $freq(\{a\}) = f_a$, and $freq(\{a, b\}) = f_{ab}$. Let x_a represent the transactions that have $\{a\}$ as set of items, x_b the ones that have $\{b\}$ as set of items, and x_{ab} the ones with $\{a, b\}$ as set of items. The (unknown) frequency of b is denoted by f_b. We translate this situation as the following linear inequalities:

$$(x_a + x_{ab} = f_a) \ \wedge \ (x_{ab} = f_{ab}) \ \wedge \ (x_b + x_{ab} = f_b)$$
$$\wedge \ (x_a \geq 0) \ \wedge \ (x_b \geq 0) \ \wedge \ (x_{ab} \geq 0) \wedge \ (x_a + x_b + x_{ab} \leq 1) \ .$$

In this system we eliminate x_a, x_b, and x_{ab}. The elimination results in the following, equivalent system:

$$(0 \leq f_a) \ \wedge \ (f_a \leq 1) \ \wedge \ (0 \leq f_b) \ \wedge \ (f_b \leq 1) \ \wedge \ (0 \leq f_{ab}) \ \wedge \ (f_{ab} \leq 1)$$
$$\wedge \ (f_{ab} \leq f_a) \ \wedge \ (f_{ab} \leq f_b) \ \wedge \ (f_{ab} \geq f_a + f_b - 1) \ .$$

Therefore, we can derive that the frequency of $\{b\}$ is in the interval

$$\left[\ \max\{0, f_{ab}\} \ , \ \min\{1, 1 + f_{ab} - f_a\} \ \right] \ .$$

This interval is tight.

Applications of Deducing Frequencies: Concise Representations Based on the deduction rules, we can identify redundancies in the set of frequent itemsets. Especially the special case with exact frequencies is interesting in this perspective. We show how we can use the deduction to build *concise representations* [62] of the set of frequent itemsets. A concise representation of the frequent itemsets is a subset that contains the same frequency information. That is: from the concise representation we can derive exactly which itemsets are frequent, and, if a set is frequent, then we can derive its exact frequency from the representation. Because the goal of the deduction rules we study is to derive frequencies as exact as possible, there is a clear link between our work and concise representations. Other concise representations proposed in the literature are *free sets* [9], *closed sets* [72, 8, 75], and *disjunction-free sets* [12]. We show how these types of concise representations can be expressed with the deduction rules we introduce. In this way, our deduction rules and approach to concise representations becomes a unifying framework for many proposals in the literature.

Related Work In artificial intelligence literature, probabilistic logics are studied intensively [41, 71]. The link with this thesis 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; that is, we can consider the transaction database as an underlying probability structure, and the itemsets as conjunctions of basic events. In Chapter 7, we discuss related work in artificial intelligence in length. Especially the links with the *probabilistic logic* of *Nilsson* [68], the logic for reasoning about probabilities of *Fagin, Hailperin and Megiddo* [26], and the work of *Lukasiewicz* [60] receive special attention.

Also connections with data mining are discussed. Interesting there is the MAXMINER algorithm of *Bayardo* [6], and the PASCAL algorithm of *Bastide et al.* [5] for mining frequent itemsets. These two algorithms also use some kind of deduction of bounds on the frequency of itemsets.

Another important body of related work concerns *concise representations* [62]. The work about concise representations in Chapter 6 is compared to other proposals such as *free sets* [9], *closed sets* [72, 8, 75], and *disjunction-free sets* [12].

Connections between the deduction rules in Chapter 4 and combinatorial theory such as *Bonferonni inequalities* [7, 50, 31], and *statistical data protection* [24] are explored as well.

Organization of the Thesis In Chapter 2, we give a formal definition of the problems we study in this thesis and we discuss the complexity of the FREQSAT problem. The special cases are studied in Chapter 3 (Lower and upper bounds), and Chapter 4 (Exact frequencies). In Chapter 5, we introduce the generic technique based on the elimination algorithm of Fourier and Motzkin. We discuss the applications in Chapter 6. Related work is discussed in Chapter 7. We conclude the thesis in Chapter 8 with a summary of the results and interesting further work.

2

Problem Description

In this chapter we formally introduce the research question we discuss in this thesis. First we will introduce the frequent itemset problem. In order to illustrate the mechanisms of algorithms that solve this problem, we discuss a stripped-down version of the Apriori-algorithm. This version will be used throughout the thesis to illustrate the practical use of the theory we develop.

We then introduce the FREQSAT problem as an abstraction of the pruning strategy in the frequent itemset mining algorithms. We show that this problem is **NP**-complete. On the one hand, this **NP**-completeness result is very important, because it is the motivation for the study of more restricted problems with nicer algorithmic properties in the next chapters. On the other hand, the proof of the **NP**-completeness gives a characterization of the FREQSAT problem in terms of the satisfiability of a linear program. This link with linear programming will be central in many of the problems we discuss later on. Besides FREQSAT, we study the complexity of entailment problems as well.

Furthermore, we discuss the difference between support and frequency. In the thesis we use frequency constraints instead of support constraints. Repercussions of this choice with respect to the results obtained are given.

We end the chapter with a practical example to illustrate the different properties we will study in the next chapters.

Bibliographic Note The proof of the **NP**-completeness of FREQSAT will be very similar to the proof of **NP**-completeness of probabilistic satisfiability given in [34]. Although no straightforward reduction from the probabilistic satisfiability problem to our problem seems to be available, many of the techniques used in [34] apply directly to our problem.

For the study of the complexity of the entailment problems, we were inspired by the systematic approach of *Lukasiewicz* in [61] to the complexity

of logical programming with conditional constraints. However, because in our model conditional probabilities are not allowed, the completeness-results presented in this chapter are stronger than the ones in [61].

Parts of this chapter were already published in [14].

2.1 Frequent Itemset Mining

Frequent Itemset Mining [1] was first mentioned in the context of market basket analysis. Imagine a retail store selling products. For each customer, at the check-out counter, all products purchased are scanned and stored as a set in a database. Such a set of products is called a *transaction*. Based on this data, analysts try to get a better understanding of the shopping behavior of the customers. An important question in this context is which products are frequently sold together. Such a popular set of products is called a *frequent itemset*. The search for frequent itemsets is the basis of many data mining problems. As such, it is one of the core problems in data mining. In the literature, a whole spectrum of related problems has been studied, such as finding the largest frequent set [6], the most surprising frequent set [77], the most correlated set [66], etc. We concentrate however on the problem of finding all frequent itemsets.

2.1.1 Frequent Itemset Problem

Definition 1 *Let $\mathcal{I}$ be a finite set, called the* set of items.

- *A* transaction over $\mathcal{I}$ *is defined as a pair (tid, I) where I is a subset of $\mathcal{I}$ and tid is a natural number, called the* identifier.

- *A* transaction database $\mathcal{D}$ over $\mathcal{I}$ *is a finite set of transactions over $\mathcal{I}$, in which every transaction has a different identifier.*

- *A subset I of $\mathcal{I}$ is called an* itemset over $\mathcal{I}$. *We will say that a transaction (tid, J) over $\mathcal{I}$* contains *an itemset I over $\mathcal{I}$, denoted $I \subseteq (tid, J)$, if I is a subset of J.*

- *The* support *of an itemset I over $\mathcal{I}$ in a transaction database $\mathcal{D}$ over $\mathcal{I}$, denoted $support(I, \mathcal{D})$, is defined as the number of transactions T in $\mathcal{D}$ that contain I.*

- *The* frequency *of an itemset I in a transaction database $\mathcal{D}$, denoted $freq(I, \mathcal{D})$, is defined as*

$$freq(I, \mathcal{D}) =_{def} \frac{support(I, \mathcal{D})}{|\mathcal{D}|} .$$

□

In all that follows, $\mathcal{I}$ is the set of all items and $\mathcal{D}$ is the transaction database. In the rest of the text we will often denote an itemset I by the list of its elements; that is, $\{a, b, c\}$ is denoted by abc.

One of the central problems in this thesis is the *Frequent Itemset problem*.

Problem 1 Frequent Itemsets (FSET(**$\mathcal{D}$,**t)**).** *Given a transaction database $\mathcal{D}$ over $\mathcal{I}$ and a threshold $t \in [0, 1]$, find all itemsets I such that $freq(I, \mathcal{D}) \geq t$.* □

Example 1 *Consider the following transaction database $\mathcal{D}$ over the set of items $\{a, b, c, d\}$.*

$\mathcal{D} =$

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

The support of the itemset $\{a, d\}$ in $\mathcal{D}$ is 3 because 3 transactions contain both a and d; only the transaction with transaction identifier (TID) 1 does not contain d. The answer to the frequent itemset problem FSET*($\mathcal{D}$,0.5) is the set $\{\phi, a, b, d, ab, ad\}$.*

□

Even though the problem statement is very simple, the problem itself is not. The following theorem, proved in [39], gives a nice illustration of the inherent complexity of the frequent itemset problem.

Theorem 1 [39] *Given a transaction database $\mathcal{D}$, and a frequency threshold t.*

- *Deciding whether* FSET$(\mathcal{D}, t)$ *contains an itemset I with $|I| \geq k$ for a given k is* **NP***-complete.*

- *Calculating the cardinality of* FSET$(\mathcal{D}, t)$ *is* #**P***-hard* [1].

[1] Let Q be a polynomially balanced binary relation. The *counting problem* associated with Q is the following: Given x, how many y are there such that $(x, y) \in Q$? The output required is an integer in binary. #**P**(number-**P**), is the class of all counting problems associated with polynomially balanced, polynomial-time decidable relations.

Input: Database $\mathcal{D}$ over $\mathcal{I}$, frequency threshold t.
Output: Set $\mathcal{F}$ of t-frequent itemsets in $\mathcal{D}$.

(1) $C_1 := \{\{i\} \mid i \in \mathcal{I}\}$;
(2) $k := 1$; $\mathcal{F} := \{\phi\}$
(3) **while** $(C_k \neq \{\})$ **loop**
(4) Count the frequencies of the sets in C_k in one pass over $\mathcal{D}$;
(5) $L_k := \{I \in C_k \mid \mathit{freq}(I, \mathcal{D}) \geq t\}$;
(6) $\mathcal{F} := \mathcal{F} \cup L_k$;
(7) $C_{k+1} := \{I \subseteq \mathcal{I} \mid |I| = k+1, \forall J \subset I : J \in \mathcal{F}\}$;
(8) **end-loop**
(9) **return** $\mathcal{F}$;

Figure 2.1: Rough sketch of the Apriori algorithm

2.1.2 The Apriori Algorithm

One of the most important observations in frequent itemset mining is the *monotonicity principle* [63]:

> Let $I_1 \subseteq I_2$ be two itemsets. In every transaction database $\mathcal{D}$, the frequency of I_2 will be at most as high as the frequency of I_1.

Based on this principle we can prune the search space of the frequent itemset problem. If we know that a certain itemset I_1 is infrequent, then it is not necessary to explore the space of all supersets of I_1. This property is exploited as much as possible by the Apriori-algorithm. In Figure 2.1, we give a rough sketch of the algorithm. Most implementations of Apriori use advanced data structures for speeding up steps (4) and (7). Reducing the number of loops is a successful strategy as well. For our purpose however, we will restrict our attention to the stripped-down version of Apriori presented in Figure 2.1.

Apriori starts with the singleton-itemsets as *candidates* in step (1). These candidates are counted in a single scan over the database in step (4). The candidates that turn out frequent are stored to be outputted in the end (steps (5) and (6)). In step (7), new candidates are generated based on the old candidates that turned out to be frequent. In the k-th loop, all itemsets of size k that cannot be pruned using the monotonicity principle are considered as candidates. That is, the new set of candidates will consist of all itemsets of size $k+1$ such that there are no subsets that were infrequent. In this way the monotonicity principle is exploited as much as possible.

A logical question to ask is whether the pruning performed in the Apriori-algorithm is optimal; that is, given the information of the frequencies counted in the previous loops, do we prune away as many candidates as possible? The answer is negative as we show shortly. This question also applies to other frequent set mining algorithms; given information about the frequencies of some itemsets, what can we derive for candidate itemsets?

2.1.3 Apriori Does Not Prune Perfectly

We show that Apriori does not prune perfectly. The example also illustrates the general technique we use later on in the proofs.

Consider the following example:

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

$$\begin{array}{l} \mathit{freq}(a, \mathcal{D}) = \mathit{freq}(b, \mathcal{D}) = \mathit{freq}(c, \mathcal{D}) = \frac{2}{3} \\ \mathit{freq}(ab, \mathcal{D}) = \mathit{freq}(ac, \mathcal{D}) = \mathit{freq}(bc, \mathcal{D}) = \frac{1}{3} \end{array} \tag{2.1}$$

Suppose we are running the Apriori-algorithm with the minimal frequency threshold set to $\frac{1}{3}$. The algorithm will start with counting the supports of the singleton-itemsets $C_1 = \{a, b, c\}$. Since they are all frequent, Apriori will consider in its second loop the candidates $C_2 = \{ab, ac, bc\}$. Again all candidates are frequent, and thus, Apriori will count $C_3 = \{abc\}$ in its third loop. However, the following simple observation shows that from the frequencies counted so far, we can derive that abc is infrequent.

We encode the situation after the second loop as a linear programming instance. A similar representation is also used in [18, 13, 14]. Let for each itemset I, the *fraction* of I in $\mathcal{D}$, denoted $\mathcal{F}_I(\mathcal{D})$, be the fraction of transactions having I as set of items, that is,

$$\mathcal{F}_I(\mathcal{D}) \ =_{def} \ \frac{|\{(tid, J) \in \mathcal{D} \mid J = I\}|}{|\mathcal{D}|} \ .$$

We will omit $\mathcal{D}$ when clear from the context. For every database satisfying

the frequencies in (2.1), the following equalities must hold:

$$\begin{cases} \mathcal{F}_{\{\}} + \mathcal{F}_a + \mathcal{F}_b + \mathcal{F}_c + \mathcal{F}_{ab} + \mathcal{F}_{ac} + \mathcal{F}_{bc} + \mathcal{F}_{abc} = 1 & (\{\}) \\ \mathcal{F}_a + \mathcal{F}_{ab} + \mathcal{F}_{ac} + \mathcal{F}_{abc} = \frac{2}{3} & (a) \\ \mathcal{F}_b + \mathcal{F}_{ab} + \mathcal{F}_{bc} + \mathcal{F}_{abc} = \frac{2}{3} & (b) \\ \mathcal{F}_c + \mathcal{F}_{ac} + \mathcal{F}_{bc} + \mathcal{F}_{abc} = \frac{2}{3} & (c) \\ \mathcal{F}_{ab} + \mathcal{F}_{abc} = \frac{1}{3} & (ab) \\ \mathcal{F}_{ac} + \mathcal{F}_{abc} = \frac{1}{3} & (ac) \\ \mathcal{F}_{bc} + \mathcal{F}_{abc} = \frac{1}{3} & (bc) \\ \mathcal{F}_{\{\}}, \mathcal{F}_a, \mathcal{F}_b, \mathcal{F}_c, \mathcal{F}_{ac}, \mathcal{F}_{bc}, \mathcal{F}_{ab}, \mathcal{F}_{abc} \geq 0 \end{cases} \tag{2.2}$$

From this system we derive:

$$\begin{cases} \mathcal{F}_a + \mathcal{F}_{ac} = \frac{1}{3} & (a - ab) \\ \mathcal{F}_a + \mathcal{F}_{ab} = \frac{1}{3} & (a - ac) \\ \mathcal{F}_b + \mathcal{F}_{bc} = \frac{1}{3} & (b - ab) \\ \mathcal{F}_b + \mathcal{F}_{ab} = \frac{1}{3} & (b - bc) \\ \mathcal{F}_c + \mathcal{F}_{bc} = \frac{1}{3} & (c - ac) \\ \mathcal{F}_c + \mathcal{F}_{ac} = \frac{1}{3} & (c - bc) \end{cases} \tag{2.3}$$

The solution of system (2.3) is $\mathcal{F}_a = \mathcal{F}_b = \mathcal{F}_c = k, \mathcal{F}_{ab} = \mathcal{F}_{ac} = \mathcal{F}_{bc} = \frac{1}{3} - k$ with k a parameter. Because also $\mathcal{F}_{\{\}}+\mathcal{F}_a+\mathcal{F}_b+\mathcal{F}_c+\mathcal{F}_{ab}+\mathcal{F}_{ac}+\mathcal{F}_{bc}+\mathcal{F}_{abc} = 1$, we derive $\mathcal{F}_{\{\}} + \mathcal{F}_{abc} = 0$. Since neither $\mathcal{F}_{\{\}}$, nor $\mathcal{F}_{abc}$ can be negative, $\mathcal{F}_{abc}$ must be 0. Therefore, $freq(abc, \mathcal{D}) = 0$, and we know *a priori* that abc cannot be frequent. Nevertheless, Apriori does not prune abc. This example shows that pruning can be improved beyond monotonicity.

2.2 Problem FREQSAT

In this section we define *frequency constraints* as a mean to model information about frequencies. Deduction rules such as the monotonicity principle are captured by *implication* of frequency constraints. Finally, we generalize the problem of pruning candidate itemsets in an algorithm-independent way by the FREQSAT problem. The FREQSAT-problem is to decide, given

information about frequencies, whether there exists a database that is consistent with these frequencies. Considering the example above, we know that the following FREQSAT problem is not satisfiable.

$$\left\{ \begin{array}{lll} freq(a) = \frac{2}{3} \ , & freq(b) = \frac{2}{3} \ , & freq(c) = \frac{2}{3} \ , \\ freq(ab) = \frac{1}{3} \ , & freq(ac) = \frac{1}{3} \ , & freq(bc) = \frac{1}{3} \ , \\ freq(abc) \in \left[\frac{1}{3}, 1\right] & & \end{array} \right\}$$

Therefore, we can conclude that, given the information

$$\left\{ \begin{array}{lll} freq(a) = \frac{2}{3} \ , & freq(b) = \frac{2}{3} \ , & freq(c) = \frac{2}{3} \ , \\ freq(ab) = \frac{1}{3} \ , & freq(ac) = \frac{1}{3} \ , & freq(bc) = \frac{1}{3} \end{array} \right\} \ ,$$

it is not possible that *abc* is frequent.

2.2.1 Definition

Definition 2

- *A* frequency constraint over $\mathcal{I}$ *is an expression* $freq(I) \in [l, u]$*, with* I *an itemset over* $\mathcal{I}$*, and* l, u *rational numbers* [2] *between 0 and 1.*

- *A transaction database* $\mathcal{D}$ *over* $\mathcal{I}$ satisfies *the constraint* $freq(I) \in [l, u]$*, denoted* $\mathcal{D} \models freq(I) \in [l, u]$*, if*

$$l \leq freq(I, \mathcal{D}) \leq u \ .$$

- *A transaction database* $\mathcal{D}$ *satisfies a set of frequency constraints* $\mathcal{C}$*, denoted* $\mathcal{D} \models \mathcal{C}$ *if* $\mathcal{D}$ *satisfies every expression in* $\mathcal{C}$*.*

- *A set of frequency constraints* $\mathcal{C}$ implies *(or* entails*) a frequency constraint* $freq(I) \in [l, u]$*, denoted* $\mathcal{C} \models freq(I) \in [l, u]$*, if every database that satisfies* $\mathcal{C}$*, also satisfies* $freq(I) \in [l, u]$*.*

- *A set of frequency constraints* $\mathcal{C}$ tightly implies *(or* tightly entails*) a frequency constraint* $freq(I) \in [l, u]$*, denoted* $\mathcal{C} \models_{tight} freq(I) \in [l, u]$*, if* $\mathcal{C} \models freq(I) \in [l, u]$*, and if for every* l', u' *such that* $\mathcal{C} \models freq(I) \in [l', u']$*, it is true that* $[l, u] \subseteq [l', u']$*. Hence,* $[l, u]$ *is the smallest interval we can derive for* I *based on* $\mathcal{C}$*.*

[2] We use rational numbers instead of real numbers because frequencies will always be fractions. Also for computational issues rational numbers are more convenient because we can easily represent them in binary.

□

An essential problem when studying the implication of frequency constraints is the following *Frequency Satisfiability Problem.*

Problem 2 Frequency Satisfiability (FREQSAT($\mathcal{C}$)) *The* FREQSAT-problem *is, given a finite set*

$$\mathcal{C} \;=\; \{\mathit{freq}(I_j) \in [l_j, u_j] \mid j = 1 \ldots n\}$$

of frequency constraints, decide whether there exists a transaction database $\mathcal{D}$ over $\mathcal{I} = \bigcup_{j=1}^{n} I_j$ such that $\mathcal{D}$ satisfies $\mathcal{C}$. □

Notice that the sets I in the definition of the FREQSAT problem are arbitrary; that is, we do not require $I \subseteq \mathcal{I}$ for a fixed set $\mathcal{I}$. The reason for this is that we do not want to fix the number of items nor limit the cardinality of I when we study the computational complexity of the FREQSAT problem.

We will often use the expression "$\mathit{freq}(I) = p$" to denote the frequency constraint "$\mathit{freq}(I) \in [p, p]$".

Example 2 *The* FREQSAT-*problem*

$$\left\{\mathit{freq}(a) \in \left[\frac{1}{2}, 1\right], \mathit{freq}(bd) \in \left[\frac{1}{4}, \frac{1}{2}\right], \mathit{freq}(abc) \in \left[0, \frac{1}{2}\right], \mathit{freq}(bcd) \in \left[0, 0\right]\right\}$$

is satisfiable; the transaction database $\mathcal{D}$ in Example 1 satisfies this instance.

The FREQSAT-*problem*

$$\left\{\mathit{freq}(a) \in \left[0, \frac{1}{2}\right], \mathit{freq}(ab) \in \left[\frac{3}{4}, 1\right]\right\}$$

is not satisfiable, since, as stated by the monotonicity rule, the frequency of ab must always be smaller than or equal to the frequency of a. □

2.2.2 Computational Complexity

We now study the complexity of the FREQSAT problem. We show that the problem is **NP**-complete.

Theorem 2 *The* FREQSAT-*problem is* **NP**-*complete.*

FREQSAT **is in NP**

We show that FREQSAT is in **NP** by reducing it to an instance of *linear programming* [40, 67, 70] in which the number of equalities is polynomial in the size of the FREQSAT-problem. As in Subsection 2.1.3, the notion of *fraction* will be very important.

Definition 3 *Let $\mathcal{D}$ be a transaction database and I be a subset of $\mathcal{I}$. We define the* I-fraction of $\mathcal{D}$, *denoted $\mathcal{F}_I(\mathcal{D})$ as*

$$\mathcal{F}_I(\mathcal{D}) \ =_{def} \ \frac{|\{(tid, J) \in \mathcal{D} \mid J = I\}|}{|\mathcal{D}|} \ .$$

Hence, the I-fraction of $\mathcal{D}$ is the fraction of transactions having I as set of items. If $\mathcal{D}$ is clear from the context, we will write $\mathcal{F}_I$. □

This definition allows us to restate the frequency of an itemset I in terms of the different fractions in the transaction database.

Lemma 1 *Let $\mathcal{D}$ be a transaction database and I be an itemset over $\mathcal{I}$. Then the following holds.*

$$freq(I, \mathcal{D}) = \sum_{I \subseteq J \subseteq \mathcal{I}} \mathcal{F}_J \ .$$

Proof
Straightforward. □

Let $\mathcal{C} = \{freq(I_1) \in [l_1, u_1], \ldots, freq(I_n) \in [l_n, u_n]\}$ be a FREQSAT problem. $\mathcal{I}$ denotes $\bigcup_{i=1}^{n} I_i$. For each $I \subseteq \mathcal{I}$ we introduce a variable X_I. X_I is associated with the fraction $\mathcal{F}_I$. A solution of the linear programming instance we will introduce, specifies conditions that the fractions need to satisfy. A solution to the linear program specifies a transaction database.

Example 3 *Consider the following transaction database T of Example 1.*

$\mathcal{D} =$

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

$\mathcal{D}$ is specified by $\mathcal{F}_{abc} = \frac{1}{4}, \mathcal{F}_{abd} = \frac{1}{4}, \mathcal{F}_{ad} = \frac{1}{2}$, and for all other I, $\mathcal{F}_I = 0$.

□

The linear programming problem $P_1(\mathcal{C})$, associated with the FREQSAT problem $\mathcal{C}$, is specified as follows.

> Does there exist a $2^{|\mathcal{I}|}$-vector $(X_\phi, X_A, \ldots, X_\mathcal{I}) \geq 0$ such that the following system $\mathcal{P}(\mathcal{C})$ of inequalities is satisfied?

$$\mathcal{P}(\mathcal{C}) \ =_{def} \ \begin{cases} \sum_{I \subseteq \mathcal{I}} X_I = 1 \\ l_i \leq \sum_{I_i \subseteq I \subseteq \mathcal{I}} X_I \leq u_i \qquad \forall i = 1, \ldots, n \end{cases}$$

Lemma 2 *Let $\mathcal{C}$ be a set of frequency constraints over $\mathcal{I}$. There exists a transaction database $\mathcal{D}$ over $\mathcal{I}$ that satisfies $\mathcal{C}$, if and only if $\mathcal{P}(\mathcal{C})$ has a rational solution in the variables X_I, $I \in \mathcal{I}$.*

Proof
If. Consider a solution $X_I = s_I, \forall I \subseteq \mathcal{I}$ of the system $\mathcal{P}(\mathcal{C})$. Let d be the least common multiple of the denominators of the rational numbers s_I. Let now $\mathcal{D}$ be the transaction database that for all $I \in \mathcal{I}$ contains exactly $d.s_I$ transactions with as set of items I. Because of the equality

$$\sum_{I \subseteq \mathcal{I}} X_I = 1 \ ,$$

the total number of transactions equals d. Therefore, for all $I \subseteq \mathcal{I}$,

$$\mathit{freq}(I, \mathcal{D}) = \frac{\sum_{I \subseteq J \subseteq \mathcal{I}} d.s_J}{d} = \sum_{I \subseteq J \subseteq \mathcal{I}} s_J \ .$$

Because for every frequency constraint $\mathit{freq}(I) \in [l, u]$ the solution must also satisfy the inequalities

$$l \leq \sum_{I \subseteq J \subseteq \mathcal{I}} s_J \leq u \ ,$$

$\mathcal{C}$ is satisfied by $\mathcal{D}$.
Only If. Let $\mathcal{D}$ be a transaction database that satisfies $\mathcal{C}$. Then $X_I = \mathcal{F}_I(\mathcal{D})$, for all $I \subseteq \mathcal{I}$ is a solution for $\mathcal{P}(\mathcal{C})$. □

By adding $2n$ *slack variables* $S_1, \ldots, S_n, S'_1, \ldots, S'_n$, we transform the linear programming problem P_1 into the following, equivalent, problem P_2.

Does there exist a $(2^{|I|}+2n)$-vector

$$(X_\phi, X_A, \ldots, X_I, S_1, \ldots, S_n, S'_1, \ldots, S'_n) \geq 0$$

such that the following system of inequalities $\mathcal{P}'(\mathcal{C})$ is satisfied?

$$\mathcal{P}'(\mathcal{C}) \ =_{def} \ \begin{cases} \sum_{I \subseteq \mathcal{I}} X_I = 1 & \\ \sum_{I \supseteq I_i} X_I - S_i = l_i & \forall i = 1, \ldots, n \\ \sum_{I \supseteq I_i} X_I + S'_i = u_i & \forall i = 1, \ldots, n \end{cases}$$

The number of equalities in the program is $ne = (2n+1)$, and the number of variables is $nv = (2^{|I|} + 2n)$. We now use a result in linear programming theory [26], stating that, if an instance of linear programming with ne equalities and nv variables has a positive solution, then it also has a solution with at most nv nonzero variables.

Theorem 3 [26] *If a system of r equalities and/or inequalities with integer coefficients each of length at most l has a nonnegative solution, then it has a nonnegative solution with at most r entries positive, and where the size of each member of the solution is $\mathcal{O}(rl + r\log(r))$.* □

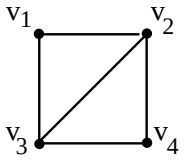
Thus, if the linear program P_2 has a solution, then the non-zero variables together with their values can serve as a *succinct certificate* [69, pp. 182]. Therefore, FREQSAT is in **NP**.

NP-Hardness of FREQSAT

The proof of **NP**-hardness is very much alike the one given for 2PSAT in [34]. We will reduce *graph 3-colorability* [33] to FREQSAT.

In the proof, the instances of FREQSAT we consider will be very specific. All intervals will be point-intervals $[f, f]$. This implies that even when we restrict ourselves to cases in which we know the frequencies of the itemsets $I_1, \ldots, I_n$ exactly, the satisfiability problem is still **NP**-complete. Even more surprising is the fact that we will only need sets of cardinality 1 or 2. Even in this very restricted case, the satisfiability problem remains **NP**-complete.

Given a graph $G = (V, E)$. G is said to be *3-colorable* if there exists a function $c : V \to \{1, 2, 3\}$ such that for each edge $[u, v]$ in E we have $c(u) \neq c(v)$. Such a function is called a *coloring* of the graph G.



TID	Items
1	$B_1,\ R_2,\ G_3,\ B_4$
2	$R_1,\ G_2,\ B_3,\ R_4$
3	$G_1,\ B_2,\ R_3,\ G_4$

Figure 2.2: Transaction database representing a solution to a 3-colorability problem

Given such a graph we construct an instance $\mathcal{C}(G)$ of FREQSAT as follows. We introduce $3|V|$ items: for each vertex $v \in V$ we consider the items B_v, R_v, G_v. B_v (R_v, G_v) stand for "vertex v has color blue (red, green)." The FREQSAT problem $\mathcal{C}(G)$ will contain the following frequency constraints.
For each vertex v:
$\mathit{freq}(\{B_v\}) = \frac{1}{3}$, $\mathit{freq}(\{R_v\}) = \frac{1}{3}$, $\mathit{freq}(\{G_v\}) = \frac{1}{3}$,
$\mathit{freq}(\{B_v, R_v\}) = 0$, $\mathit{freq}(\{B_v, G_v\}) = 0$, $\mathit{freq}(\{R_v, G_v\}) = 0$.
For each edge $[u, v]$:
$\mathit{freq}(\{B_u, B_v\}) = 0$, $\mathit{freq}(\{R_u, R_v\}) = 0$, $\mathit{freq}(\{G_u, G_v\}) = 0$. This reduction can clearly be carried out in logarithmic space.

Suppose $\mathcal{D}$ is a transaction database satisfying this FREQSAT problem, and T is a transaction in $\mathcal{D}$. The first six conditions make sure that for each vertex v, T contains exactly one of B_v, R_v, G_v. The last three conditions make sure that for each pair of vertices u, v that are connected by an edge, T cannot contain both B_u and B_v or both R_u and R_v or both G_u and G_v. Therefore, every transaction represents a valid coloring of the graph G. Since the empty transaction database is not a solution of $\mathcal{C}(G)$, it is true that if the FREQSAT problem is satisfiable, then there exists a coloring. This connection is illustrated in Figure 2.2.

Suppose G has a coloring c. We can construct the following transaction T: T contains B_v if $c(v) = 1$, R_v if $c(v) = 2$, and G_v if $c(v) = 3$. Consider now the colorings c' and c'' resulting from cyclically rearranging the colors $1, 2, 3$ in the coloring c. Also for c' and c'' we can construct transactions T' and T''. The transaction database $\mathcal{D} = \{T, T', T''\}$ satisfies the FREQSAT problem $\mathcal{C}(G)$.

Hence, the FREQSAT problem $\mathcal{C}(G)$ has a solution if and only if G has a 3-coloring, and thus is FREQSAT **NP**-hard.

Note In the proof we only need itemsets of cardinality 1 or 2, and point intervals. We can even restrict further to sets of cardinality 2 only, by adding two auxiliary items d_1, d_2. We add the frequency constraint $freq(\{d_1, d_2\}) = 1$, and every frequency constraint $freq(\{i\}) = p$ in $\mathcal{C}$ is replaced by the constraint $freq(\{i, d_1\}) = p$. It is easy to see that the resulting FREQSAT problem has a solution if and only if the original system $\mathcal{C}$ has one.

In the FREQSAT-problem, we restrict our frequency constraints to closed intervals. This choice is merely out of convenience, rather than fundamental. If we remove the requirement that the intervals are closed, the complexity results still obtain.

2.3 Graphical Interpretation of FREQSAT

The restatement of the FREQSAT-problem in last section allows for a graphical interpretation. Let $\mathcal{C}$ be the following set of frequency constraints.

$$\mathcal{C} \;=\; \{freq(I_1) \in [l_1, u_1], \ldots, freq(I_n) \in [l_n, u_n]\} \quad .$$

Lemma 2 states that $\mathcal{C}$ is satisfiable if and only if the following system of inequalities has a solution with all variables X_I positive.

$$\mathcal{P}(\mathcal{C}) \;=_{def}\; \begin{cases} \displaystyle\sum_{I \subseteq \mathcal{I}} X_I = 1 & \\ \displaystyle l_i \leq \sum_{I_i \subseteq I \subseteq \mathcal{I}} X_I \leq u_i & \forall i = 1, \ldots, n \end{cases}$$

Such a solution then corresponds to a transaction database, namely, the one with $\mathcal{F}_I = X_I$, for all $I \subseteq \mathcal{I}$.

It is with this system of linear inequalities $\mathcal{P}(\mathcal{C})$ that the graphical interpretation is associated. Every possible solution $(x_{\{\}}, \ldots, x_{\mathcal{I}})$ of $\mathcal{P}(\mathcal{C})$ is in fact a point in the $d = 2^{|\mathcal{I}|}$-dimensional space. We will call this d-dimensional space the *fraction-space*. Some of the points in the fraction-space correspond to a valid transaction database. These points are exactly the points in the set

$$V \;=_{def}\; \left\{ (X_{\{\}}, \ldots, X_{\mathcal{I}}) \;\middle|\; \begin{array}{l} \sum_{I \subseteq \mathcal{I}} X_I = 1, \\ X_I \geq 0 \quad \forall I \subseteq \mathcal{I} \end{array} \right\} \quad .$$

V is a closed, convex and bounded subset of the d-dimensional space, since it is the intersection of the hyperplane defined by $\sum_{I \subseteq \mathcal{I}} X_I = 1$, and the d half-spaces $X_I \geq 0$. Let $\mathcal{D}(p)$ be the database that corresponds to the point $p(p_{\{\}}, \ldots, p_{\mathcal{I}}) \in V$. The frequency of a set I in the database $\mathcal{D}(p)$ is

then $\sum_{I \subseteq J} p_J$. Hence, the databases $\mathcal{D}$ that satisfy a frequency constraint $freq(I) \in [l, u]$ correspond exactly with the points p in V with $\sum_{I \subseteq J} p_J \in [l, u]$. The set of points that satisfy $freq(I) \in [l, u]$ is again convex, bounded, and closed, since it is the intersection of V with the half-spaces defined by $\sum_{I \subseteq J} x_J \geq l$ and $\sum_{I \subseteq J} x_J \leq u$. The points that correspond to databases that satisfy $\mathcal{C}$ are thus exactly the points in the intersection of all these sets for each of the frequency constraints. Let $V(\mathcal{C})$ denote the set of points that correspond to databases that satisfy $\mathcal{C}$. Also $V(\mathcal{C})$ is closed, convex and bounded.

Example 4 *Consider the set of frequency constraints*

$$\mathcal{C} \;=\; \left\{ freq(a) \in \left[\frac{1}{2}, 1\right], freq(ab) \in \left[\frac{1}{4}, \frac{3}{4}\right] \right\} \;.$$

The set of points that correspond with databases that satisfy $\mathcal{C}$ is depicted in Figure 2.3. In this figure, only 3 of the 4 dimensions have been given. This is because the coordinates in this fourth dimension that corresponds to $\mathcal{F}_{\{\}}$ are redundant in the set V. Indeed, every point p in V has coordinates $(1 - p_a - p_b - p_{ab}, p_a, p_b, p_{ab})$. □

Suppose that we want to find the tight bounds on the frequency of an itemset I, given a set of frequency constraints $\mathcal{C}$. Since each database that satisfies $\mathcal{C}$ corresponds to a point in $V(\mathcal{C})$, and the frequency of I in $\mathcal{D}(p)$ equals $\sum_{I \subseteq J} p_J$, we can conclude that the possible frequency values for I, given $\mathcal{C}$ are exactly given by

$$F(I) \;=_{def}\; \left\{ \sum_{I \subseteq J} p_J \mid p \in V(\mathcal{C}) \right\} \;.$$

Thus, the set of possible values for the frequency of I is the image of the continuous function $f : p \to \sum_{I \subseteq J}$ on the set $V(\mathcal{C})$. It is well-known that the image of a closed and bounded (and hence compact) set through a continuous function is closed and bounded again. Therefore, $F(I)$ is a compact set, and thus an interval. Geometrically, the fact that $F(I)$ must be an interval can also be seen as follows. Because $V(\mathcal{C})$ is closed and bounded, the infinum and supremum of f on $V(\mathcal{C})$ are also minimum and maximum. Thus, we can find points p and q in $V(\mathcal{C})$, such that f reaches its minimum and maximum in respectively p and q. Let this minimum and maximum value of f be l and u. Because $V(\mathcal{C})$ is convex, every point $\alpha p + (1 - \alpha)q$ with $\alpha \in [0, 1]$, must be in $V(\mathcal{C})$ as well. The frequency of I in $\alpha p + (1 - \alpha)q$ equals

$$\sum_{I \subseteq J} (\alpha p_J + (1 - \alpha) q_J) = \alpha \sum_{I \subseteq J} p_J + (1 - \alpha) \sum_{I \subseteq J} q_J = \alpha l + (1 - \alpha) u \;.$$

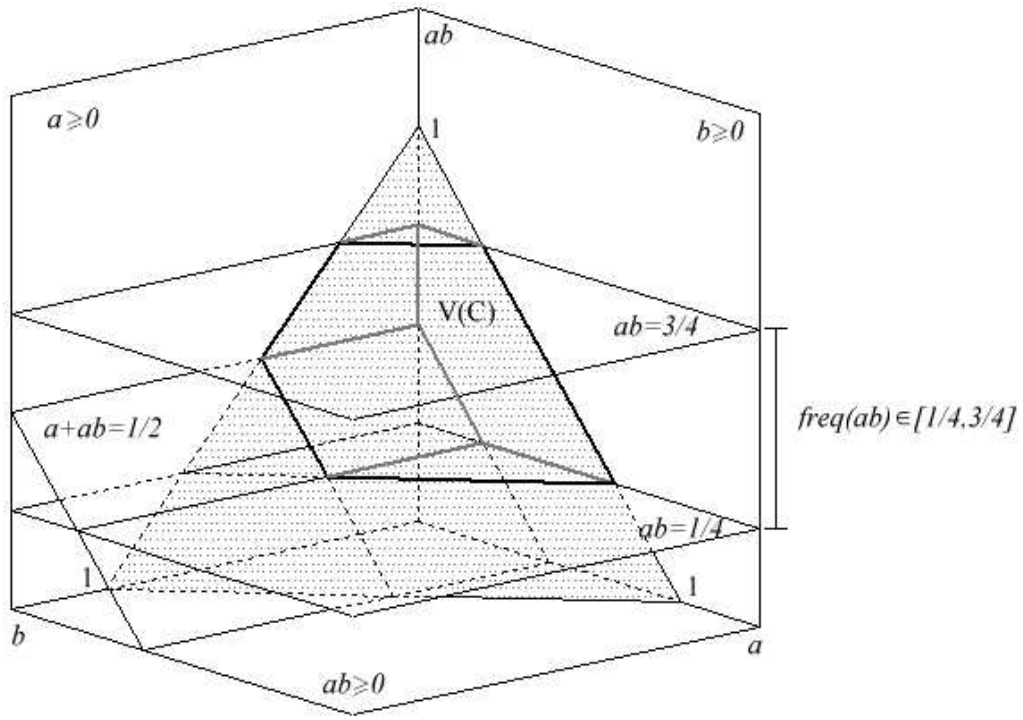


Figure 2.3: Fraction space

Thus, every value between l and u is possible as frequency of I.

2.4 Entailment of Frequency Constraints

Besides the decision problem FREQSAT, we also study the entailment problems FREQENT and T-FREQENT. Instead of asking whether a set of frequency constraints is satisfiable, in these problems we give a set of frequency constraints $\mathcal{C}$ plus a target frequency constraint and we ask if the target constraint is entailed (tightly entailed) by $\mathcal{C}$.

Problem 3 FREQENT($\mathcal{C}$, $freq(I) \in [l,u]$) *Given a set of frequency constraints $\mathcal{C}$, and a target frequency constraint $freq(I) \in [l,u]$. Decide whether $\mathcal{C} \models freq(I) \in [l,u]$.* □

Problem 4 T-FREQENT($\mathcal{C}$, $freq(I) \in [l,u]$) *Given a set of frequency constraint $\mathcal{C}$, and a frequency constraints $freq(I) \in [l,u]$. Decide whether $\mathcal{C} \models_{tight} freq(I) \in [l,u]$.* □

The complexity of these two problems is very much related to the complexity of FREQSAT. The following theorem gives the exact complexities of FREQENT and T-FREQENT.

Theorem 4

- FREQENT *is* **co-NP** *complete.*

- T-FREQENT *is* **DP**-*complete* [3].

Proof
Let $\mathcal{C}$ be a set of frequency constraints. On the one hand,

$$\mathcal{C} \models \mathit{freq}(I) \in [l, u]$$

if and only if $\mathcal{C} \cup \{\mathit{freq}(I) \in\ [0, l[\}$ and $\mathcal{C} \cup \{\mathit{freq}(I) \in\ ]u, 1]\}$ are both not satisfiable. On the other hand, $\mathcal{C}$ is satisfiable if and only if it is not true that

$$\mathcal{C} \models \mathit{freq}(\{\}) \in [1, 0]\ .$$

This implies that co-FREQSAT reduces to FREQENT, and FREQENT to co-FREQSAT, which proofs the first statement.

For the second statement, suppose that

$$\mathcal{C} \models_{\mathit{tight}} \mathit{freq}(I) \in [l, u]\ .$$

This is possible if and only if

$\mathcal{C} \models \mathit{freq}(I) \in [l, u]$, and	**co-NP**
$\mathcal{C} \cup \{\mathit{freq}(I) = l\}$ is satisfiable, and	**NP**
$\mathcal{C} \cup \{\mathit{freq}(I) = u\}$ is satisfiable.	**NP**

Therefore, T-FREQENT is in **DP**.

We show that T-FREQENT in **DP**-hard by reducing the SAT-UNSAT problem to it. The SAT-UNSAT problem is, given two formulas φ and φ', decide whether φ is satisfiable, and φ' is unsatisfiable. That is, the answer is "yes" if and only if φ is satisfiable and *at the same time* φ' is unsatisfiable. As in the traditional SAT-problem, φ and φ' are both boolean expressions in conjunctive normal form and each clause is a disjunction consisting of three literals. It is well-known that SAT-UNSAT is **DP**-complete [69, pp. 413].

[3] **DP** [69, pp. 412] is the complexity class that contains all languages L such that there exist languages $L_1 \in$ **NP** and $L_2 \in$ **coNP** with $L = L_1 \cap L_2$. The D in **DP** stands for *Difference*, since every language in **DP** is the set difference of two languages in **NP**.

Let $x_1, \ldots, x_k$ be the variables in φ, and $x'_1, \ldots, x'_{k'}$ be the variables in φ'. We assume without loss of generality that $\{x_1, \ldots, x_k\}$ and $\{x'_1, \ldots, x'_{k'}\}$ are disjunct. Furthermore, let

$$\begin{aligned} \varphi &= \bigwedge_{i=1}^{n} (l_i^1 \vee l_i^2 \vee l_i^3) \ , \text{ and} \\ \varphi' &= \bigwedge_{i=1}^{n'} (l'^1_i \vee l'^2_i \vee l'^3_i) \ . \end{aligned}$$

l_i and $l'i$ are literals; that is, x or $\neg x$ with x a variable. For each variable x_i (x'_i) we introduce two items X_i and $\overline{X_i}$ (X'_i and $\overline{X'_i}$). Also for each clause we introduce one variable: $C_1, \ldots, C_n$ for φ and $C'_1, \ldots, C'_{n'}$ for φ'. The set of frequency constraints $\mathcal{C}$ consists of the following expressions.

(1) For each variable x_i, $i = 1 \ldots k$, the constraints

$$\mathit{freq}(X_i) = \frac{1}{2} \ , \ \mathit{freq}(\overline{X_i}) = \frac{1}{2} \ , \text{ and } \mathit{freq}(X_i \overline{X_i}) = 0 \ .$$

(2) For each clause $C_i = (l_i^1 \vee l_i^2 \vee l_i^3)$, $i = 1 \ldots n$, the constraint

$$\mathit{freq}(\{C_i, \overline{L_i^1}, \overline{L_i^2}, \overline{L_i^3}\}) = 0 \ .$$

$\overline{L_i^j}$ denotes X_p if $l_i^j = \neg x_p$, and is $\overline{X_p}$ if $l_i^j = x_p$.

(3) The constraint $\mathit{freq}(\{C_1, \ldots, C_n\}) = \frac{1}{2}$.

(4) For each variable x'_i, $i = 1 \ldots k'$, the constraints

$$\mathit{freq}(X'_i) = \frac{1}{2} \ , \ \mathit{freq}(\overline{X'_i}) = \frac{1}{2} \ , \text{ and } \mathit{freq}(X'_i \overline{X'_i}) = 0 \ .$$

(5) For each clause $C'_i = (l'^1_i \vee l'^2_i \vee l'^3_i)$, $i = 1 \ldots n'$, the constraint

$$\mathit{freq}(\{C'_i, \overline{L'^1_i}, \overline{L'^2_i}, \overline{L'^3_i}\}) = 0 \ .$$

$\overline{L'^j_i}$ denotes X'_p if $l'^j_i = \neg x'_p$, and is $\overline{X'_p}$ if $l'^j_i = x'_p$.

The total number of constraints is $3k + n + 1 + 3k' + n'$. Now, it holds that φ is satisfiable and φ' is unsatisfiable if and only if

$$\mathcal{C} \models_{\mathit{tight}} \mathit{freq}(\{C'_1, \ldots, C'_{n'}\}) \in [0, 0] \ .$$

This can be seen as follows. We discuss the sets of constraints {(1), (2), (3)}, and {(4), (5)} in isolation. We show that the constraints {(1), (2), (3)} are satisfiable if and only if φ is. Suppose that φ is satisfiable. Let $V : \{x_1, \ldots, x_k\} \to \{0, 1\}$ be a valuation function that makes φ true. Let T be the transaction with as set of items

$$\{X_i, i = 1 \ldots k \mid V(x_i) = 1\} \cup \{\overline{X_i}, i = 1 \ldots k \mid V(x_i) = 0\} \cup \{C_1, \ldots, C_n\} \ .$$

Let $\overline{T}$ be the transaction with items

$$\{X_i, i = 1 \ldots k \mid V(x_i) = 0\} \cup \{\overline{X_i}, i = 1 \ldots k \mid V(x_i) = 1\} \ .$$

The database $\mathcal{D} = \{T, \overline{T}\}$ satisfies the constraints (1), (2), and (3). Conditions (1) and (3) are fulfilled, as can be seen easily. For condition (2), consider one of the clauses $C_i = (l_i^1 \vee l_i^2 \vee l_i^3)$ of φ. Because V is a satisfying assignment for φ, for at least one of the literals in C_i, V must assign 1. Without loss of generality we assume that $V(l_i^1) = 1$. Thus, T does not contain $\overline{L_i^1}$, and therefore, T does not contain the itemset $\{C_i, \overline{L_i^1}, \overline{L_i^2}, \overline{L_i^3}\}$. Also the other transaction $\overline{T}$ does not contain this itemset, since C_i is not in $\overline{T}$. Therefore, also (2) is satisfied.

For the other direction, suppose there exists a transaction database $\mathcal{D}$ that satisfies $\mathcal{C}$. Since the empty database is not a solution, $\mathcal{D}$ is not empty. Because $\mathcal{D} \models \mathit{freq}(\{C_1, \ldots, C_n\}) = \frac{1}{2}$, there must be at least one transaction T in $\mathcal{D}$ that contains $\{C_1, \ldots, C_n\}$. Because of (1), for each variable x_i of φ, T must contain exactly one of X_i and $\overline{X_i}$. Let now V be the valuation that assigns 1 to x_i if and only if $X_i \in T$. We show that this assignment satisfies φ. Let $C_i = (l_i^1 \vee l_i^2 \vee l_i^3)$ be one of the clauses of φ. Because of (2), at least one of $\overline{L_i^1}, \overline{L_i^2}, \overline{L_i^3}$ is not in T. We assume without loss of generality that $\overline{L_i^1}$ is not in T. Thus, L_i^1 is in T, and hence $V(l_i^1) = 1$. Therefore, also $V(C_i) = 1$. Because C_i was chosen arbitrarily, also $V(\varphi) = 1$, and thus is φ satisfiable.

For φ' we can establish a similar reasoning. There exists a database that satisfies (4) and (5), and at the same time has $\mathit{freq}(\{C'_1, \ldots, C'_{n'}\})$ greater than 0, if and only if φ' is satisfiable. If φ is not satisfiable, then there exists a database that satisfies (4), and (5), but $\mathit{freq}(\{C'_1, \ldots, C'_{n'}\})$ must be 0 (This statement can be proved with a similar reasoning as in the last paragraph; every transaction that does contain $\{C'_1, \ldots, C'_{n'}\}$ implies a satisfying assignment for φ').

We can now easily combine a database $\mathcal{D}_1$ satisfying (1), (2), (3) and a database $\mathcal{D}_2$ satisfying (4), (5). Because the items used in (1), (2), and (3) are completely disjunct of the ones used in (4), and (5), we can take the "Cartesian product" of $\mathcal{D}_1$ and $\mathcal{D}_2$. That is, for each pair of transactions

(T_1, T_2) with $T_1 \in \mathcal{D}_1$, and $T_2 \in \mathcal{D}_2$, there is one transaction T in $\mathcal{D}_1 \times \mathcal{D}_2$ with as set of items the union of the sets of items of T_1 and T_2. $\mathcal{D}_1 \times \mathcal{D}_2$ does satisfy $\mathcal{C}$ whenever $\mathcal{D}_1$ satisfies (1), (2), and (3), and $\mathcal{D}_2$ satisfies (4), (5).

We are now ready to proof that φ is satisfiable and φ' is unsatisfiable if and only if

$$\mathcal{C} \models_{\mathit{tight}} \mathit{freq}(\{C'_1, \ldots, C'_{n'}\}) \in [0, 0] \ .$$

We consider four cases.

φ satisfiable, φ' satisfiable Since φ is satisfiable, there exists a database $\mathcal{D}_1$ that satisfies (1), (2), and (3). Because φ' is satisfiable, there exists a database $\mathcal{D}_2$ that satisfies (4), (5), and has $\mathit{freq}(\{C'_1, \ldots, C'_{n'}\})$ greater than 0. Thus, $\mathcal{D}_1 \times \mathcal{D}_2$ satisfies $\mathcal{C}$, and has $\mathit{freq}(\{C'_1, \ldots, C'_{n'}\})$ greater than 0. Therefore, the tight interval entailed for $\{C'_1, \ldots, C'_{n'}\}$ is $[0, s]$, with s strictly greater than 0.

φ satisfiable, φ' unsatisfiable The same remark as in the previous case applies; since φ is satisfiable, there exists a database $\mathcal{D}_1$ that satisfies (1), (2), and (3). Because φ' is not satisfiable, every database $\mathcal{D}_2$ that satisfies (4) and (5) has $\mathit{freq}(\{C'_1, \ldots, C'_{n'}\}) = 0$. Such a database $\mathcal{D}_2$ exists, for example

$$\mathcal{D}_2 \ = \ \{(1, \{X'_1, \ldots, X'_{n'}\}), (2, \{\overline{X'_1}, \ldots, \overline{X'_{k'}}\})\} \ .$$

Therefore, the tight interval entailed for $\{C'_1, \ldots, C'_{n'}\}$ is $[0, 0]$.

φ unsatisfiable, φ' satisfiable In this case there does not exist a database $\mathcal{D}$ that satisfies $\mathcal{C}$. Therefore, the tight interval entailed for the itemset $\{C'_1, \ldots, C'_{n'}\}$ is the empty interval $[1, 0]$.

φ unsatisfiable, φ' unsatisfiable The same as the previous case; the tight interval entailed for $\{C'_1, \ldots, C'_{n'}\}$ is the empty interval $[1, 0]$.

Thus, from this case study we derive that the only case in which $[0, 0]$ is the tight interval entailed for itemset $\{C'_1, \ldots, C'_{n'}\}$ is indeed when φ is satisfiable, and φ' is unsatisfiable. □

2.5 Integer Bounds Versus Rational Bounds

In the FREQSAT problem we use the frequency of the itemsets in order to model information, instead of the support. In this section we discuss the differences between support and frequency. Even though it seems that support

constraints and frequency constraints are very similar, we show that there is no straightforward reduction between them. The problem with support constraints differs a lot from and seems more complex than the equivalent problem with frequencies.

First we define support constraints and implication of support constraints in a similar way as we defined frequency constraints and implication.

Definition 4

- *A* support constraint over $\mathcal{I}$ *is an expression* $support(I) \in [L, U]$*, with* I *an itemset over* $\mathcal{I}$*, and* L, U *positive integers.*

- *A transaction database* $\mathcal{D}$ *over* $\mathcal{I}$ satisfies $support(I) \in [L, U]$*, denoted* $\mathcal{D} \models support(I) \in [L, U]$*, if*

$$L \leq support(I, \mathcal{D}) \leq U \ .$$

- *A transaction database* $\mathcal{D}$ *satisfies a set of support constraints* $\mathcal{S}$*, denoted* $\mathcal{D} \models \mathcal{S}$*, if* $\mathcal{D}$ *satisfies every constraint in* $\mathcal{S}$*.*

- *A set of support constraints* $\mathcal{S}$ implies *a support constraint* $support(I) \in [L, U]$*, denoted* $\mathcal{S} \models support(I) \in [L, U]$*, if every database that satisfies* $\mathcal{S}$*, also satisfies* $support(I) \in [L, U]$*.*

- *A set of support constraints* $\mathcal{S}$ tightly implies *a frequency constraint* $support(I) \in [L, U]$*, denoted* $\mathcal{S} \models_{tight} support(I) \in [L, U]$*, if* $\mathcal{S} \models support(I) \in [L, U]$*, and if for every* L', U' *such that* $\mathcal{S} \models support(I) \in [L', U']$*, it is true that* $[L, U] \subseteq [L', U']$*. Hence,* $[L, U]$ *is the smallest interval we can derive for* I *based on* $\mathcal{S}$*.* □

Problem 5 Support Satisfiability (SUPPSAT($\mathcal{S}$)) *The* SUPPSAT*-problem is, given a finite set*

$$\mathcal{S} \ = \ \{support(I_j) \in [L_j, U_j] \mid j = 1 \ldots n\}$$

of support constraints, decide whether there exists a transaction database $\mathcal{D}$ *over* $\mathcal{I} = \bigcup_{i=1}^{n} I_i$ *such that* $\mathcal{D}$ *satisfies* $\mathcal{S}$*.* □

We can find a similar representation as a linear system of inequalities for the SUPPSAT problem as we had for the FREQSAT problem. Only, this problem is an *integer* linear programming problem. It is well-known

that integer programming is more complex than linear programming (**NP**-complete versus **P**-complete) [69].

We illustrate the difference between supports and frequencies with an example. Consider the following set of frequency expressions:

$$\mathcal{C} \;=\; \left\{ \mathit{freq}(ab) = \frac{1}{2}, \mathit{freq}(ac) = \frac{1}{2}, \mathit{freq}(bc) = \frac{1}{2} \right\}$$

A straightforward "reduction" to a SUPPSAT problem would be the following:

$$\mathcal{S} \;=\; \left\{ \begin{array}{ll} \mathit{support}(\{\}) = 2, & \mathit{support}(ab) = 1, \\ \mathit{support}(ac) = 1, & \mathit{support}(bc) = 1 \end{array} \right\}$$

However,

$$\mathcal{C} \models_{\mathit{tight}} \mathit{freq}(abc) \in \left[\frac{1}{4}, \frac{1}{2} \right] \quad ,$$

while

$$\mathcal{S} \models_{\mathit{tight}} \mathit{support}(abc) = 1 \quad .$$

The following database proves that $\mathit{freq}(abc) = \frac{1}{4}$ is indeed possible:

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

The problem in a reduction from FREQSAT to SUPPSAT is that we need to set an upper bound on the number of transactions. As the following example shows, the least common multiplier of the denominators of the FREQSAT-problem is not a good choice. Let the set of items $\mathcal{I}$ be $\{a, b, c, d, e, f, g, h, i, j, k, l\}$.

$$\mathcal{C}_2 \;=\; \left\{ \begin{array}{ll} \multicolumn{2}{l}{\mathit{freq}(ab) = \mathit{freq}(ac) = \mathit{freq}(bc) = \frac{1}{2}} \\ \multicolumn{2}{l}{\mathit{freq}(de) = \mathit{freq}(df) = \mathit{freq}(ef) = \frac{1}{2}} \\ \multicolumn{2}{l}{\mathit{freq}(gh) = \mathit{freq}(gi) = \mathit{freq}(hi) = \frac{1}{2}} \\ \multicolumn{2}{l}{\mathit{freq}(jk) = \mathit{freq}(jl) = \mathit{freq}(kl) = \frac{1}{2}} \\ & \\ \mathit{freq}(abc\ def) = 0 & \mathit{freq}(abc\ ghi) = 0 \\ \mathit{freq}(abc\ jkl) = 0 & \mathit{freq}(def\ ghi) = 0 \\ \mathit{freq}(def\ jkl) = 0 & \mathit{freq}(ghi\ jkl) = 0 \end{array} \right\}$$

Notice that, as argued above,

$$\left\{ freq(ab) = \frac{1}{2}, freq(ac) = \frac{1}{2}, freq(bc) = \frac{1}{2} \right\} \models_{tight} freq(abc) \in \left[\frac{1}{4}, \frac{1}{2}\right]$$

Similarly, we get $\mathcal{C}_2 \models_{tight} freq(def) \in \left[\frac{1}{4}, \frac{1}{2}\right]$, $\mathcal{C}_2 \models_{tight} freq(ghi) \in \left[\frac{1}{4}, \frac{1}{2}\right]$, and $\mathcal{C}_2 \models_{tight} freq(jkl) \in \left[\frac{1}{4}, \frac{1}{2}\right]$. Furthermore, because no two of the four sets abc, def, ghi, and jkl can appear in the same transaction, $\mathcal{C}$ can never be satisfied by a database with 2 transactions. It is however satisfiable with 4 transactions, as the following example shows:

TID	Items
1	$a, b, c, d, f, g, i, j, k$
2	$a, b, d, e, f, g, h, k, l$
3	$a, c, d, e, g, h, i, j, l$
4	$b, c, e, f, h, i, j, k, l$

Theorem 5 SUPPSAT *is in* **PSPACE**.

Proof
Let $\mathcal{S} = \{support(I_j) \in [L_j, U_j] \mid j = 1 \ldots n\}$ be a SUPPSAT problem. Suppose that there exists a database $\mathcal{D}$ that satisfies $\mathcal{S}$. Remove from $\mathcal{D}$ all transactions that do not contain any of the sets I_j, $j = 1 \ldots n$. The resulting database still satisfies $\mathcal{S}$, since the transactions that were removed had no influence on the support of the itemsets in $\mathcal{S}$. Furthermore, the resulting database has at most $\sum_{j=1\ldots n} U_j$ transactions. Therefore, $\mathcal{S}$ is satisfiable if and only if there exists a database $\mathcal{D}$ with at most $\sum_{j=1\ldots n} U_j$ transactions that satisfies $\mathcal{S}$.

We show a non-deterministic procedure to decide the satisfiability of $\mathcal{S}$ that uses at most polynomial space in the length of $\mathcal{S}$. In this way we show that SUPPSAT is in **NPSPACE**, and thus by Savitch's Theorem [69, pp. 149-150], also in **PSPACE**.

We will "guess" a database $\mathcal{D}$. First we guess the number of transactions d. This number is bounded above by $\sum_{j=1\ldots n} U_j$, and can thus be represented using polynomial space in the input. For each set I_j, $j = 1 \ldots n$, we keep a counter. In the beginning these counters are all initialized to 0. We then guess the transactions one by one, reusing space. After we guess a transaction, the counters are updated. The counter associated with set I_j, $j = 1 \ldots n$ is incremented by 1 if I_j is contained in the generated transaction. By maintaining one more counter, we make sure that we stop generating transactions after the d-th transaction was generated. Then we check whether the supports we counted are consistent with the support constrains in the input. If this is

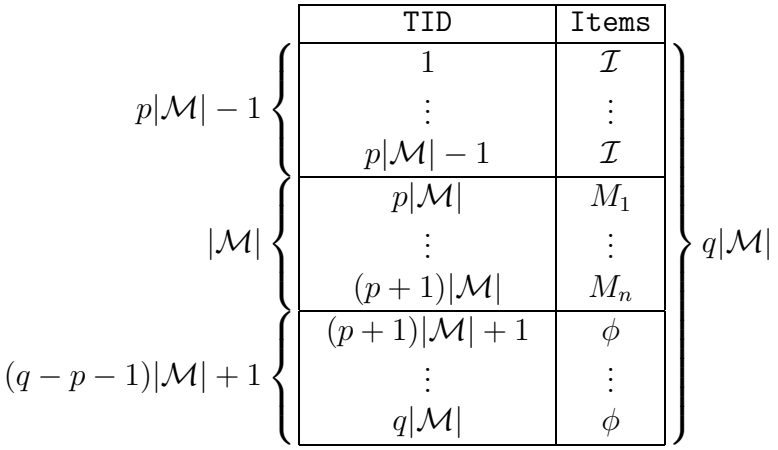


Figure 2.4: Database $\mathcal{D}$ constructed in Lemma 3.

the case, we answer "yes", otherwise "no". It is clear that if there exists a database that satisfies $\mathcal{S}$, this database will be generated in at least one of the execution paths of this non-deterministic procedure. Furthermore, the space requirement is polynomial, since the maximum number on a counter is bounded by $\sum_{j=1\ldots n} U_j$, which has a representation that is polynomial in the input. The total number of counters is the number of sets in the input, plus two extra. The generated transactions have linear length in the input. □

2.6 Special Cases of FREQSAT

The **NP**-completeness of the FREQSAT problem motivates the study of special cases that are interesting in practice and have better complexity properties.

The most straightforward example is that we only consider constraints of the form $\mathit{freq}(I) \in [t, 1]$ and $\mathit{freq}(I) \in [0, t[$, for a fixed threshold t between 0 and 1. $\mathit{freq}(I) \in [t, 1]$ represents the information that I is frequent, and $\mathit{freq}(I) \in [0, t[$ that I is infrequent. We discuss this problem here to illustrate the different properties of the special cases we will study in later chapters. We discuss the satisfiability problem, an axiomatization, and the complexity of entailment.

First, we prove a lemma that is important for this example.

Lemma 3 *Let $\mathcal{F}$ be a set of itemsets over $\mathcal{I}$, and let t be a rational number larger than or equal to 0, and strictly smaller than 1. There exists a database*

$\mathcal{D}$ over $\mathcal{I}$ such that $\text{FSET}(\mathcal{D}, t) = \mathcal{F}$ *if and only if $\mathcal{F}$ is downward closed* [4].

Proof

If. Let $\mathcal{M}$ be the maximal elements of $\mathcal{F}$ w.r.t. the subset ordering. Suppose $t = \frac{p}{q}$. We construct a database $\mathcal{D}$ with $q|\mathcal{M}|$ transactions. This construction is illustrated in Figure 2.4. $\mathcal{D}$ contains $p|\mathcal{M}| - 1$ transactions that contain all items of $\mathcal{I}$. Furthermore, $\mathcal{D}$ contains for each set $I \in \mathcal{M}$, one transaction with I as set of items. $\mathcal{D}$ also contains $(q-p)|\mathcal{M}| + 1 - |\mathcal{M}|$ transactions with $\{\}$ as set of items. Notice that the number of transactions was chosen in such a way that this number $(q-p)|\mathcal{M}|+1-|\mathcal{M}|$ is always positive. The total number of transactions is

$$p|\mathcal{M}| - 1 + |\mathcal{M}| + (q-p)|\mathcal{M}| + 1 - |\mathcal{M}| \;=\; q|\mathcal{M}| \;.$$

For every set $I \in \mathcal{M}$, the frequency of I equals

$$\mathit{freq}(I, \mathcal{D}) = \frac{(p|\mathcal{M}| - 1) + 1}{q|\mathcal{M}|} \;=\; \frac{p}{q} \;=\; t \;.$$

Because of the monotonicity principle, for all the other sets in $\mathcal{F}$ it is also true that the frequency is at least t. On the other hand, for a set not in $\mathcal{F}$, there are only $p|\mathcal{M}| - 1$ transactions that contain this set.

Only If. Because of the monotonicity principle, $\text{FSET}(\mathcal{D}, t)$ must be downward closed. □

Notice that in the lemma we require that t is not 1. The case $t = 1$ is special. Indeed, suppose that I_1 and I_2 both have frequency 1, then also $I_1 \cup I_2$ has frequency 1. Therefore, for every database $\mathcal{D}$, $\text{FSET}(\mathcal{D}, 1)$ is downward closed and closed under union. In the rest of this section we assume that t is strictly smaller than 1.

In the special cases we study, we always look for a sound and complete set of axioms. These axioms allow us to better understand the interactions between the frequencies. They will also lead to deduction rules we can use to derive bounds on target itemsets. We also always study the complexity of computing bounds on a target itemset.

For the special case presented here, we get the following two axioms.

$\boldsymbol{A_1}$ If $I \subseteq J$ then

$$\mathit{freq}(I) \in [0, t[\;\models\; \mathit{freq}(J) \in [0, t[$$

[4] A set of sets $\mathcal{F}$ is said to be *downward closed* if for every two sets $I_1 \subseteq I_2$ it holds that $I_2 \in \mathcal{F}$ implies that $I_1 \in \mathcal{F}$. That is, whenever $\mathcal{F}$ contains a set I, it must contain all subsets of I as well.

$\boldsymbol{A_2}$ If $I \subseteq J$ then

$$freq(J) \in [t, 1] \;\models\; freq(I) \in [t, 1]$$

The soundness of these two axioms follows from the monotonicity principle. The completeness follows from Lemma 3. In this very specific case the monotonicity rule is complete.

The axioms can be used as deduction rules to derive information from a set of constraints. For example, using axiom $\boldsymbol{A_1}$, we can derive $freq(ab) \in [t, 1]$ from $\{freq(abc) \in [t, 1]\}$. In every case we will show how to apply the axioms (in which order e.g.) to derive the intervals that are tightly implied, in a finite number of steps.

Although such a procedure, based on a finite number of applications of the deduction rules, is an effective method to derive bounds on the frequency, it is not always the best method. In many cases it is more efficient to solve the linear systems associated with the deduction problems. In this respect we study the computational complexity of the different cases in detail.

The following table summarizes all special cases studied in this thesis.

Case Name	Form of the set of constraints $\mathcal{C}$
Chapter 2	
1. General	$\{freq(I_1) \in [l_1, u_1], \ldots, freq(I_n) \in [l_n, u_n]\}$
2. Threshold t	$\{freq(I_1) \in [0, t[, \ldots, freq(I_k) \in [0, t[,$ $freq(I_{k+1}) \in [t, 1], \ldots, freq(I_n) \in [t, 1]\}$
Chapter 3	
3. Lower Bounds	
Systems of Frequent Sets	$\{freq(I) \in [l, 1],\ \forall I \subseteq \mathcal{I}\}$
Sparse Systems of FS	$\{freq(I_1) \in [l_1, 1], \ldots, freq(I_n) \in [l_n, 1]\}$
4. Upper Bounds	$\{freq(I_1) \in [0, u_1], \ldots, freq(I_n) \in [0, u_n]\}$
Chapter 4	
5. Exact Frequency, All subsets of I	$\{freq(J) \in [f_J, f_J],\ \forall J \subset I\}$

3

Lower and Upper Bounds in Isolation

In this chapter we consider lower and upper bounds in isolation. For the lower-bound case, we only consider constraints of the form $freq(I) \in [l, 1]$ with l a rational number. Such an expression will be called a *Frequent Set Expression*, and will be denoted by $freq(I) \geq l$. Of course, a set of such expressions is always satisfiable. In this chapter we consider the notion of *completeness* of a set of frequent set expressions. A set of frequent set expressions $\mathcal{C}$ is complete if it contains all information that is implied by it. That is, for each expression $freq(I) \geq l$ in $\mathcal{C}$, it must be that $\mathcal{C} \models_{tight} freq(I) \geq l$. We give an axiomatization for complete sets of frequent set expressions. In a first phase we only consider sets $\mathcal{C}$ in which for each subset I of $\mathcal{I}$, one expression $freq(I) \geq l$ has been given. Later on we also consider *sparse systems*, that is, sets of constraints that do not necessarily for every itemset contain an expression. We show that deciding completeness can be done in polynomial time.

For the upper-bound case we consider expressions of the form $freq(I) \in [0, u]$ with u a rational number. Such an expression will be called an *Infrequent Set Expression*. Also for this case we describe an axiomatization for complete systems. This axiomatization however, is much simpler than the one for frequent set expressions. Because of this, the complexity of deciding completeness for such systems is lower (only logarithmic space).

Finally, we show that in general a combination of the axioms for the lower bounds, together with the axioms for the upper bounds do not provide us with a complete axiomatization for the general case.

Bibliographic Note We already published large parts of this chapter in [17, 18, 19].

3.1 Lower Bounds

3.1.1 Systems of Frequent Sets

Complete Systems

We introduce a system of frequent sets as a full collection of frequent set expressions. Logical implication and completeness of systems are as defined for sets of frequency constraints. In this chapter we will often use $freq(I) \geq l$ to denote the frequency constraint $freq(I) \in [l, 1]$.

Definition 5 *Let $\mathcal{I} = \{i_1, \ldots, i_n\}$ be a set of items.*

- *A* system of frequent sets *over $\mathcal{I}$ is a collection*

$$\{freq(I) \geq p_I \mid I \subseteq \mathcal{I}\}$$

  of frequent set expressions, with one expression for each $I \subseteq \mathcal{I}$.

- *A system of frequent sets $\mathcal{S} = \{freq(I) \geq p_I \mid I \subseteq \mathcal{I}\}$ is* complete *if for each $freq(I) \geq p$ logically implied by $\mathcal{S}$, $p \leq p_I$ holds. Hence, for every $I \subseteq \mathcal{I}$, $\mathcal{S} \models_{tight} freq(I) \geq p_I$.*

□

Proof-Databases

Very important in the completeness proof of the axiomatization are the so-called *proof-databases*.

Definition 6 *Let $\mathcal{S} = \{freq(I) \geq l_I \mid I \subseteq \mathcal{I}\}$ be a system of frequent sets, and $J \subseteq \mathcal{I}$. A transaction database $\mathcal{D}$ over $\mathcal{I}$ is called a* proof-database for J in $\mathcal{S}$ *if $\mathcal{D} \models \mathcal{S}$ and $freq(J, \mathcal{D}) = l_J$.* □

In order to show that a certain system $\mathcal{S} = \{freq(I) \geq l_I \mid I \subseteq \mathcal{I}\}$ is complete, we need to construct a proof-database $\mathcal{D}_I$ for every $I \subseteq \mathcal{I}$ in $\mathcal{S}$. Suppose $\mathcal{S} \models freq(I) \geq l$. Then $freq(I, \mathcal{D}_I) \geq l$, since $\mathcal{D}_I$ satisfies $\mathcal{S}$. Hence, $l \leq l_I$. Thus, a proof-database for I in $\mathcal{S}$ shows that the frequency l_I given in the system S cannot be improved.[1]

[1] Observe the similarities with Armstrong relations in functional dependency theory [27].

Database $\mathcal{D}$

TID	Items
1	a,c,e,f
2	a,c,e,f
3	b,d,e
4	a,b,c,f
5	a,d,f
6	b,d,e,f
7	a,b,d,e,f
8	c,f
9	a,b,c,e
10	a,c,f

$freq(a) = 0.7$
$freq(b) = 0.5$
$freq(ab) = 0.3$
$freq(def) = 0.2$
$freq(ef) = 0.4$

Figure 3.1: A transaction database

Example 5 *Let $\mathcal{I} = \{a, b, c, d, e, f\}$. Consider the following system:*

$$\mathcal{S} = \{freq(I) \geq l_I \mid I \subseteq \mathcal{I}\} \quad ,$$

where $l_a = 0.7$, $l_b = 0.5$, $l_{ab} = 0.3$, $l_{def} = 0.2$, and $l_I = 0$ for all other itemsets I. The database $\mathcal{D}$ in Fig. 3.1 satisfies $\mathcal{S}$. $\mathcal{S}$ is not complete, because in every database satisfying $freq(def) \geq 0.2$, the frequency of de must be at least 0.2, and $\mathcal{S}$ contains $freq(de) \geq 0$. Furthermore, $\mathcal{S}$ does not *logically imply $freq(ef) \geq 0.5$, since $\mathcal{D}$ satisfies $\mathcal{S}$, and $\mathcal{D}$ does not satisfy $freq(ef) \geq 0.5$.*

Consider the following system over $\mathcal{I} = \{a, b, c\}$:

$$\left\{ \begin{array}{llll} freq(\phi) \geq 1, & freq(a) \geq 0.6, & freq(b) \geq 0.8, & freq(c) \geq 0.8, \\ freq(ab) \geq 0.6, & freq(ac) \geq 0.4 & freq(bc) \geq 0.6, & freq(abc) \geq 0.4 \end{array} \right\}$$

This system is complete. In Fig. 3.2, a possible set of proof-databases is given. □

Notice that when a system is complete, it is not necessary that there exists one database that is a proof-database for all itemsets at once. Consider for example the following system:

$$\left\{ \begin{array}{llll} freq(\phi) \geq 1, & freq(a) \geq 0.5, & freq(b) \geq 0.5, & freq(c) \geq 0.1, \\ freq(ab) \geq 0, & freq(ac) \geq 0, & freq(bc) \geq 0, & freq(abc) \geq 0 \end{array} \right\}$$

This system is complete. However, we will never find a database in which the following six conditions are simultaneously true: $freq(a) = 0.5$, $freq(b) = 0.5$,

Proof-DB for b, c, bc, abc

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

Proof-DB for a, ab, ac

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

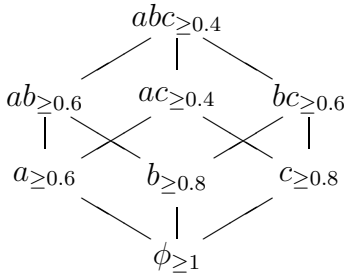


Figure 3.2: Proof-databases for a system of frequent sets

$freq(c) = 0.1$, $freq(ab) = 0$, $freq(ac) = 0$, and $freq(bc) = 0$, because due to $freq(a) = 0.5$, $freq(b) = 0.5$, and $freq(ab) = 0$, every transaction contains a or b. So, every transaction containing c also contains either a or b, and thus violates either $freq(ac) = 0$, or $freq(bc) = 0$.

Completion

When a system $\mathcal{S}$ is not complete, we can improve the system. Suppose $\mathcal{S} = \{freq(I) \geq l_I \mid I \subseteq \mathcal{I}\}$ is not complete. Then there exists a frequent set expression $freq(I) \geq l'_I$ with $l'_I > l_I$ that is logically implied by $\mathcal{S}$. We can improve $\mathcal{S}$ by replacing $freq(I) \geq l_I$ with $freq(I) \geq l'_I$. The next theorem states that for every system $\mathcal{S}$, there exists a unique complete system $\mathcal{C}(\mathcal{S})$, logically implied by $\mathcal{S}$.

Theorem 6 *Let $\mathcal{S}$ be a system of frequent sets. There exists a unique system $\mathcal{C}(\mathcal{S})$, the* completion of $\mathcal{S}$, *such that $\mathcal{S} \models \mathcal{C}(\mathcal{S})$, and $\mathcal{C}(\mathcal{S})$ is a complete system.*

Proof
Let $\mathcal{L}_I = \{l_I \mid \mathcal{S} \models freq(I) \geq l_I\}$. $\mathcal{L}_I$ always contains its own supremum: suppose a database $\mathcal{D}$ satisfies $\mathcal{S}$. Let l be $freq(I, \mathcal{D})$. $\mathcal{D}$ satisfies $\mathcal{S}$, hence for all $l_I \in \mathcal{L}_I$, $l \geq l_I$ holds, and therefore $l \geq \sup(\mathcal{L}_I)$ holds. Thus, every database satisfying $\mathcal{S}$, also satisfies $freq(I) \geq \sup(\mathcal{L}_I)$, and therefore $\mathcal{S} \models freq(I) \geq sup(\mathcal{L}_I)$. It is now straightforward that the system $\{freq(I) \geq \sup(\mathcal{L}_I) \mid I \subseteq \mathcal{I}\}$ is the unique completion of $\mathcal{S}$. □

Example 6 *Let $\mathcal{I} = \{a, b, c\}$. The unique completion of the system*

$$\mathcal{S}_1 = \left\{ \begin{array}{lll} \boldsymbol{freq(\phi) \geq 0.8}, & freq(a) \geq 0.6, & freq(b) \geq 0.8, \\ freq(c) \geq 0.8, & freq(ab) \geq 0.6, & freq(ac) \geq 0.4, \\ \boldsymbol{freq(bc) \geq 0.4}, & freq(abc) \geq 0.4 & \end{array} \right\}$$

is the system

$$\mathcal{S}_2 = \left\{ \begin{array}{lll} \boldsymbol{freq(\phi) \geq 1}, & freq(a) \geq 0.6, & freq(b) \geq 0.8, \\ freq(c) \geq 0.8, & freq(ab) \geq 0.6, & freq(ac) \geq 0.4, \\ \boldsymbol{freq(bc) \geq 0.6}, & freq(abc) \geq 0.4 & \end{array} \right\}$$

$freq(bc) \geq 0.6$ is implied by $\mathcal{S}_1$, since there is an overlap of at least 0.6 between the transactions containing b and the transactions containing c. The completeness of $\mathcal{S}_2$ has already been shown in Example 5. □

3.1.2 Systems of Rare Sets

Before we go into an axiomatization for complete systems of frequent sets, we first introduce rare sets. The introduction of rare sets simplifies the notations in subsequent proofs and will make reasoning easier.

Definition 7

- *Let $\mathcal{D}$ be a transaction database over $\mathcal{I}$. The* rareness *of an itemset $I \subseteq \mathcal{I}$ in $\mathcal{D}$, denoted $rare(I, \mathcal{D})$, is the fraction of transactions in $\mathcal{D}$ such that at least one of the items in I is absent. That is,*

$$rare(I, \mathcal{D}) \ =_{def} \ \frac{|\{(tid, J) \in \mathcal{D} \mid I - J \neq \phi\}|}{|\mathcal{D}|} \ .$$

- *A* rare set expression *over $\mathcal{I}$ is an expression $rare(I) \leq p_I$ with $I \subseteq \mathcal{I}$ and p_I a rational number with $0 \leq p_I \leq 1$.*

- *A database $\mathcal{D}$ over $\mathcal{I}$* satisfies *$rare(I) \leq p_I$, denoted $\mathcal{D} \models rare(I) \leq p_I$, if $rare(I, \mathcal{D}) \leq p_I$. Hence, itemset I has rareness at most p_I.*

- *A* system of rare sets *over $\mathcal{I}$ is a collection $\{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$ of rare set expressions, with one expression for each $I \subseteq \mathcal{I}$.*

- *A database $\mathcal{D}$ over $\mathcal{I}$* satisfies *the system $\mathcal{S} = \{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$, denoted $\mathcal{D} \models \mathcal{S}$, if $\mathcal{D}$ satisfies all $rare(I) \leq p_I$ in $\mathcal{S}$.*

- *A system of rare sets* $\mathcal{S}$ logically implies $rare(I) \leq p$, *denoted* $\mathcal{S} \models rare(I) \leq p$ *if every database that satisfies* $\mathcal{S}$ *also satisfies* $rare(I) \leq p$. *System* $\mathcal{S}_1$ *logically implies system* $\mathcal{S}_2$, *denoted* $\mathcal{S}_1 \models \mathcal{S}_2$, *if every* $rare(I) \leq p$ *in* $\mathcal{S}_2$ *is logically implied by* $\mathcal{S}_1$.

- *A system of rare sets* $\mathcal{S} = \{rare(I) \leq p_I \mid I \in \mathcal{I}\}$ *is* complete *if for each* $rare(I) \leq p$ *logically implied by* $\mathcal{S}$, $p_I \leq p$ *holds.*

□

Example 7 *In Fig. 3.1, the database* $\mathcal{D}$ *satisfies* $rare(a) \leq 0.4$, *because fewer than* 0.4 *of the transactions have 0 in a.* $\mathcal{D}$ *does not satisfy* $rare(b) \leq 0.3$. *Let* $\mathcal{I} = \{a, b\}$. *The system*

$$\{\boldsymbol{rare(\phi) \leq 0.4}, rare(a) \leq 0.3, rare(b) \leq 0.4, \boldsymbol{rare(ab) \leq 0.8}\}$$

is not complete. The unique completion of this system is

$$\{\boldsymbol{rare(\phi) \leq 0}, rare(a) \leq 0.3, rare(b) \leq 0.4, \boldsymbol{rare(ab) \leq 0.7}\} \quad .$$

□

The next proposition connects rare sets with frequent sets. The connection between the two is straightforward. Indeed: the transactions that miss one of the items in I are exact the complement of the transactions having all items of I.

Proposition 1 *For every database* $\mathcal{D}$ *over* $\mathcal{I}$ *and every subset* I *of* $\mathcal{I}$ *holds that*

- $freq(I, \mathcal{D}) + rare(I, \mathcal{D}) = 1$.

- $\mathcal{D}$ *satisfies* $rare(I) \leq p_I$ *if and only if* $\mathcal{D}$ *satisfies* $freq(I) \geq 1 - p_I$.

Proof
Straightforward. □

Notice that a proof-database $\mathcal{D}_I$ for an itemset I in a system of frequent sets $\{freq(I) \geq p_I \mid I \subseteq \mathcal{I}\}$ is also a proof-database for I in the system of rare sets $\{rare(I) \leq 1 - p_I \mid I \subseteq \mathcal{I}\}$.

In the following subsection we prove an axiomatization for complete systems of rare sets. From this axiomatization, we can easily derive an axiomatization for frequent sets, using Proposition 1.

3.1.3 Axioms for Complete Systems of Rare Sets

We first define bags.

Definition 8 *Let S be a finite set, and $s, s_1, \ldots, s_k \in S$.*

(a) A bag over S *is a total function from S into $\mathbb{N}$. Intuitively, a bag is a set in which elements can appear more then once.*

(b) $\mathcal{M} = \langle s_1, \ldots, s_k \rangle$ denotes the bag over S where for all $s \in S$, $\mathcal{M}(s)$ is the number of occurrences of s in the list $\langle s_1, \ldots, s_k \rangle$. As a shorthand, we denote c occurrences of s by $c \cdot s$.

Let $\mathcal{M}, \mathcal{N}$ be bags over S.

(c) $|\mathcal{M}| \ =_{def} \ \sum_{s \in S} \mathcal{M}(s)$ is the cardinality of $\mathcal{M}$.

(d) It is said that s appears n times in $\mathcal{M}$ *if $\mathcal{M}(s) = n$. The notation $s \in \mathcal{M}$ stands for $\mathcal{M}(s) \geq 1$.*

(e) The bag-union $\mathcal{M} \bigcup \mathcal{N}$ *is defined as follows: for all $t \in S$,*

$$(\mathcal{M} \bigcup \mathcal{N})(t) \ =_{def} \ \mathcal{M}(t) + \mathcal{N}(t) \ .$$

(f) Associate with each element $s \in S$ a real number n_s. $\sum_{s \in \mathcal{M}} n_s$ is shorthand for $\sum_{s \in S} \mathcal{M}(s) n_s$.

(g) Let $\phi(m)$ be a condition on m. $\langle m \in \mathcal{M} \mid \phi(m) \rangle$ denotes the bag $\mathcal{K}$ with for each $s \in S$, $\mathcal{K}(s) = \mathcal{M}(s)$ if $\phi(s)$ holds; else $\mathcal{K}(s) = 0$.

Let $\mathcal{K}$ be a bag over the subsets of S; that is, the elements of $\mathcal{K}$ are subsets of S.

(h) $\bigcup \mathcal{K}$ is the following bag over S: $\forall s \in S$, $(\bigcup \mathcal{K})(s)$ is the number of occurrences of sets in $\mathcal{K}$ that contain s; that is,

$$\left(\bigcup \mathcal{K} \right)(s) \ =_{def} \ |\langle K \in \mathcal{K} \mid s \in K \rangle| \ .$$

(i) The degree of s in $\mathcal{K}$, *denoted $deg(s, \mathcal{K})$ is $(\bigcup \mathcal{K})(s)$. The* minimal degree of $\mathcal{K}$, *denoted $mdeg(\mathcal{K})$, is $min_{s \in K}(deg(s, \mathcal{K}))$.*

□

Example 8 *$\mathcal{K} = \langle \{a, b\}, 2 \cdot \{b, c\}, 2 \cdot \{b, d\} \rangle$ is a bag over the subsets of $\{a, b, c, d\}$. $|\mathcal{K}| = 5$, $\bigcup \mathcal{K} = \langle a, 5 \cdot b, 2 \cdot c, 2 \cdot d \rangle$, $deg(b, \mathcal{K}) = 5$, and $mdeg(\mathcal{K}) = 1$.* □

The Axioms for Complete Systems of Rare Sets

$\boldsymbol{\mathcal{R}_1}$ $p_\phi = 0$

$\boldsymbol{\mathcal{R}_2}$ If $I_2 \subseteq I_1$, then $p_{I_2} \leq p_{I_1}$

$\boldsymbol{\mathcal{R}_3}$ Let $I \subseteq \mathcal{I}$, $\mathcal{M}$ a bag of subsets of I, with $deg(i, \mathcal{M}) \geq 1$ for all $i \in I$. Then

$$p_I \leq \frac{\sum_{M \in \mathcal{M}} p_M}{k},$$

with $k = mdeg(\mathcal{M})$.

We now show that these axioms are sound and complete for complete systems of rare sets.

Theorem 7 Axiomatization of Rare Sets *Let*

$$\mathcal{S} \;=\; \{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$$

be a system of rare sets over $\mathcal{I}$. The following two statements are equivalent:

- *$\mathcal{S}$ is a complete system.*
- *$\mathcal{S}$ satisfies $\boldsymbol{\mathcal{R}_1}$, $\boldsymbol{\mathcal{R}_2}$, and $\boldsymbol{\mathcal{R}_3}$.*

We split the proof into two parts. In the first part, the soundness of the axioms is proved. In the second part, the completeness of the axioms is showed. Proposition 2 and Proposition 3 together then proof Theorem 7.

Soundness of the Axioms for Rare Sets

Proposition 2 Soundness *Let $\mathcal{S} = \{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$ be a system of rare sets over $\mathcal{I}$. If $\mathcal{S}$ is complete, then $\boldsymbol{\mathcal{R}_1}$, $\boldsymbol{\mathcal{R}_2}$, and $\boldsymbol{\mathcal{R}_3}$ are satisfied.*

Proof

The soundness of $\boldsymbol{\mathcal{R}_1}$ and $\boldsymbol{\mathcal{R}_2}$ is straightforward.

For $\boldsymbol{\mathcal{R}_3}$, let $\mathcal{S} = \{rare(I) \leq p_I \mid I \in \mathcal{I}\}$ be a complete system, and let $\mathcal{M}$ be a bag over the subsets of $I \subseteq \mathcal{I}$, with $deg(i, \mathcal{M}) \geq 1$, for all $i \in I$. We prove that $\frac{\sum_{M \in \mathcal{M}} p_M}{k} \geq p_I$, with $k = mdeg(\mathcal{M})$.

Let $\mathcal{D}$ be a database over $\mathcal{I}$ such that $\mathcal{D} \models \mathcal{S}$. Let for all $J \subseteq \mathcal{I}$, $\mathcal{B}_J$ denote the bag over the itemsets, with

$$\mathcal{B}_J(K) \;=_{def}\; \begin{cases} |\{(tid, I) \in \mathcal{D} \mid I = K\}| & \text{if } J - K \neq \phi \\ 0 & \text{else} \end{cases}$$

Notice that with this definition,

$$rare(J, \mathcal{D}) \;=\; \frac{|\mathcal{B}_J|}{|\mathcal{D}|} \;\; .$$

Let K be a set in $\mathcal{B}_I$. Since K is in $\mathcal{B}_I$, there is at least one item $i \in I$ that is not in T. Because the minimal degree of $\mathcal{M}$ is k, there are at least k sets in $\mathcal{M}$ in which i is present. Therefore, K must be in at least k of the bags in $\langle \mathcal{B}_M \mid M \in \mathcal{M} \rangle$. Thus,

$$k\,|\mathcal{B}_I| \leq \sum_{M \in \mathcal{M}} |\mathcal{B}_M| \leq \sum_{M \in \mathcal{M}} p_M\,|\mathcal{D}| \;\; .$$

Hence,

$$rare(I, \mathcal{D}) \leq \frac{\sum_{M \in \mathcal{M}} p_M}{k} \;\; .$$

Since $\mathcal{S}$ is complete, we conclude

$$p_I \;=\; \max_{\mathcal{D}}(rare(I, \mathcal{D})) \leq \frac{\sum_{M \in \mathcal{M}} p_M}{k} \;\; .$$

□

Completeness of the Axioms for Rare Sets

First we prove a couple of lemmas. In the proof of completeness we will make the link between completeness of a system of frequent sets and the satisfiability of a specific set of inequalities, in a similar way as in the proof of Theorem 2. The next lemmas deal with this representation and with properties of systems of inequalities.

Lemma 4 *Let $\mathcal{S} = \{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$ be a system satisfying $\boldsymbol{\mathcal{R}_1}$ and $\boldsymbol{\mathcal{R}_2}$. If for all $I \subseteq \mathcal{I}$, the system*

$$\left\{ p_I - p_J \leq \sum_{i \in I} X_i - \sum_{j \in J} X_j \leq p_I, \quad \forall J \subseteq I \right. \tag{3.1}$$

has a rational solution, then $\mathcal{S}$ is complete.

Proof
Let $I \subseteq \mathcal{I}$. We show that there exists a proof-database $\mathcal{D}$ for I. Let $(\forall i \in I)X_i = \beta_i$ be a solution of (3.1). We have $(\forall i \in I)0 \leq \beta_i \leq 1$,

and $\sum_{i\in I}\beta_i = p_I \leq 1$ (from the case $J = \{\}$) Let $\mathcal{D}$ be a database satisfying: (a) a fraction β_i of the transactions has as items $\mathcal{I} - i$, for all $i \in I$; (b) a fraction $1 - \sum_{i\in I}\beta_i$ of transactions that contain all items of $\mathcal{I}$. Because all β_i's are rational, there exists such a database with as number of transactions the least general multiplier of the denominators. $\mathcal{D}$ is a proof-database for $rare(I) \leq p_I$. □

Lemma 5 *Let $I \subseteq \mathcal{I}$ an itemset. Let for every nonempty itemset $J \subseteq \mathcal{I}$, l_I, u_I be two rational numbers. Consider the following system of inequalities:*

$$l_J \leq \sum_{j\in J} X_j \leq u_J, \quad \forall J \subseteq I$$

This system has a solution $(x_1, \ldots, x_{|I|})$, with x_i rational, if and only if for all $\mathcal{K}$ and $\mathcal{L}$, bags of subsets of I with $\bigcup\mathcal{K} = \bigcup\mathcal{L}$ it is true that $\sum_{K\in\mathcal{K}} l_K \leq \sum_{L\in\mathcal{L}} u_L$.

Proof
We will use induction on $|I|$.

$|I| = 0$ Trivially fulfilled.

General case Suppose the lemma holds for $1, 2, \ldots, |I|-1$. Let $i_1 \in I$, and

$$\begin{array}{rcl} UB & = & \left\{(\sum_{L\in\mathcal{L}} u_L - \sum_{K\in\mathcal{K}} l_K)/\alpha \mid \bigcup\mathcal{K} \cup \langle\alpha\cdot i_1\rangle = \bigcup\mathcal{L}\right\} \\ LB & = & \left\{(\sum_{L\in\mathcal{L}} l_L - \sum_{K\in\mathcal{K}} u_K)/\alpha \mid \bigcup\mathcal{K} \cup \langle\alpha\cdot i_1\rangle = \bigcup\mathcal{L}\right\} \end{array} \quad (3.2)$$

We show that $\max(LB) \leq \min(UB)$. Let $\mathcal{K}, \mathcal{L}, \alpha, \mathcal{K}', \mathcal{L}', \alpha'$ be such that

$$\bigcup\mathcal{K} \cup \{\alpha\cdot i_1\} = \bigcup\mathcal{L}, \text{ and } \bigcup\mathcal{K}' \cup \{(\alpha')\cdot i_1\} = \bigcup\mathcal{L}'.$$

Then $\bigcup(\alpha'\mathcal{L} \cup \alpha\mathcal{K}') = \bigcup(\alpha\mathcal{L}' \cup \alpha'\mathcal{K})$ is true. Therefore

$$\alpha'\sum_{L\in\mathcal{L}} l_L + \alpha\sum_{K\in\mathcal{K}'} l_K \leq \alpha\sum_{L\in\mathcal{L}'} u_L + \alpha'\sum_{K\in\mathcal{K}} u_K,$$

and thus

$$\left(\sum_{L\in\mathcal{L}} l_L - \sum_{K\in\mathcal{K}} u_K\right)\Big/\alpha \leq \left(\sum_{L\in\mathcal{L}'} u_L - \sum_{K\in\mathcal{K}'} l_K\right)\Big/\alpha'.$$

Choose now β_1 rational such that $\max(LB) \leq \beta_1 \leq \min(UB)$.

Consider the following system (3.3) (X_1 has been replaced by β_1), $l'_K = \max(l_K, l_{(K\cup\{i_1\})} - \beta_1)$, and $u'_K = \min(u_K, u_{(K\cup\{i_1\})} - \beta_1)$, for all $K \subseteq I - \{i_1\}$.

$$\begin{cases} l'_K \leq \sum_{k\in K} X_k \leq u'_K, & \forall K \subseteq I - \{i_1\} \end{cases} \tag{3.3}$$

We use induction to show this system has a solution. Therefore, we need to show that whenever $\bigcup \mathcal{K} = \bigcup \mathcal{L}$,

$$\sum_{K\in\mathcal{K}} \max(l_K, l_{K\cup\{i_1\}} - \beta_1) \leq \sum_{L\in\mathcal{L}} \min(u_L, u_{L\cup\{i_1\}} - \beta_1) \tag{3.4}$$

holds. Let $\mathcal{K} = \mathcal{K}' \bigcup \mathcal{K}''$, $\mathcal{L} = \mathcal{L}' \bigcup \mathcal{L}''$, where

$$\begin{array}{l} \mathcal{K}' = \langle K \in \mathcal{K} \mid l_K < l_{K\cup\{i_1\}} - \beta_1 \rangle, \\ \mathcal{L}' = \langle L \in \mathcal{L} \mid l_L < l_{L\cup\{i_1\}} - \beta_1 \rangle. \end{array} .$$

Suppose $|\mathcal{L}'| > |\mathcal{K}'|$. Then we have

$$\overbrace{\bigcup_{L\in\mathcal{L}'} (L\cup\{i_1\}) \cup \bigcup \mathcal{L}''}^{\mathcal{N}} = \overbrace{\bigcup_{K\in\mathcal{K}'} (K\cup\{i_1\}) \cup \bigcup \mathcal{K}''}^{\mathcal{M}} \cup (|\mathcal{L}'| - |\mathcal{K}'| \{i_1\}).$$

Since $\beta_1 \geq \max(LB)$,

$$\beta_1 \geq \frac{\sum_{M\in\mathcal{M}} l_M - \sum_{N\in\mathcal{N}} u_N}{|\mathcal{L}'| - |\mathcal{K}'|}$$

holds. In case $|\mathcal{L}'| < |\mathcal{K}'|$, a similar argument can be used, but with UB instead of LB (3.2). Therefore, (3.4) holds, and by induction the second system has a solution $\beta_2, \ldots, \beta_{|I|}$. It is easy to see that $\beta_1, \ldots, \beta_{|I|}$ is a solution for the original system.

□

Lemma 6 *Let $\mathcal{S} = \{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$ be a system of rare sets, and I be an itemset over $\mathcal{I}$. If $\mathcal{S}$ satisfies $\boldsymbol{\mathcal{R}_1}$, $\boldsymbol{\mathcal{R}_2}$, and $\boldsymbol{\mathcal{R}_3}$, then the system*

$$\begin{cases} p_I - p_J \leq \sum_{i\in I} X_i - \sum_{j\in J} X_j \leq p_I, & \forall J \subseteq I \end{cases} \tag{3.5}$$

has a rational solution.

Proof
According to Lemma 5, (3.5) has a solution if and only if for all bags $\mathcal{M}$ and $\mathcal{N}$ over the subsets of I, such that $\bigcup \mathcal{M} = \bigcup \mathcal{N}$,

$$\sum_{M \in \mathcal{M}} p_I - p_{I-M} \leq \sum_{N \in \mathcal{N}} p_N$$

holds. Let $\mathcal{L} = \mathcal{N} \bigcup \langle I - M \mid M \in \mathcal{M} \rangle$.
Then, by $\boldsymbol{\mathcal{R}_3}$ we have that $\sum_{L \in \mathcal{L}} p_L \geq k p_I$, with

$$k = min_{i \in I} \left| \langle N \mid i \in N \wedge N \in \mathcal{N} \rangle \bigcup \langle M \mid M \in \mathcal{M} \wedge i \notin M \rangle \right|.$$

Because $|\langle M \mid M \in \mathcal{M} \wedge i \in M \rangle| = |\langle N \mid N \in \mathcal{N} \wedge i \in N \rangle|$, it follows that $k = |\mathcal{M}|$.
Therefore, $\sum_{L \in \mathcal{L}} p_L \geq |\mathcal{M}|\, p_K$ holds.
Since

$$\sum_{L \in \mathcal{L}} p_L = \sum_{N \in \mathcal{N}} p_N + \sum_{M \in \mathcal{M}} p_{K-M},$$

and $|\mathcal{M}|\, p_K = \sum_{M \in \mathcal{M}} p_I$,
$\sum_{M \in \mathcal{M}} p_I - p_{I-M} \leq \sum_{N \in \mathcal{N}} p_N$ holds. □

Proposition 3 *Let $\mathcal{S} = \{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$ be a system of rare sets over $\mathcal{I}$. If $\mathcal{S}$ satisfies $\boldsymbol{\mathcal{R}_1}$, $\boldsymbol{\mathcal{R}_2}$, and $\boldsymbol{\mathcal{R}_3}$, then $\mathcal{S}$ is complete.*

Proof
If S satisfies $\boldsymbol{\mathcal{R}_1}$, $\boldsymbol{\mathcal{R}_2}$, and $\boldsymbol{\mathcal{R}_3}$, then

$$\left\{ p_I - p_J \leq \sum_{i \in I} X_i - \sum_{j \in J} X_j \leq p_I, \quad \forall J \subseteq I \right.$$

has a rational solution (Lemma 6.) Therefore, $\mathcal{S}$ is complete (Lemma 4.) □

Example 9 *Consider the following systems:*

$$\mathcal{S}_1 = \left\{ \begin{array}{lll} \boldsymbol{rare(\phi) \leq 0.2}, & rare(a) \leq 0.8, & rare(b) \leq 0.4, \\ rare(c) \leq 0.4, & rare(ab) \leq 0.4, & rare(ac) \leq 0.4, \\ rare(bc) \leq 0.8, & rare(abc) \leq 1 & \end{array} \right\}$$

$$\mathcal{S}_2 = \left\{ \begin{array}{lll} rare(\phi) \leq 0, & \boldsymbol{rare(a) \leq 0.8}, & rare(b) \leq 0.4, \\ rare(c) \leq 0.4, & \boldsymbol{rare(ab) \leq 0.4}, & rare(ac) \leq 0.4, \\ rare(bc) \leq 0.8, & rare(abc) \leq 1 & \end{array} \right\}$$

$$\mathcal{S}_3 = \left\{ \begin{array}{lll} rare(\phi) \leq 0, & rare(a) \leq 0.4, & rare(b) \leq 0.4, \\ rare(c) \leq 0.4, & \boldsymbol{rare(ab) \leq 0.4}, & \boldsymbol{rare(ac) \leq 0.4}, \\ rare(bc) \leq 0.8, & \boldsymbol{rare(abc) \leq 1} & \end{array} \right\}$$

$$\mathcal{S}_4 = \left\{ \begin{array}{lll} rare(\phi) \leq 0, & rare(a) \leq 0.4, & rare(b) \leq 0.4, \\ rare(c) \leq 0.4, & rare(ab) \leq 0.4, & rare(ac) \leq 0.4, \\ rare(bc) \leq 0.8, & rare(abc) \leq 0.8 & \end{array} \right\}$$

$\mathcal{S}_1$ is not complete, since $rare(\phi) \leq 0.2$ violates $\boldsymbol{\mathcal{R}_1}$. $\mathcal{S}_2$ is not complete, since $rare(ab) \leq 0.4$ and $rare(a) \leq 0.8$ violate $\boldsymbol{\mathcal{R}_2}$. The system $\mathcal{S}_3$ is not complete, since $rare(ab) \leq 0.4$, $rare(ac) \leq 0.4$, and $rare(abc) \leq 1$ violate $\boldsymbol{\mathcal{R}_3}$. The system $\mathcal{S}_4$ is complete, since it satisfies $\boldsymbol{\mathcal{R}_1}$, $\boldsymbol{\mathcal{R}_2}$, and $\boldsymbol{\mathcal{R}_3}$. $\mathcal{S}_4$ is the unique completion of $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$. □

Why Bags are Necessary in $\mathcal{R}_3$

In the previous section we proved that $\boldsymbol{\mathcal{R}_1}$, $\boldsymbol{\mathcal{R}_2}$, and $\boldsymbol{\mathcal{R}_3}$ are sound and complete for complete systems of rare set expressions. In rule $\boldsymbol{\mathcal{R}_3}$, we state a condition that has to be tested for all bags over the subsets of all itemsets I. Later on we will show that it is not necessary to test all bags. We will describe a finite class of bags that is sufficient to test. Here we prove that in rule $\boldsymbol{\mathcal{R}_3}$, we cannot change the condition "$\mathcal{M}$ is a *bag* of subsets of I" into "$\mathcal{M}$ is a *set* of subsets of I". Therefore we will prove that $\boldsymbol{\mathcal{R}_1}$, $\boldsymbol{\mathcal{R}_2}$, and

> $\overline{\boldsymbol{\mathcal{R}_3}}$ = Let $I \subseteq \mathcal{I}$, $\mathcal{M}$ a **subset** of 2^I, with $deg(i, \mathcal{M}) \geq 1$, for all $i \in I$. Then
>
> $$p_I \leq \frac{\sum_{M \in \mathcal{M}} p_M}{k},$$
>
> with $k = mdeg(\mathcal{M})$

are not complete.

Consider the following system of rare sets:

$$\mathcal{S} = \left\{ \begin{array}{lll} rare(\phi) \leq 0, & rare(a) \leq 0.4, & rare(b) \leq 0.4, \\ rare(c) \leq 0.4, & rare(d) \leq 0.4, & rare(ab) \leq 0.4, \\ rare(ac) \leq 0.4, & rare(ad) \leq 0.4, & rare(bc) \leq 0.8, \\ rare(bd) \leq 0.8, & rare(cd) \leq 0.8, & rare(abc) \leq 0.8, \\ rare(abd) \leq 0.8, & rare(acd) \leq 0.8, & rare(bcd) \leq 0.8, \\ \boldsymbol{rare(abcd) \leq 1} & & \end{array} \right\} \quad (3.6)$$

This system is not complete as can be seen by $\boldsymbol{\mathcal{R}_3}$ with $I = abcd$ and

$$\mathcal{M} = \langle ab, ac, ad, 2 \cdot bcd \rangle.$$

Application of $\boldsymbol{\mathcal{R}_3}$ gives:

$$p_{abcd} \leq \frac{p_{ab} + p_{ac} + p_{ad} + 2p_{bcd}}{3} = \frac{14}{15}.$$

However, we show next that $\mathcal{S}$ satisfies $\overline{\boldsymbol{\mathcal{R}_3}}$.

Lemma 7 *Let for each $I \subseteq \mathcal{I}$, p_I be a rational number in $[0,1]$. Let $S_1, S_2 \subseteq 2^I$, and $S_1 \cap S_2 = \phi$. If $mdeg(S_1) + mdeg(S_2) = mdeg(S_1 \cup S_2)$, then it holds that*

$$\frac{\sum_{M \in (S_1 \cup S_2)} p_M}{mdeg(S_1 \cup S_2)} \geq \min \left(\frac{\sum_{M \in S_1} p_M}{mdeg(S_1)}, \frac{\sum_{M \in S_2} p_M}{mdeg(S_2)} \right).$$

Proof
Let $md_1 = mdeg(S_1)$, $md_2 = mdeg(S_2)$, $md_\cup = mdeg(S_1 \cup S_2)$. Without loss of generality, we can assume that

$$\frac{\sum_{M \in S_1} p_M}{md_1} \leq \frac{\sum_{M \in S_2} p_M}{md_2}.$$

$$\begin{aligned} \frac{\sum_{M \in (S_1 \cup S_2)} p_M}{md_\cup} &= \frac{\sum_{M \in S_1} p_M}{md_\cup} + \frac{\sum_{M \in S_2} p_M}{md_\cup} \\ &= \frac{\sum_{M \in S_1} p_M}{md_1} \frac{md_1}{md_\cup} + \frac{\sum_{M \in S_2} p_M}{md_2} \frac{md_2}{md_\cup} \\ &\geq \frac{\sum_{M \in S_1} p_M}{md_1} \frac{md_1}{md_\cup} + \frac{\sum_{M \in S_1} p_M}{md_1} \frac{md_2}{md_\cup} \\ &= \frac{\sum_{M \in S_1} p_M}{md_1}. \end{aligned}$$

□

Proposition 4 *The system of rare sets $\mathcal{S}$ given in (3.6) satisfies $\overline{\boldsymbol{\mathcal{R}_3}}$.*

Proof
Consider the following three proof databases.

$\mathcal{D}_1$

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

$\mathcal{D}_2$

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

$\mathcal{D}_3$

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

$\mathcal{S}$

$rare(\phi) \leq 0$	$rare(bc) \leq 0.8$
$rare(a) \leq 0.4$	$rare(bd) \leq 0.8$
$rare(b) \leq 0.4$	$rare(cd) \leq 0.8$
$rare(c) \leq 0.4$	$rare(abc) \leq 0.8$
$rare(d) \leq 0.4$	$rare(abd) \leq 0.8$
$rare(ab) \leq 0.4$	$rare(acd) \leq 0.8$
$rare(ac) \leq 0.4$	$rare(bcd) \leq 0.8$
$rare(ad) \leq 0.4$	$rare(abcd) \leq 1$

$\mathcal{D}_1$ is a proof-database for $b, c, d, ab, ac, ad, bc, abc$, and bcd in $\mathcal{S}$, $\mathcal{D}_2$ is a proof-database for a in $\mathcal{S}$, and $\mathcal{D}_3$ is a proof-database for abd, acd, bd, and cd is $\mathcal{S}$.

These proof-matrices show for all rare set expressions except for the expression $rare(abcd) \leq 1$ that the system $\mathcal{S}$ cannot be improved. Since $\boldsymbol{\mathcal{R}_1}, \boldsymbol{\mathcal{R}_2}, \overline{\boldsymbol{\mathcal{R}_3}}$ are sound, the only way in which $\mathcal{S}$ can violate $\boldsymbol{\mathcal{R}_1}, \boldsymbol{\mathcal{R}_2}, \overline{\boldsymbol{\mathcal{R}_3}}$ is in $abcd$ with rule $\overline{\boldsymbol{\mathcal{R}_3}}$. Therefore, to prove the proposition, we need to show that for every set S of subsets of $\{a, b, c, d\}$, the sum $\frac{\sum_{K \in S} p_K}{mdeg(S)}$ is at least 1, and thus p_{ABCD} cannot be improved with rule $\overline{\boldsymbol{\mathcal{R}_3}}$.

Consider the system $\mathcal{S}'$ that we get by replacing $rare(ab) \leq 0.4$ by $rare(ab) \leq 0.8$ in $\mathcal{S}$. $\mathcal{S}'$ is complete. $\mathcal{D}_1$ is a proof-database for b, c, d, ac, ad, bc, abc, and bcd in $\mathcal{S}'$, $\mathcal{D}_2$ is a proof-matrix for a in $\mathcal{S}'$, and $\mathcal{D}_3$ is a proof-matrix for abd, acd, bd, and cd in $\mathcal{S}'$. Two proof-databases $\mathcal{D}_4$, and $\mathcal{D}_5$ for respectively ab and $abcd$ in $\mathcal{S}'$ are given next.

$\mathcal{D}_4$

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

$\mathcal{D}_5$

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

$\mathcal{S}'$

$rare(\phi) \leq 0$	$rare(bc) \leq 0.8$
$rare(a) \leq 0.4$	$rare(bd) \leq 0.8$
$rare(b) \leq 0.4$	$rare(cd) \leq 0.8$
$rare(c) \leq 0.4$	$rare(abc) \leq 0.8$
$rare(d) \leq 0.4$	$rare(abd) \leq 0.8$
$rare(ab) \leq 0.8$	$rare(acd) \leq 0.8$
$rare(ac) \leq 0.4$	$rare(bcd) \leq 0.8$
$rare(ad) \leq 0.4$	$rare(abcd) \leq 1$

This completeness of system $\mathcal{S}'$ shows that for every set S over the subsets of $abcd$ that does not contain ab, the sum $\frac{\sum_{K \in S} p_K}{mdeg(S)}$ will be bigger than or equal to 1, because $\overline{\boldsymbol{\mathcal{R}_3}}$ is sound, and $\mathcal{S}'$ agrees with $\mathcal{S}$ on the frequency of every itemset except for ab, and thus, every expression $rare(abcd) \leq p_{abcd}$, derived from $\mathcal{S}$ without using ab, is also implied by $\mathcal{S}'$.

Since every permutation of b, c, d leaves $\mathcal{S}$ unchanged, the same result can be proven for ac and ad.

Consider also the system $\mathcal{S}''$ that we get by replacing $rare(bcd) \leq 0.8$ by $rare(bcd) \leq 1$ in the system S. Again we can show that the resulting system $\mathcal{S}''$ is complete, with the following proof-database $\mathcal{D}_6$ for bcd and $abcd$ in $\mathcal{S}''$.

$\mathcal{D}_6$

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

$\mathcal{S}''$

$rare(\phi) \leq 0$	$rare(bc) \leq 0.8$
$rare(a) \leq 0.4$	$rare(bd) \leq 0.8$
$rare(b) \leq 0.4$	$rare(cd) \leq 0.8$
$rare(c) \leq 0.4$	$rare(abc) \leq 0.8$
$rare(d) \leq 0.4$	$rare(abd) \leq 0.8$
$rare(ab) \leq 0.4$	$rare(acd) \leq 0.8$
$rare(ac) \leq 0.4$	$rare(bcd) \leq 1$
$rare(ad) \leq 0.4$	$rare(abcd) \leq 1$

Therefore, for every set S of subsets of $\{a, b, c, d\}$ that is not a superset of $\{ab, ac, ad, bcd\}$, the sum $\frac{\sum_{K \in S} p_K}{mdeg(S)}$ is at least 1. We will now use Lemma 7 to argue that every superset S of $\{ab, ac, ad, bcd\}$ will also give a sum of at least 1. For every possible superset S of $\{ab, ac, ad, bcd\}$ we will identify a subset S' such that S' has the same degree in a, b, c, and d. Then we can split S into S' and $S'' = S - S'$ such that $mdeg(S) = mdeg(S') + mdeg(S'')$. According to Lemma 7, the sum over S will be bigger than the minimum of the sum over S' and the sum over S''. Since in all cases neither S' nor S''

will be supersets of $\{ab, ac, ad, bcd\}$, both sums will be at least 1. The next tabular considers all cases systematically.

$S = \{ab, ac, ad, bcd\}$	$\frac{p_{ab} + p_{ac} + p_{ad} + p_{bcd}}{2} = 1$
$a \in S$	$S' = \{a, bcd\}$
$b \in S$	$S' = \{b, ac, ad, bcd\}$
$c \in S$	$S' = \{c, ab, ad, bcd\}$
$d \in S$	$S' = \{d, ab, ac, bcd\}$
$bc \in S$	$S' = \{ab, ac, ad, bc\}$
$abc \in S$	$S' = \{ad, abc, bcd\}$
$abd \in S$	$S' = \{ac, abd, bcd\}$
$acd \in S$	$S' = \{ab, acd, bcd\}$
$abcd \in S$	$S' = \{abcd\}$

□

Axiomatization of Frequent Sets

From Theorem 7, we can now easily derive the following axiomatization for frequent sets, using Proposition 1.

Theorem 8 Axiomatization of Frequent Sets *Let*

$$\mathcal{S} \;=\; \{freq(I) \geq l_I \mid I \subseteq \mathcal{I}\}$$

be a system of frequent sets over $\mathcal{I}$. $\mathcal{S}$ is a complete system if and only if $\mathcal{S}$ satisfies

$\mathcal{F}_1$ $l_\phi = 1$,

$\mathcal{F}_2$ *If $I_2 \subseteq I_1$, then $l_{I_2} \geq l_{I_1}$, and*

$\mathcal{F}_3$ *Let $I \subseteq \mathcal{I}$, $\mathcal{M}$ a bag of subsets of I, with $deg(i, \mathcal{M}) \geq 1$, for all $i \in I$. Then*

$$l_K \geq 1 - \frac{|\mathcal{M}| - \sum_{M \in \mathcal{M}} l_M}{k},$$

with $k = mdeg(\mathcal{M})$.

3.1.4 Computing Completions of Systems

In the rest of the section we continue working with rare sets. The results obtained for rare sets can, just like the axiomatization, easily be carried over to frequent sets.

In the previous section we introduced and proved an axiomatization for complete systems of rare and frequent sets. There is however still one problem with this axiomatization. $\boldsymbol{\mathcal{R}_3}$ states a property that has to be checked for all bags over the subsets of $\mathcal{I}$. This number of bags is infinite. In this section we show that it suffices to check only a finite number of bags: the minimal multi-covers. We show that the number of minimal multi-covers over a set is finite, and that they can be computed.

We also look at the following problem: when an incomplete system is given, can we compute its completion using the axioms? We show that this computation is indeed possible. We use $\boldsymbol{\mathcal{R}_1}$, $\boldsymbol{\mathcal{R}_2}$, and $\boldsymbol{\mathcal{R}_3}$ as inference rules to adjust rareness values in the system; whenever we detect an inconsistency with one of the rules, we improve the system. When the rules are applied in a systematic way, this method leads to a complete system within a finite number of steps.

Actually, the completion of a system of frequent sets can be computed in an obvious way by using linear programming [40]. For all itemsets I, we can minimize p_I with respect to a system of inequalities expressing that the frequencies obey the system of rare sets. Since the system of inequalities has polynomial size in the number of frequent itemsets, this algorithm is polynomial in the size of the system. However, as argued in [30], an axiomatization has as advantage that it provides human-readable proofs, and that, when the inference is stopped before termination, still a partial inference of the frequencies is provided.

Minimal Multi-covers

In the axiomatization for complete systems of rare sets, $\boldsymbol{\mathcal{R}_3}$ expresses a condition that has to be checked for every bag over the subsets of every itemset. Since the number of bags is infinite, rule $\boldsymbol{\mathcal{R}_3}$ cannot be used in a practical implementation. Therefore, we will show that it is not necessary to check *every bag*, but it suffices to check all *minimal multi-covers*, which are finite in number.

Definition 9

- *A* k-cover *of a set* S *is a bag* $\mathcal{K}$ *over the subsets of* S *such that for all* $s \in S$, $deg(s, \mathcal{K}) = k$.
- *A bag* $\mathcal{K}$ *over the subsets of a set* S *is a* multi-cover *of* S *if there exists an integer* k *such that* $\mathcal{K}$ *is a* k*-cover of* S.
- *A* k*-cover* $\mathcal{K}$ *of* S *is* minimal *if it cannot be decomposed as* $\mathcal{K} = \mathcal{K}_1 \bigcup \mathcal{K}_2$, *with* $\mathcal{K}_1$ *and* $\mathcal{K}_2$ *respectively* k_1*- and* k_2*-covers of* S, $k_1, k_2 > 0$.

□

Example 10 *Let* $I = \{a, b, c, d\}$. $\langle ab, bc, cd, ad, abcd\rangle$ *is a* 3*-cover of* I. *It is not minimal, because it can be decomposed into the following two minimal multi-covers of* I: $\langle ab, bc, cd, ad\rangle$ *(a* 2*-cover) and* $\langle abcd\rangle$ *(a* 1*-cover).* □

The Rule $\mathcal{R}_3'$

The new rule that replaces $\mathcal{R}_3$ states that it is not necessary to check all bags; we only need to check the minimal multi-covers. This adaptation gives the following $\mathcal{R}_3'$:

$\mathcal{R}_3'$ Let $I \subseteq \mathcal{I}$, $\mathcal{M}$ a minimal k-cover of I. Then

$$p_I \leq \frac{\sum_{M \in \mathcal{M}} p_M}{k}.$$

We show that the rules $\mathcal{R}_1$, $\mathcal{R}_2$, and $\mathcal{R}_3'$ are equivalent to the rules $\mathcal{R}_1$, $\mathcal{R}_2$, and $\mathcal{R}_3$. First we have to proof some lemmas.

Lemma 8 *Let* $a_1, \ldots a_n, b_1, \ldots, b_n$ *be strict positive reals. Then, for at least one* i,

$$\frac{a_1 + \ldots + a_n}{b_1 + \ldots + b_n} \geq \frac{a_i}{b_i}$$

holds.

Proof
Suppose, for the sake of contradiction, that for all i,

$$\frac{a_1 + \ldots + a_n}{b_1 + \ldots + b_n} < \frac{a_i}{b_i} \quad .$$

These inequalities imply that for each i,

$$a_1 b_i + \ldots a_n b_i < a_i b_1 + \ldots + a_i b_n \ .$$

If we sum all these inequalities, we get:

$$\sum_{1 \leq i,j \leq n} a_i b_j < \sum_{1 \leq i,j \leq n} a_i b_j \ .$$

This inequality is clearly a contradiction. □

Lemma 9 *Every k-cover $\mathcal{M}$ has a decomposition into minimal multi-covers $\mathcal{M}_1, \ldots, \mathcal{M}_n$, such that $\bigcup_{i=1\ldots n} \mathcal{M}_i = \mathcal{M}$.*

Proof
The proof is an easy induction on the cardinality of the k-cover $\mathcal{M}$. The base case $|\mathcal{M}| = 1$ is trivially fulfilled. In general, either $\mathcal{M}$ is already minimal, and in that case the lemma holds, or $\mathcal{M}$ is not minimal, in which case we can, by definition, split $\mathcal{M}$ into two covers $\mathcal{M}_1$, $\mathcal{M}_2$ such that $\mathcal{M} = \mathcal{M}_1 \bigcup \mathcal{M}_2$. We can use the induction hypothesis to split $\mathcal{M}_1$ and $\mathcal{M}_2$ into minimal covers. Those two collections of minimal covers together forms a decomposition as required. □

Notice incidentally that this decomposition is not necessarily unique; the bag $\langle ab, ac, bc, a, b, c\rangle$ can be split either as $\langle ab, ac, bc\rangle \bigcup \langle a, b, c\rangle$ or as $\langle a, bc\rangle \bigcup \langle b, ac\rangle \bigcup \langle c, ab\rangle$. For the proof this non-uniqueness is however not an issue.

Theorem 9 *Let $\mathcal{S}$ be a system of rare sets over $\mathcal{I}$. The following statements are equivalent:*

1. *$\mathcal{S}$ satisfies $\boldsymbol{\mathcal{R}_1}, \boldsymbol{\mathcal{R}_2}$, and $\boldsymbol{\mathcal{R}_3}$.*
2. *$\mathcal{S}$ satisfies $\boldsymbol{\mathcal{R}_1}, \boldsymbol{\mathcal{R}_2}$, and $\boldsymbol{\mathcal{R}_3}'$.*

Proof
$1 \Rightarrow 2$ is trivial, since $\boldsymbol{\mathcal{R}_3}'$ is less restrictive than $\boldsymbol{\mathcal{R}_3}$.
$2 \Rightarrow 1$ Suppose that the system $\mathcal{S} = \{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$ satisfies $\boldsymbol{\mathcal{R}_1}$ and $\boldsymbol{\mathcal{R}_2}$, but does not satisfy $\boldsymbol{\mathcal{R}_3}$. We will show that it is impossible that it satisfies $\boldsymbol{\mathcal{R}_3}$'.

There must be a set $I \subseteq \mathcal{I}$, and a bag $\mathcal{M}$ over the subsets of I, such that $p_I > \frac{\sum_{M \in \mathcal{M}} p_M}{k}$ with $k = \min_{i \in I}(deg(i, \mathcal{M}))$. For each $i \in I$ such that

$deg(i, \mathcal{M}) > k$, we replace $deg(i, \mathcal{M}) - k$ of the sets $J \in \mathcal{M}$ that contain i by $J - \{i\}$. In this way, we construct a k-cover $\mathcal{M}'$ of I.
Because $\mathcal{S}$ satisfies $\boldsymbol{\mathcal{R}_2}$,

$$\sum_{M \in \mathcal{M}} p_M \geq \sum_{M \in \mathcal{M}'} p_M \ .$$

The k-cover $\mathcal{M}'$ can be decomposed into minimal multi-covers $\mathcal{M}_1, \ldots, \mathcal{M}_n$ of I, with M_i a k_i-cover of I (Lemma 9). Because

$$\frac{\sum_{m \in \mathcal{M}'} p_M}{k} = \frac{\sum_{M \in \mathcal{M}_1} p_M + \ldots + \sum_{M \in \mathcal{M}_n} p_M}{k_1 + \ldots + k_n} \ ,$$

for at least one i,

$$\frac{\sum_{M \in \mathcal{M}_i}}{k_i} < p_I$$

must hold (Lemma 8.) Therefore, $\boldsymbol{\mathcal{R}_3}'$ is violated. □

The Number of Minimal Multi-Covers of a Finite Set is Finite

$\boldsymbol{\mathcal{R}_3}'$ is clearly a restriction of rule $\boldsymbol{\mathcal{R}_3}$, but we still have to show that this rule is finite. Before we give the proof, we give a very technical lemma we will need.

Lemma 10 *Let C be a positive integer, and let $\mathcal{N}$ be a bag over $\{-C, -C+1, \ldots, -1, 0, 1, \ldots, C-1, C\}$ with $\sum_{n \in \mathcal{N}} n = 0$. If $|\mathcal{N}| \geq 2C^3$, then there exists a bag $\mathcal{M} \subset \mathcal{N}$ ($\phi \neq \mathcal{M} \neq \mathcal{N}$), with $\sum_{m \in \mathcal{M}} m = 0$.*

Proof
If $0 \in \mathcal{N}$, the lemma clearly holds. Assume $0 \notin \mathcal{N}$. $\mathcal{N}^+ = \langle n \in \mathcal{N} \mid n > 0 \rangle$, $\mathcal{N}^- = \langle n \in \mathcal{N} \mid n < 0 \rangle$. At least one of $\mathcal{N}^-$ and $\mathcal{N}^+$ contains at least C^3 elements. Assume that $|\mathcal{N}^+| \geq C^3$. Therefore, there is at least one positive integer p that occurs C times. Because the sum of the elements in $\mathcal{N}^+$ is at least C^3, the sum of the elements in $\mathcal{N}^-$ is at most $-C^3$. Therefore, there are at least C^2 elements in $\mathcal{N}^-$, and thus there is a negative element n such that $deg(n, \mathcal{N}^-) \geq C$. (The same result obtains if $|\mathcal{N}^-| \geq C^3$) It is also clear that $|n|p = -pn$, and thus the bag $\langle |n| \cdot p, p \cdot n \rangle$ has sum 0, and is a non-empty subbag of $\mathcal{N}$. □

Theorem 10 *Let I be a finite set. The minimal multi-covers of I are finite in number and computable.*

Proof
We will prove this theorem by induction on the size of I.
Base case $|I| = 1$. Trivial, since $\langle I \rangle$ is the only non-empty minimal multi-cover of I.
General case. We assume by induction that the theorem holds for sets J with size up to $|I| - 1$. Thus, the degree and the cardinality of a minimal multi-cover of a set J of cardinality smaller than $|I|$ is bounded, since there is only a finite number of them. Let d, c be the respective bounds on the degree and the cardinality of the minimal multi-covers of sets of size at most $|I| - 1$.

Let $\mathcal{K}$ be a minimal k-cover of I, $i \in I$. It is clear that

$$\mathcal{L} = proj(\mathcal{K}, I - \{i\}) \ =_{def} \ \langle K - \{i\} \mid K \in \mathcal{K} \rangle$$

is a (not necessarily minimal) multi-cover of $I - \{i\}$. According to Lemma 9, we can split $\mathcal{L} = \mathcal{L}_1 \cup \ldots \cup \mathcal{L}_n$ with $\mathcal{L}_j$ a minimal l_j-cover of $I - \{i\}$. Therefore, we can split $\mathcal{K} = \mathcal{K}_1 \cup \ldots \cup \mathcal{K}_n$ with $\mathcal{L}_j = proj(\mathcal{K}_j, K - \{a\})$. By induction, $l_j \leq d$ and $|\mathcal{L}_j| \leq c$. Consider now the bag

$$\mathcal{M} = \langle l_1 - deg(i, \mathcal{K}_1), \ldots, l_n - deg(i, \mathcal{K}_n) \rangle \ .$$

The sum of the bag is 0, since

$$\sum_{j=1}^{n} l_j = k = \sum_{j=1}^{n} deg(i, \mathcal{K}_j) \ .$$

Notice also that

$$-c \leq l_j - deg(a, \mathcal{K}_j) \leq d \leq c \ .$$

Because $\mathcal{K}$ is minimal, for every sub-bag not equal to $\mathcal{M}$, the sum is not 0, otherwise the union of the $\mathcal{K}_j$'s that correspond to this subbag, would be a multi-cover of I, and thus $\mathcal{K}$ would not be minimal. Therefore, via Lemma 10, the cardinality of $\mathcal{M}$ is bounded by $2c^3$. Thus, $|\mathcal{K}| \leq 2c^4$. Hence, there are at most $2^{2|I|c^4}$ minimal multi-covers of I and thus the number of minimal multi-covers is finite. □

Computing the Completion of a System

We prove that by applying $\boldsymbol{\mathcal{R}_1}$, $\boldsymbol{\mathcal{R}_2}$, and $\boldsymbol{\mathcal{R}_3}$ as rules, we can compute the completion of any given system.

Applying for example rule $\boldsymbol{\mathcal{R}_2}$ means that whenever we see a situation $I_1 \subseteq I_2$, and the system states $rare(I_1) \leq p_{I_1}$ and $rare(I_2) \leq p_{I_2}$, and $p_{I_2} < p_{I_1}$, we improve the system by replacing $rare(I_1) \leq p_{I_1}$ by $rare(I_1) \leq p_{I_2}$. $\boldsymbol{\mathcal{R}_1}$ can only be applied once; $\boldsymbol{\mathcal{R}_2}$ and $\boldsymbol{\mathcal{R}_3}$ never create situations in which $\boldsymbol{\mathcal{R}_1}$ can be applied again.

$\boldsymbol{\mathcal{R}_2}$ is a *top-down operation*, in the sense that the rareness values of smaller sets is adjusted using values of larger sets. So, for a given system $\mathcal{S}$ we can easily reach a fixpoint for rule $\boldsymbol{\mathcal{R}_2}$, by going top-down; we first try to improve the frequencies of the largest itemsets, before continuing with the smaller ones.

$\boldsymbol{\mathcal{R}_3}$ is a *bottom-up operation*; values of smaller sets are used to adjust the values of larger sets. So, again, for a given system $\mathcal{S}$, we can reach a fixpoint for rule $\boldsymbol{\mathcal{R}_3}$, by applying it bottom-up.

A trivial algorithm to compute the completion of a system is the following: apply $\boldsymbol{\mathcal{R}_1}$, and then keep applying $\boldsymbol{\mathcal{R}_2}$ and $\boldsymbol{\mathcal{R}_3}$ randomly until a fixpoint is reached. The *limit* of this approach yields a complete system, but it is not true that always a fixpoint will be reached within a finite number of steps. In Fig. 3.3 an infinite run is illustrated. The completion of the system is all rareness values equal to 0, because for every database satisfying the system, every transaction contains ab, and every transaction contains bc, so there are no items missing in any transaction at all. When we keep applying the rules as in Fig. 3.3, we never reach this fixpoint, since in step $2n$, the value for abc is $\left(\frac{1}{2}\right)^n$. We will now show that when we apply the rules $\boldsymbol{\mathcal{R}_2}$ and $\boldsymbol{\mathcal{R}_3}$ in a systematic way, we always reach a fixpoint within a finite number of steps. This systematic approach is illustrated in Fig. 3.4. First, we apply $\boldsymbol{\mathcal{R}_2}$ top-down until we reach a fixpoint for $\boldsymbol{\mathcal{R}_2}$, and next, we apply $\boldsymbol{\mathcal{R}_3}$ bottom-up until we reach a fixpoint for $\boldsymbol{\mathcal{R}_3}$. The systematic approach is written down in Fig. 3.5. These two meta steps are all there is needed to reach the completion.

Definition 10 *Let $J \subseteq \mathcal{I}$ be an itemset, and $\mathcal{S} = \{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$ be a system of rare sets over $\mathcal{I}$. Let $J \subseteq \mathcal{I}$ be an itemset. The* projection of $\mathcal{S}$ on J, *denoted* $\mathrm{Proj}(\mathcal{S}, J)$, *is the system $\mathcal{S}' = \{rare(I) \leq p_I \mid I \subseteq J\}$.* □

Lemma 11 *Let $J \subseteq \mathcal{I}$ be an itemset, and let $\mathcal{S} = \{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$ be a system of rare sets over $\mathcal{I}$.*

(1) If $\mathcal{S}$ is complete, then $\mathrm{Proj}(\mathcal{S}, J)$ *is complete as well.*

(2) if $\mathcal{S}$ satisfies $\boldsymbol{\mathcal{R}_2}$, then $\mathrm{Proj}(\mathcal{C}(\mathcal{S}), J) = \mathcal{C}(\mathrm{Proj}(\mathcal{S}, J))$.

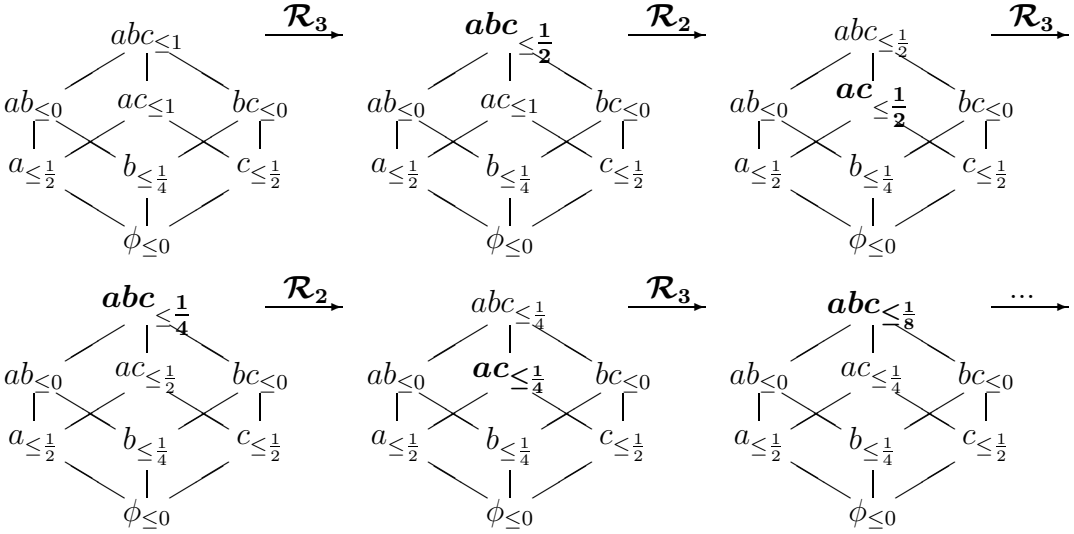


Figure 3.3: "Random" application of the rules can lead to infinite loops

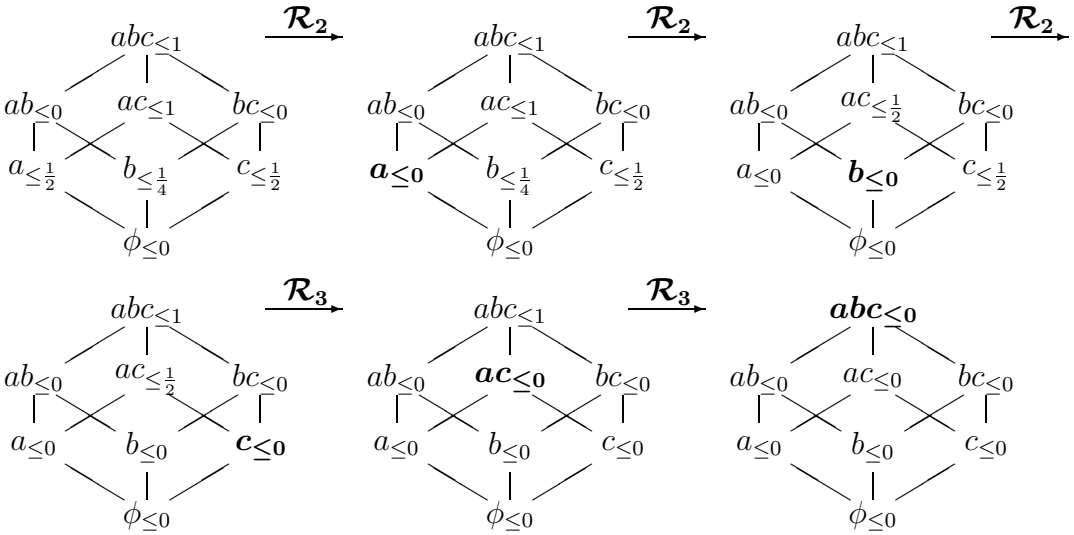


Figure 3.4: Systematic application of the rules avoids infinite computations

Input: System of rare sets $\mathcal{S} = \{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$ over $\mathcal{I}$.
Output: Completion of $\mathcal{S}$.

Complete(S)
  $p_\phi = 0$
  TopDwn(S)
  BotUp(S)

TopDwn(S)
  for $i = n$ **downto** 1 **do**
    for all itemsets I of cardinality i **do**
      make $p_I = \min_{I \subseteq J}(p_J)$

BotUp(S)
  for $i = 1$ **to** n **do**
    for all itemsets I of cardinality i **do**

$$\text{make } p_I = \min_{\mathcal{K},\ \text{minimal } k\text{-cover of } I} \left(\frac{\sum_{K \in \mathcal{K}} p_K}{k} \right)$$

Figure 3.5: Algorithm Complete for finding the completion of the system $\mathcal{S} = \{rare(I) \leq p_I \mid I \subseteq \mathcal{I}\}$ over $\mathcal{I}$

Proof
(1) is straightforward.
(2) Let

$$\mathcal{C}(\text{Proj}(\mathcal{S}, J)) = \{rare(I) \leq p_I \mid I \subseteq J\} \quad .$$

Then, for every $I \subseteq J$, we can construct a proof-database $\mathcal{D}_I$, such that $rare(I, \mathcal{D}_I) = p_I$, and for all $I' \subseteq J$, $rare(I', \mathcal{D}_I) \leq p_{I'}$.[2] We will now extend this proof-database $\mathcal{D}_I$ over J to a proof-database $\widehat{\mathcal{D}_I}$ of I over $\mathcal{I}$. $\widehat{\mathcal{D}_I}$ contains the same number of transactions as $\mathcal{D}_I$, and is formed by adding all items in $\mathcal{I} - J$ to every transaction in $\mathcal{D}_I$. $\widehat{\mathcal{D}_I}$ satisfies $\mathcal{S}$, since it is constructed in such a way that for all $I' \subseteq \mathcal{I}$ holds that

$$rare(I', \widehat{\mathcal{D}_I}) = rare(I' \cap I, \mathcal{D}_I) \leq p_{I' \cap I} \leq^{(\boldsymbol{\mathcal{R}_2})} p_{I'} \quad .$$

Therefore, $\mathcal{C}(\mathcal{S})$ must contain $rare(I) \leq p_I$, since $\widehat{\mathcal{D}_I}$ is a proof-database for $rare(I) \leq p_I$. □

Theorem 11 *The algorithm Complete in Fig. 3.5 computes the completion of the system of rare sets $\mathcal{S}$.*

[2]The existence of this proof-database can easily be established from the proof of the completeness of $\boldsymbol{\mathcal{R}_1}$, $\boldsymbol{\mathcal{R}_2}$ and $\boldsymbol{\mathcal{R}_3}$.

Proof
We will prove this theorem by induction on $|\mathcal{I}|$. In the base case $|\mathcal{I}| = 0$ the theorem is trivially true. Suppose the theorem holds for $1, \ldots, |\mathcal{I}| - 1$.

$\text{BotUp}(\text{Proj}(\text{TopDwn}(\mathcal{S}), J)) = \text{Proj}(\text{BotUp}(\text{TopDwn}(\mathcal{S})), J)$, and
$\text{TopDwn}(\text{Proj}(\text{TopDwn}(\mathcal{S}), J)) = \text{Proj}(\text{TopDwn}(\mathcal{S}), J)$
with $J \subseteq \mathcal{I}$.

Therefore, for all $J \subset \mathcal{I}$ it holds that:

$$\begin{array}{rcll} \text{Proj}(\mathcal{C}(\mathcal{S}), J) & = & \text{Proj}(\mathcal{C}(\text{TopDwn}(\mathcal{S})), J) & (\mathcal{S} \models \text{TopDwn}(\mathcal{S})) \\ & = & \mathcal{C}(\text{Proj}(\text{TopDwn}(\mathcal{S}), J)) & (\text{Lemma 11}) \\ & = & \text{BotUp}(\text{Proj}(\text{TopDwn}(\mathcal{S}), J)) & (\text{Induction hypothesis}) \\ & = & \text{Proj}(\text{BotUp}(\text{TopDwn}(\mathcal{S})), J) & (\boldsymbol{\mathcal{R}_3} \text{ only uses subsets}) \end{array}$$

We now show that the rareness value for $\mathcal{I}$ in $\text{BotUp}(\text{TopDwn}(\mathcal{S}))$ equals the rareness value in $\mathcal{C}(\mathcal{S})$. This equality is straightforward, since all other rareness values between these two systems are equal, and $\mathcal{I}$ can only be adjusted by the bottom-up rule $\boldsymbol{\mathcal{R}_3}$, and this bottom-up rule is applied in the last step of BotUp(.). □

3.1.5 Extending the Axiomatization to Sparse Systems

Until now we required that in a system of frequent sets for every itemset a frequent set expression was present. Now we drop this requirement. This has however repercussions for the axiomatization.

Definition 11

- *A* sparse system *of rare sets is a collection*

$$\{rare(I) \leq p_I \mid I \in \mathcal{P}\}$$

  of rare set expressions, with $\mathcal{P} \subseteq 2^{\mathcal{I}}$. Hence, not every subset of $\mathcal{I}$ has to be present in the system.

- *A database $\mathcal{D}$ over $\mathcal{I}$* satisfies *a sparse system $\mathcal{S}$ if $\mathcal{D}$ satisfies $rare(I) \leq p_I$ for all I in $\mathcal{P}$.*

- *A sparse system $\mathcal{S}$* implies *a rare set expression $rare(I) \leq p$, if every database that satisfies $\mathcal{S}$, also satisfies $rare(I) \leq p$.*

- *A sparse system $\{rare(I) \leq p_I \mid I \in \mathcal{P}\}$ is* complete *if for all $rare(I) \leq p$ with $I \in \mathcal{P}$, that are implied by the system, $p_I \leq p$ holds.* □

The following proposition states that every complete sparse system can be extended to a complete full system.

Proposition 5 *Let $\mathcal{S} = \{rare(I) \leq p_I \mid I \in \mathcal{P}\}$ be a sparse system. The following two statements are equivalent:*

- *$\mathcal{S}$ is complete.*
- *There exists a complete full system $\overline{\mathcal{S}} = \{rare(I) \leq \widehat{p_I} \mid I \subseteq \mathcal{I}\}$, such that for all $I \in \mathcal{P}$, $p_I = \widehat{p_I}$ holds.*

Proof
($\Rightarrow$) Let $\mathcal{D}$ be an arbitrary database satisfying $\mathcal{S}$. Then $\mathcal{D}$ satisfies the system

$$\widehat{\mathcal{S}} = \{rare(I) \leq q_I \mid I \subseteq \mathcal{I}\} \ ,$$

with $q_I = p_I$ if $I \in \mathcal{P}$, and $q_I = 1$ else. Hence, $\mathcal{D}$ satisfies the complete system

$$\overline{\mathcal{S}} = C(\widehat{\mathcal{S}}) = \{rare(I) \leq c_I \mid I \subseteq \mathcal{I}\} \ .$$

Therefore, $\mathcal{D}$ satisfies the sparse system

$$\{rare(I) \leq c_I \mid I \in \mathcal{P}\} \ .$$

This system has to be equal to $\mathcal{S}$, because $\mathcal{S}$ is complete, and $c_I \leq p_I$ for all $I \subseteq \mathcal{I}$.
($\Leftarrow$) $\overline{\mathcal{S}}$ is complete. Therefore, for every $I \in \mathcal{P}$, there exists a proof-database $\mathcal{D}_I$ such that $\mathcal{D}_I$ satisfies $\overline{\mathcal{S}}$, and $rare(I, \mathcal{D}_I) = \widehat{p_I}$. Since $\mathcal{D}_I$ also satisfies $\mathcal{S}$, $\mathcal{S}$ must be complete. □

The proposition leads to the following algorithm for computing the completion $\mathcal{C}(\mathcal{S})$ of the sparse system $\mathcal{S} = \{rare(I) \leq p_I \mid I \in \mathcal{P}\}$.

1. Let $\overline{\mathcal{S}} = \{rare(I) \leq q_I \mid I \subseteq \mathcal{I}\}$, with $q_I = p_I$ if $I \in \mathcal{P}$, else $q_I = 1$.
2. Compute the completion $\mathcal{C}(\overline{\mathcal{S}}) = \{rare(I) \leq c_I \mid I \subseteq \mathcal{I}\}$ of $\overline{\mathcal{S}}$ with the methods in Section 3.1.4.
3. Let $\mathcal{C}(\mathcal{S}) = \{rare(I) \leq c_I \mid I \in \mathcal{P}\}$.

However, it is clear that when the number of sets in $\mathcal{P}$ is small, this approach is not very efficient. Suppose that we are given a sparse system with $|\mathcal{P}| = m$ rare set expressions over a set with $|\mathcal{I}| = n$ items. To compute the completion, we calculate the completion of a system with 2^n expressions, where the input contained m expressions. The following proposition shows that there are more efficient ways to calculate the completion of a sparse system. It shows that we do not need all subsets of $\mathcal{I}$.

Theorem 12 *The following are equivalent:*

1. *The sparse system* $\mathcal{S} = \{rare(I_1) \leq p_1, \ldots, rare(I_n) \leq p_n\}$ *is complete*
2. $\mathcal{S}$ *satisfies*

   $\boldsymbol{\mathcal{S}_1}$ $p_\phi = 0$

   $\boldsymbol{\mathcal{S}_2}$ *If* $I_2 \subseteq I_1$, *then* $p_{I_2} \leq p_{I_1}$

   $\boldsymbol{\mathcal{S}_3}$ *Let* $\mathcal{M}$ *be a minimal* k*-cover of* I_i. *Then*

   $$p_{I_i} \leq \frac{\sum_{M \in \mathcal{M}} \min_{M \subseteq I_j}(p_{I_j})}{k}.$$

3. $\mathcal{S}$ *satisfies*

   $\boldsymbol{\mathcal{S}_1}$ $p_\phi = 0$

   $\boldsymbol{\mathcal{S}_2}$ *If* $I_2 \subseteq I_1$, *then* $p_{I_2} \leq p_{I_1}$

   $\boldsymbol{\mathcal{X}}$ *Let* $\mathcal{M}$ *be a bag over* $\{I_j \cap I \mid 0 \leq j \leq n\}$ *with minimal degree* k. *Then*

   $$p_I \leq \frac{\sum_{M \in \mathcal{M}} \min_{M \subseteq I_j}(p_{I_j})}{k} \quad .$$

Proof
$1 \Leftrightarrow 2$ **Soundness** of $\boldsymbol{\mathcal{S}_1}$, $\boldsymbol{\mathcal{S}_2}$, and $\boldsymbol{\mathcal{S}_3}$ is straightforward.
Completeness. Suppose the sparse system

$$\mathcal{S} = \{rare(I) \leq p_I \mid I \in \mathcal{P}\}$$

satisfies $\boldsymbol{\mathcal{S}_1}$, $\boldsymbol{\mathcal{S}_2}$, and $\boldsymbol{\mathcal{S}_3}$. Let

$$\mathcal{S}' = \{rare(I) \leq p'_I \mid I \subseteq \mathcal{I}\} \quad ,$$

with $p'_I = p_I$ if $I \in \mathcal{P}$, and $p'_I = 1$ else. Suppose

$$\mathcal{C}(\mathcal{S}') = \{rare(I) \leq q_I \mid I \subseteq \mathcal{I}\} \quad .$$

We show by contradiction that for all $I \in \mathcal{P}$, $p_I = q_I$ holds. Suppose there is a $I \in \mathcal{P}$ such that $p_I \neq q_I$.

$$\mathcal{C}(\mathcal{S}') = \text{BotUp}(\text{TopDwn}(\mathcal{S}')) \ ,$$

by Theorem 11. Since $\mathcal{S}$ satisfies $\boldsymbol{\mathcal{S}_1}$ and $\boldsymbol{\mathcal{S}_2}$, the rareness of I in $\mathcal{C}(\mathcal{S}')$ comes from the bottom-up step, and thus there exists a minimal k-cover $\mathcal{K}$ over the subsets of I, such that $\frac{\sum_{K \in \mathcal{K}} q_K}{k} < p_I$. The q_K's in this step can on their

turn be obtained in the top-down step, or in the bottom-up step. If q_K was obtained in the top-down step, then it is easy to see that $q_K = \min_{K \subseteq I_i} p_{I_i}$; that is, the minimum rareness of all supersets of K that were given as input. In the other case, q_K was obtained by a bottom-up step. In that case, there exists a minimal l-cover $\mathcal{L}$ over the subsets of L, such that $q_L = \sum_{L' \in \mathcal{L}} q_{L'}$. We now construct a kl-cover $\mathcal{K}'$ of K as follows:

$$\mathcal{K}' = (\mathcal{K} - \langle L \rangle) \bigcup \mathcal{L} \ .$$

$\mathcal{K}'$ is a kl-cover. In this way we can get rid of all q_L's that were obtained by application of a bottom-up step, because we can iteratively replace each q_L that was obtained by application of $\boldsymbol{\mathcal{R}_3}$, by a sum of $q_{L'}$'s, where all $L' \subset L$. When these L' are obtained by $\boldsymbol{\mathcal{R}_3}$, we can replace them by $q_{L''}$ of even smaller sets L''. Since the singleton sets can only be obtained by $\boldsymbol{\mathcal{R}_2}$, this recursion must stop, and thus there exists an m-cover $\mathcal{M}$ such that

$$\frac{\sum_{M \in \mathcal{M}} q_M}{m} < p_I \ ,$$

and all q_M's are obtained by $\boldsymbol{\mathcal{R}_2}$. As such, for all M,

$$q_M = \min_{M \subseteq I_i} p_{K_i} \ ,$$

and thus

$$\frac{\sum_{M \in \mathcal{M}} \min_{M \subseteq I_i} p_{I_i}}{m} < p_I \ .$$

There is still one problem: $\mathcal{M}$ is not necessarily minimal. We can cope with this problem in exactly the same way as at the end of the proof of Theorem 9.

$2 \Leftrightarrow 3$ Suppose system

$$\mathcal{S} = \{rare(I_1) \leq p_1, \ldots, rare(I_n) \leq p_n\}$$

satisfies $\boldsymbol{\mathcal{S}_1}$, $\boldsymbol{\mathcal{S}_2}$, but does not satisfy $\boldsymbol{\mathcal{S}_3}$. We will show that it also does not satisfy $\boldsymbol{\mathcal{X}}$. Hence, there exists a bag $\mathcal{M}$ with minimal degree k and a set I such that

$$p_{I_i} > \frac{\sum_{M \in \mathcal{M}} \min_{M \subseteq I_j}(p_{I_j})}{k} \ .$$

For each $M \in \mathcal{M}$, fix a $I_M \in \{I_1, \ldots, I_n\}$, such that $M \subseteq I_M$, and $p_{I_M} = \min_{M \subseteq I_j}(p_{I_j})$. Let $\mathcal{K}$ be the following bag: $\langle I_M \cap K \mid M \in \mathcal{M} \rangle$. The minimal degree of $\mathcal{K}$ is at least k (since $M \subseteq I_M \cap I$ for all $M \in \mathcal{M}$), and hence

$$p_{I_i} > \frac{\sum_{K \in \mathcal{K}} min_{M \subseteq I_j}(p_{I_j})}{mdeg(\mathcal{K})} \ .$$

This inequality is a violation of $\boldsymbol{\mathcal{X}}$.

The other direction is trivial, since $\boldsymbol{\mathcal{S}_3}$ is equivalent to:

> Let $\mathcal{M}$ be a bag over the subsets of I with minimal degree k. Then
> $$p_I \leq \frac{\sum_{M \in \mathcal{M}} \min_{M \subseteq I_j}(p_{I_j})}{k} \ .$$

Since $\boldsymbol{\mathcal{X}}$ is a specialization of this rule, $\boldsymbol{\mathcal{X}}$ holds whenever $\boldsymbol{\mathcal{S}_3}$ holds. □

Application of Sparse Systems

Suppose only the frequencies for the single-itemsets are given, and we want to derive a lower bound on itemset I. Using a sparse system, the problem is equivalent to finding the completion of the sparse system

$$\mathcal{S} = \{freq(\{i\}) \geq p_i \mid i \in \mathcal{I}\} \cup \{freq(I) \geq 0\} \, .$$

It is easy to see that $\mathcal{C}(\mathcal{S})$ contains

$$freq(I) \geq \left(\sum_{i \in I} p_i - (|I| - 1) \right) \ ,$$

since $\langle \{i\} \mid i \in I \rangle$ is the only minimal cover of I using the single-itemsets.

3.1.6 Complexity of Deciding and Computing Completion

The following lemma is of key importance in assessing the complexity of deciding whether a sparse system of frequent sets is complete.

Lemma 12 *Let $\mathcal{S} = \{freq(I_1) \geq p_1, \ldots, freq(I_n) \geq p_n\}$ be a complete sparse system of frequent sets over I. For every set I_j, $j = 1 \ldots n$ there exists a proof database $\mathcal{D}$ such that $\mathcal{D}$ only contains transactions of the form $(tid, \mathcal{I} - i)$, with i an item in I_j.*

Proof
Because $\mathcal{S}$ is complete, there exists a proof database $\mathcal{D}$ for I_j in $\mathcal{S}$. We now transform this database in the following way: first, we add to every transaction T in $\mathcal{D}$ the items $\mathcal{I} - I_j$. In this way we get the database $\mathcal{D}'$. Second, for

every transaction $(tid, J) \in \mathcal{D}'$ such that $I_j - J$ is nonempty, we randomly pick one item i in $I_j - J$, and we add the items in $\mathcal{I} - i$ to this transaction. The resulting database is called $\mathcal{D}''$. In this way we do not affect the frequency of the set I_j. From $\mathcal{D}$ to $\mathcal{D}'$, we only added items that were not in I_j, thus $freq(I_j, \mathcal{D}) = freq(I_j, \mathcal{D}')$. In the second step, from $\mathcal{D}'$ to $\mathcal{D}''$, we added items to transactions that did not contain I_j. But, we made sure that we did not add all items of I_j, and thus, the number of transactions containing I_j remained the same. Therefore, $freq(I_j, \mathcal{D}) = freq(I_j, \mathcal{D}'')$. For the other itemsets, the frequency can only become higher from $\mathcal{D}$ to $\mathcal{D}''$, since we only added items. Thus, $\mathcal{D}''$ still satisfies $\mathcal{S}$, and $freq(I_j, \mathcal{D}) = p_j$. □

Example 11 *Consider the proof-database*

$\mathcal{D}$ =

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

for $freq(abc) \geq 0.2$ *in*

$$\mathcal{S} = \left\{ \begin{array}{cccc} freq(ab) \geq 0.6 & freq(bc) \geq 0.2 & freq(ac) \geq 0.2 & freq(abc) \geq 0.2 \\ freq(d) \geq 0.2 & freq(ad) \geq 0.1 & freq(ef) \geq 0 & \end{array} \right\}.$$

We construct $\mathcal{D}'$ *by adding* d, e, *and* f *to every transaction in* $\mathcal{D}$.

$\mathcal{D}'$ =

TID	Items
1	a, b, c, d, e, f
2	a, b, d, e, f
3	a, b, d, e, f
4	b, d, e, f
5	c, d, e, f

Then, every transaction that does not contain abc *is extended until it contains all items of* abc *but one. In this way we get* $\mathcal{D}''$. *For example, to transaction* 4 *we add* c, *and to transaction* 5 *we add* a.

$\mathcal{D}''$ =

TID	Items
1	a, b, c, d, e, f
2	a, b, d, e, f
3	a, b, d, e, f
4	b, c, d, e, f
5	a, c, d, e, f

$\mathcal{D}''$ is still a proof-database for freq(abc) ≥ 0.2 in $\mathcal{S}$, and $\mathcal{D}''$ only has transactions (tid, J) with J one of $\mathcal{I} - a$, $\mathcal{I} - b$, and $\mathcal{I} - c$. □

The lemma allows us to restate the problem of finding the best lower bound for I_j as minimizing the frequency of I_j over these special kinds of databases.

Theorem 13 *Let $\mathcal{S} = \{freq(I_1) \geq p_1, \ldots, freq(I_n) \geq p_n\}$ be a sparse system of frequent sets. $\mathcal{S}$ is complete if and only if for each $j = 1 \ldots n$, p_j equals the solution of the following linear programming problem:*

$$\textit{Minimize } x \textit{ subject to } \begin{cases} \left(\sum_{i \in I_j - I_1} x_i\right) + x \geq p_1 \\ \quad \ldots \\ \left(\sum_{i \in I_j - I_n} x_i\right) + x \geq p_n \\ \\ \left(\sum_{i \in I} x_i\right) + x = 1 \\ x_i \geq 0 \qquad \forall i \in I \\ x \geq 0 \end{cases}$$

Proof
If. Suppose l is the solution of the linear program. Then there exist values for the variables such that all inequalities is satisfied. Let d be the least common multiple of the denominators of the values for the variables. The database that consists of dx_i transactions with as set of items $\mathcal{I} - i$, for all $i \in I_j$, and x transactions with as set of items $\mathcal{I}$, is a proof database for I_j in $\mathcal{S}$.
Only If. Because of Lemma 12, there exists a proof database $\mathcal{D}$ for I_j in $\mathcal{S}$ that only contains transactions with as set of items either $\mathcal{I}$ or $\mathcal{I} - i$ for an $i \in I_j$. The assignments

$$x := \frac{|\{(tid, I) \in \mathcal{D} \mid I = \mathcal{I}\}|}{|\mathcal{D}|} \text{, and } x_i := \frac{|\{(tid, I) \in \mathcal{D} \mid I = \mathcal{I} - i\}|}{|\mathcal{D}|}$$

are a solution to the system of inequalities. Because of the If-part, this solution results in the smallest value for x, because otherwise $\mathcal{S}$ would not be complete. □

Corollary 1 *Deciding whether a sparse system of frequent sets is complete, and computing the completion of a sparse system of frequent sets can both be done in polynomial time.*

Proof
It is well known that linear programming can be done in polynomial time [69]. □

3.2 Upper Bounds

3.2.1 System of Infrequent Sets

We introduce a system of infrequent sets as a full collection of infrequent set expressions. Logical implication and completeness of systems are as defined for sets of frequency constraints. In this chapter we will often use $freq(I) \leq l$ to denote the frequency constraint $freq(I) \in [l, 1]$.

Definition 12 *Let $\mathcal{I} = \{i_1, \ldots, i_n\}$ be a set of items, and $\mathcal{P}$ be a set of subsets of $\mathcal{I}$.*

- *A* (sparse) system of infrequent sets *over $\mathcal{I}$ is a collection*

$$\{freq(I) \leq u_I \mid I \in \mathcal{P}\}$$

  of infrequent set expressions.

- *A system of infrequent sets $\mathcal{S} = \{freq(I) \leq u_I \mid I \in \mathcal{P}\}$ is* complete *if for all $freq(I) \leq u$ with $I \in \mathcal{P}$, that are logically implied by the system, $u_I \leq u$ holds.*

□

3.2.2 Axioms for Complete Systems of Infrequent Sets

The following two axioms are sound and complete for complete systems of infrequent set expressions.

$\boldsymbol{\mathcal{IF}_1}$ $\{\} \leq 1$

$\boldsymbol{\mathcal{IF}_2}$ If $I_1 \subseteq I_2$, then $u_{I_2} \leq u_{I_1}$

Theorem 14 *$\mathcal{IF}_1$ and $\mathcal{IF}_2$ are sound and complete for complete systems of infrequent sets.*

Proof
Soundness is straightforward.
Completeness Suppose $\mathcal{S} = \{freq(I) \leq u_I \mid I \subseteq \mathcal{I}\}$ satisfies $\mathcal{IF}_1$ and $\mathcal{IF}_2$. We will proof the system is complete by constructing a database $\mathcal{D}_I$ for each I such that $\mathcal{D}_I \models \mathcal{S}$, and $freq(I, \mathcal{D}_I) = p_I$. Let $I = \{i_1, \ldots, i_k\}$, and $u_I = \frac{p}{q}$.

$\mathcal{D}_I$ =

TID	Items
1	$i_1, \ldots, i_k$
...	...
p	$i_1, \ldots, i_k$
$p+1$	ϕ
...	...
q	ϕ

It is now clear that $freq(I, \mathcal{D}_I) = u_I$, and if $J \subseteq I$, then due to $\mathcal{IF}_2$ is $u_I \leq u_J$, and thus $freq(J, \mathcal{D}_I) = u_I \leq u_J$. If $J \not\subseteq I$, then $freq(J, \mathcal{D}_I) = 0 \leq u_J$. □

Surprisingly, this axiomatization and the proof of the completeness is almost trivial, whereas in the frequent set case the axiomatization was very hard to prove, and included a rather complex third rule $\mathcal{F}_3$. In the infrequent case, apparently no counterpart of rule $\mathcal{F}_3$ is needed.

Because of these two simple axioms, deciding whether a system of infrequent sets is closed and computing the closure is very simple.

Theorem 15 *Deciding whether a system of infrequent sets is complete and computing the completion of a system of infrequent sets can both be done with only logarithmic space on a work tape.*

Proof
We will only give the procedure for computing the closure. The result for the decision problem is then trivial.

The main idea is to maintain counters. One counter is used to keep the position of the set in the input under consideration. The other counter is then used to scan the input searching for supersets of the current set. If such a superset is found, it is checked whether the upper bound on the frequency of this superset is lower than the lowest found so far. To keep track of the lowest frequency found so far, yet another, third counter is used. Once the end of the tape is reached while scanning for supersets, the current set is

printed on the output tape, together with the frequency pointed to by the third counter. Since, in the simulation, we only use three counters, the required space on the work tape is logarithmic in the size of the input. □

3.3 Lower and Upper Bounds Together

The axioms for the lower bounds and the axioms for the upper bounds together do not form a complete axiomatization for arbitrary systems of frequency constraints. Thus, in arbitrary systems, lower bounds of one set can influence the upper bounds of another and vice versa, as the following example illustrates.

Example 12 *Let $\mathcal{C}$ be the set of frequency constraints we also used in the example in Subsection 2.1.3, to show that* Apriori *does not prune perfectly.*

$$\mathcal{C} = \left\{ \begin{array}{lll} freq(a) \in \left[\frac{2}{3}, \frac{2}{3}\right], & freq(b) \in \left[\frac{2}{3}, \frac{2}{3}\right], & freq(c) \in \left[\frac{2}{3}, \frac{2}{3}\right], \\ freq(ab) \in \left[\frac{1}{3}, \frac{1}{3}\right], & freq(ac) \in \left[\frac{1}{3}, \frac{1}{3}\right], & freq(bc) \in \left[\frac{1}{3}, \frac{1}{3}\right] \end{array} \right\}$$

Notice that $\mathcal{C}$ is complete *in the sense that for all expressions $freq(I) \in [l, u]$ in $\mathcal{C}$ it holds that $\mathcal{C} \models_{tight} freq(I) \in [l, u]$. That is, $\mathcal{C}$ contains explicitly all information (implicitly) implied by it, as the following proof-database shows.*

$\mathcal{D}$ =

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

In Subsection 2.1.3, we saw that

$$\mathcal{C} \models_{tight} freq(abc) \in [0, 0] \ .$$

However, if we use the axioms for the lower bounds and the upper bounds, we can only derive

$$\mathcal{C} \models freq(abc) \in \left[0, \frac{1}{3}\right] \ .$$

A proof-database for $freq(abc) \leq \frac{1}{3}$ *in*

$$\mathcal{C}^U = \left\{ \begin{array}{lll} freq(a) \leq \frac{2}{3}, & freq(b) \leq \frac{2}{3}, & freq(c) \leq \frac{2}{3}, \\ freq(ab) \leq \frac{1}{3}, & freq(ac) \leq \frac{1}{3}, & freq(bc) \leq \frac{1}{3} \\ freq(abc) \leq \frac{1}{3} & & \end{array} \right\}$$

is for example the following:

$\mathcal{D}^U =$

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

□

This result is however not a surprise; FREQSAT is **NP**-complete, and deriving both lower and upper bounds can be done deterministically in polynomial time.

4

Point Intervals

In this chapter we discuss the most interesting special case. This case is based on the information we have in the Apriori-algorithm.

The main goal of this chapter is to present several new methods to identify redundancies in the set of all frequent itemsets and to exploit these redundancies, resulting in a concise representation of all frequent itemsets and significant performance improvements of a mining operation. Therefore, we study deduction rules for the entailment of tight bounds on a target itemset I, when we know the frequencies of all the subsets of I exact. That is, we give a complete set of deduction rules to derive the interval $[l, u]$ such that

$$\mathcal{C} \models_{tight} freq(I) \in [l, u] \ ,$$

with $\mathcal{C}$ a set of frequency constraints that consists of exactly one expression $freq(J) = f_J$ for each $J \subset I$.

Based on these deduction rules, the notion of *derivable itemsets* is introduced. Let $\mathcal{D}$ be a given database, and let $freq(J, \mathcal{D}) = f_J$ for all itemsets J. An itemset I is called *derivable in* $\mathcal{D}$ if

$$\{freq(J) = f_J \mid J \subset I\} \models_{tight} freq(I) = f_I \ .$$

Hence, the frequency of I can be determined perfectly from the frequencies of the subsets. Derivable itemsets are interesting, in the sense that they represent redundant information. They will be very important in the applications we discuss in Chapter 6. We give interesting properties of derivable itemsets, and based on these properties we give the NDI-algorithm that finds all frequent itemsets that are not derivable.

Bibliographic Note Large parts of this chapter were already published in [14, 15]. The paper [15] received the *Best Paper Award* at the *European Conference on Principles of Data Mining and Knowledge Discovery (PKDD)* in Helsinki, 2002.

4.1 Deduction Rules

4.1.1 Fraction and Extension

We describe sound and complete rules for deducing tight bounds on the support of an itemset $I \subseteq \mathcal{I}$, if the supports of all its subsets are given. In order to do this, we will not consider itemsets that are no subset of I, and we can assume that all items in $\mathcal{D}$ are elements of I. Indeed, "projecting away" the other items in a transaction database does not change the supports of the subsets of I.

Definition 13 **(I-Projection)** *Let $I \subseteq \mathcal{I}$ be an itemset.*

- *The* I-projection of a transaction $T = (tid, J)$, *denoted* $\pi_I T$, *is defined as* $\pi_I T =_{def} (tid, I \cap J)$.
- *The* I-projection of a transaction database $\mathcal{D}$, *denoted* $\pi_I \mathcal{D}$, *consist of all* I*-projected transactions from* $\mathcal{D}$.

□

Lemma 13 *Let I, J be itemsets, with $J \subseteq I$. For every transaction database $\mathcal{D}$, the following holds:*

$$freq(J, \mathcal{D}) = freq(J, \pi_I \mathcal{D}).$$

Before we introduce the deduction rules, we first extend the definition of fraction and we introduce extension.

Definition 14 **(J-Fraction)** *Let I, J be itemsets, such that $J \subseteq I$. The* J-fraction *of $\pi_I \mathcal{D}$, denoted by $\mathcal{F}_J^I(\mathcal{D})$ is defined as*

$$\mathcal{F}_J^I(\mathcal{D}) =_{def} \frac{\{(tid, J') \in \mathcal{D} \mid J' \cap I = J\}}{|\mathcal{D}|} .$$

□

If $\mathcal{D}$ is clear from the context, we will write $\mathcal{F}_J^I$, and if $I = \mathcal{I}$, we will write $\mathcal{F}_J$. For each set I, the frequency of an itemset $J \subseteq I$ is then

$$freq(J, \mathcal{D}) = \sum_{J \subseteq K \subseteq I} \mathcal{F}_K^I .$$

Definition 15 (Extension) *Let $I \subseteq \mathcal{I}$ be an itemset. The* extension of I in $\mathcal{D}$, *denoted by $ext(I, \mathcal{D})$, consists of all transactions in $\mathcal{D}$ that contain I.* □

We will write $ext(I)$ if $\mathcal{D}$ is clear from the context.

4.1.2 Inclusion-Exclusion Principle

Let $I, J \subseteq \mathcal{I}$ be itemsets, and $J \subseteq I, I - J = \{a_1, \ldots, a_n\}$. Notice that

$$ext(I) = \bigcap_{i=1}^{n} ext(J \cup \{a_i\}) \ ,$$

and that

$$\left| \bigcup_{i=1}^{n} ext(J \cup \{a_i\}) \right| = |ext(J)| - \mathcal{F}_J^I \, |\mathcal{D}| \ .$$

From the well-known *inclusion-exclusion principle* [55, p.181] we learn

$$\begin{aligned} \left| \bigcup_{i=1}^{n} ext(J \cup \{a_i\}) \right| \ = \ & \sum_{1 \le i \le n} |ext(J \cup \{a_i\})| \\ & - \sum_{1 \le i < j \le n} |ext(J \cup \{a_i, a_j\})| \\ & + \ \cdots \\ & -(-1)^n |ext(I)| \ , \end{aligned} \tag{4.1}$$

and since

$$freq(J \cup \{a_{i_1}, \ldots, a_{i_\ell}\}) = \frac{|ext(J \cup \{a_{i_1}, \ldots, a_{i_\ell}\})|}{|\mathcal{D}|} \ ,$$

we obtain

$$\begin{aligned} (-1)^n freq(I) - \mathcal{F}_J^I \ = \ & -freq(J) + \sum_{1 \le i \le n} freq(J \cup \{a_i\}) \\ & - \sum_{1 \le i < j \le n} freq(J \cup \{a_i, a_j\}) \\ & + \ \cdots \\ & + (-1)^n \sum_{1 \le i \le n} freq(I - \{a_i\}) \end{aligned}$$

Thus,

$$freq(I) - (-1)^n \mathcal{F}_J^I = \sum_{J \subseteq J' \subset I} (-1)^{|I-J'|+1} freq(J') \ .$$

From now on, we will denote the sum on the right-hand side of this last equation by $\sigma(J, I)$. Notice that in $\sigma(J, I)$ exactly all supersets of J appear that are also subset of I.

Since $\mathcal{F}_J^I$ is always positive, we obtain the following theorem.

Theorem 16 *For all itemsets $I, J \subseteq \mathcal{I}$, with $J \subseteq I$, $\sigma(J, I)$ is a lower (upper) bound on $freq(I)$ if $|I - J|$ is even (odd). The difference $|freq(I) - \sigma(J, I)|$ is given by $\mathcal{F}_J^I$.*

We will refer to the rule involving $\sigma(J, I)$ as $\mathcal{R}_I(J)$ and omit I when clear from the context.

Corollary 2 *None of the rules $\mathcal{R}_I(J)$ is redundant. For all itemsets $J \subseteq I$ there exists a database $\mathcal{D}$ such that $\mathcal{R}_I(J)$ gives the unique best approximation for the frequency of I.*

Proof
According to Theorem 16 the difference $|freq(I) - \sigma(J, I)|$ is given by $\mathcal{F}_J^I$. Consider now a database in which for every subset $J' \neq J$ of I there is a transaction T with set of items J'. It is clear that in this database, for every subset $J' \neq J$ of I, $\mathcal{F}_{J'}^I > 0$, and $\mathcal{F}_J^I = 0$. Therefore, $|freq(I) - \sigma(J', I)| > 0$ for all subsets J' of I, except for J. Thus, $\mathcal{R}_I(J)$ gives the unique best approximation for the frequency of I. □

4.1.3 Completeness of the Rules

If for each subset $J \subset I$, the frequency $freq(J, \mathcal{D}) = f_J$ is given, then the rules $\mathcal{R}_I(.)$ allow for calculating lower and upper bounds on the frequency of I. Let l denote the greatest lower bound and u the least upper bound we can derive with these rules. Since the rules are sound, the frequency of I must be in the interval $[l, u]$. We will show that these bounds on the frequency of I are *tight*; that is, for every smaller interval $[l', u'] \subset [l, u]$, we can find a database $\mathcal{D}'$ such that for each subset J of I, $freq(J, \mathcal{D}') = f_J$, but the frequency of I is not within $[l', u']$.

Recall Lemma 2:

Let
$$\mathcal{C} = \{freq(I_1) \in [l_1, u_1], \ldots, freq(I_n) \in [l_n, u_n]\}$$
be a set of frequency constraints over $\mathcal{I}$. There exists a transaction database $\mathcal{D}$ over $\mathcal{I}$ that satisfies $\mathcal{C}$, if and only if the following system of inequalities $\mathcal{P}(\mathcal{C})$ has a rational solution in the variables X_I, $I \subseteq \mathcal{I}$.

$$\mathcal{P}(\mathcal{C}) =_{def} \begin{cases} \sum_{I \subseteq \mathcal{I}} X_I = 1 \\ l_i \leq \sum_{I_i \subseteq I \subseteq \mathcal{I}} X_I \leq u_i \qquad \forall i = 1, \ldots, n \end{cases}$$

Corollary 3 *Given a set of items $\mathcal{I}$, $I \subseteq \mathcal{I}$, and a rational number f_J for each $J \subseteq I$. There exists a transaction database $\mathcal{D}$ satisfying $\forall J \subseteq I : freq(J, \mathcal{D}) = f_J$ if and only if the following system of inequalities has a solution.*

$$\begin{cases} X_J \geq 0 & \forall J \subseteq I \\ \sum_{J \subseteq I} X_J = 1 & \\ \sum_{J \subseteq K \subseteq I} X_K = f_J & \forall J \subseteq I \end{cases}$$

Proof
The proof is based on Lemma 13 and Lemma 2. Using Lemma 13, we can assume that we are working in a transaction database $\mathcal{D}$ over I instead of over $\mathcal{I}$. The system of inequalities now follows from Lemma 2. □

Theorem 17 *Let I be an itemset, and let for each $J \subseteq I$, f_J be a rational number. There exists a database that satisfies*

$$\{freq(J) = f_J \mid J \subseteq I\},$$

if and only if $f_{\{\}} = 1$, and for all $J \subseteq I$, the rule $\mathcal{R}_I(J)$ is obeyed.

Therefore, for every itemset $I \subseteq \mathcal{I}$, the rules $\{\mathcal{R}_I(J) \mid J \subseteq I\}$ are sound and complete for deducing tight bounds on the frequency of I based on the frequencies of all subsets of I.

Proof
If. Let I be an itemset, and $\mathcal{D}$ a transaction database over I. Let f_J denote

$freq(J, \mathcal{D})$, for all $J \subseteq I$. From Lemma 3, we derive the following equalities.

$$\begin{cases} f_{\{\}} &= \mathcal{F}_{\{\}} + \mathcal{F}_a + \mathcal{F}_b + \mathcal{F}_c + \mathcal{F}_d + \ldots + \mathcal{F}_{ab} + \mathcal{F}_{ac} + \ldots + \mathcal{F}_I \\ f_a &= \mathcal{F}_a + \mathcal{F}_{ab} + \mathcal{F}_{ac} + \ldots + \mathcal{F}_{abc} + \mathcal{F}_{abd} + \ldots + \mathcal{F}_I \\ f_b &= \mathcal{F}_b + \mathcal{F}_{ab} + \mathcal{F}_{bc} + \ldots + \mathcal{F}_{abc} + \mathcal{F}_{abd} + \ldots + \mathcal{F}_I \\ \ldots &\ldots \ldots \\ f_{ab} &= \mathcal{F}_{ab} + \mathcal{F}_{abc} + \mathcal{F}_{abd} + \ldots + \mathcal{F}_I \\ \ldots &\ldots \ldots \\ f_{I-a} &= \mathcal{F}_{(I-a)} + \mathcal{F}_I \\ f_I &= \mathcal{F}_I \end{cases} \tag{4.2}$$

This system of equalities contains $2^{|I|}$ equations and $2^{|I|}$ variables. Thus, we can eliminate the fractions $\mathcal{F}_J$. The solution of this system is then:

$$\begin{cases} \mathcal{F}_{\{\}} &= f_{\{\}} - f_a - f_b - f_c - f_d + f_{ab} + \ldots - f_{abc} - \ldots + (-1)^{|I|} f_I \\ \mathcal{F}_a &= f_a - f_{ab} - f_{ac} - \ldots + f_{abc} + \ldots - f_{abcd} - \ldots + (-1)^{|I|-1} f_I \\ \ldots &\ldots \ldots \\ \mathcal{F}_{ab} &= f_{ab} - f_{abc} - f_{abd} - \ldots + f_{abcd} + f_{abce} + \ldots + (-1)^{|I|-2} f_I \\ \ldots &\ldots \ldots \\ \mathcal{F}_{I-a} &= f_{I-a} - f_I \\ \mathcal{F}_I &= f_I \end{cases} \tag{4.3}$$

Thus, the values of the f_J's determine the values of the $\mathcal{F}_J$'s uniquely and vice versa. Therefore, there is a 1-1 correspondence between the assignments of the f_J's and the assignments for the $\mathcal{F}_J$'s.

On the other hand, we know that the solution in the $\mathcal{F}_J$'s represents a valid transaction database if and only if the following conditions are satisfied by the solution:

$$\begin{cases} \mathcal{F}_{\{\}} \geq 0 \\ \mathcal{F}_a \geq 0 \\ \mathcal{F}_b \geq 0 \\ \ldots \ldots \ldots \\ \mathcal{F}_I \geq 0 \\ \mathcal{F}_{\{\}} + \mathcal{F}_a + \mathcal{F}_b + \ldots + \mathcal{F}_I = 1 \end{cases} \tag{4.4}$$

By applying these conditions to the solution of (4.2), we get the following

conditions.

$$\begin{cases} (-1)^{|I|} f_I & \geq & 1 - \Big[(-1)^{|I|}(f_{I-a} + f_{I-b} + \ldots) + (-1)^{|I|-1}(f_{I-ab} + \ldots) \\ & & + \ldots + f_a + f_b + \ldots\Big] \\ (-1)^{|I|-1} f_I & \geq & -f_a + f_{ab} + f_{ac} + \ldots - f_{abc} - \ldots + f_{abcd} \ldots \\ \ldots & \ldots & \ldots \\ (-1)^{|I|-2} f_I & \geq & -f_{ab} + f_{abc} + f_{abd} + \ldots - f_{abcd} - f_{abce} - \ldots \\ & & + f_{abcdef} + f_{abcdeg} + \ldots \\ \ldots & \ldots & \ldots \\ -f_I & \geq & f_{I-a} \\ f_I & \geq & 0 \\ f_{\{\}} & = & 1 \end{cases} \tag{4.5}$$

These inequalities are exactly the rules $\mathcal{R}_I(J)$, $J \subseteq I$. We can thus conclude that there exists a database that satisfies $\{freq(J) = f_J \mid J \subseteq I\}$ if all rules $\mathcal{R}_I(J)$, $J \subseteq I$ are satisfied, because a choice of f_J's that satisfies (4.5) corresponds to an assignment for the fractions that determine a transaction database. This database $\mathcal{D}$ must have $freq(J, \mathcal{D}) = f_J$, for all $J \subseteq I$, because of the 1-1 correspondence between the fractions and the frequencies.
Only If. This part is already established because we obtained the rules from the (sound) inclusion-exclusion principle. □

Example 13 *Consider the following transaction database.*

$\mathcal{D} =$

TID	Items
1	a, b, c
2	a, c, d
3	a, b, d
4	c, d
5	b, c, d
6	a, d
7	b, d
8	b, c, d
9	b, c, d
10	a, b, c, d

$$\begin{array}{llllll} f_a = \frac{5}{10}, & f_b = \frac{7}{10}, & f_c = \frac{7}{10}, \\ f_d = \frac{9}{10}, & f_{ab} = \frac{3}{10}, & f_{ac} = \frac{3}{10}, \\ f_{ad} = \frac{4}{10}, & f_{bc} = \frac{5}{10}, & f_{bd} = \frac{6}{10}, \\ f_{cd} = \frac{6}{10}, & f_{abc} = \frac{2}{10}, & f_{abd} = \frac{2}{10}, \\ f_{acd} = \frac{2}{10}, & f_{bcd} = \frac{4}{10}. \end{array}$$

Figure 4.1 gives the rules to determine tight bounds on the frequency of abcd. Using these deduction rules, we derive the following bounds on $freq(abcd, \mathcal{D})$ without counting in the database.

$$
\begin{cases}
f_{abcd} \geq f_{abc} + f_{abd} + f_{acd} + f_{bcd} - f_{ab} - f_{ac} - f_{ad} & \mathcal{R}(\{\}) \\
\qquad -f_{bc} - f_{bd} - f_{cd} + f_a + f_b + f_c + f_d - 1 & \\
f_{abcd} \leq f_a - f_{ab} - f_{ac} - f_{ad} + f_{abc} + f_{abd} + f_{acd} & \mathcal{R}(a) \\
f_{abcd} \leq f_b - f_{ab} - f_{bc} - f_{bd} + f_{abc} + f_{abd} + f_{bcd} & \mathcal{R}(b) \\
f_{abcd} \leq f_c - f_{ac} - f_{bc} - f_{cd} + f_{abc} + f_{acd} + f_{bcd} & \mathcal{R}(c) \\
f_{abcd} \leq f_d - f_{ad} - f_{bd} - f_{cd} + f_{abd} + f_{acd} + f_{bcd} & \mathcal{R}(d) \\
f_{abcd} \geq f_{abc} + f_{abd} - f_{ab} & \mathcal{R}(ab) \\
f_{abcd} \geq f_{abc} + f_{acd} - f_{ac} & \mathcal{R}(ac) \\
f_{abcd} \geq f_{abd} + f_{acd} - f_{ad} & \mathcal{R}(ad) \\
f_{abcd} \geq f_{abc} + f_{bcd} - f_{bc} & \mathcal{R}(bc) \\
f_{abcd} \geq f_{abd} + f_{bcd} - f_{bd} & \mathcal{R}(bd) \\
f_{abcd} \geq f_{acd} + f_{bcd} - f_{cd} & \mathcal{R}(cd) \\
f_{abcd} \leq f_{abc} & \mathcal{R}(abc) \\
f_{abcd} \leq f_{abd} & \mathcal{R}(abd) \\
f_{abcd} \leq f_{acd} & \mathcal{R}(acd) \\
f_{abcd} \leq f_{bcd} & \mathcal{R}(bcd) \\
f_{abcd} \geq 0 & \mathcal{R}(abcd)
\end{cases}
$$

Figure 4.1: Tight bounds on f_{abcd}. f_I denotes $freq(I)$

Lower bound: $f_{abcd} \geq \frac{1}{10}$ *(Rule $\mathcal{R}(ac)$)*
Upper bound: $f_{abcd} \leq \frac{1}{10}$ *(Rule $\mathcal{R}(a)$)*

Therefore, we can conclude, without having to scan the database, that the frequency of abcd in $\mathcal{D}$ is exactly $\frac{1}{10}$, while a standard monotonicity check would yield an upper bound of $\frac{2}{10}$. □

4.2 Non-Derivable Itemsets

Based on the deduction rules, it is possible to generate a summary of the set of frequent itemsets. Indeed, suppose that the deduction rules allow for deducing the frequency of a frequent itemset I *exactly*, based on the frequencies of its subsets. Then there is no need to explicitly count the frequency of I requiring a complete database scan; if we need the frequency of I, we can always derive it using the deduction rules. Such a set I, of which we can perfectly derive the frequency, will be called a *Derivable Itemset* (DI), all other itemsets are called *Non-Derivable Itemsets* (NDIs). For each set I, let LB_I (UB_I) denote the lower (upper) bound we can derive using the

deduction rules. Hence,

$$\begin{array}{rcl} LB(I) & =_{def} & \max\{\sigma(J,I) \mid |I-J| \text{ even}\} \ , \text{ and} \\ UB(I) & =_{def} & \min\{\sigma(J,I) \mid |I-J| \text{ odd}\} \ . \end{array}$$

Since $LB(I)$ and $UB(I)$ use the frequency of the subsets of I, they depend on the underlying database $\mathcal{D}$.

Definition 16 *An itemset I is called* derivable in $\mathcal{D}$ *if $LB(I) = UB(I)$.* □

We show in this section that the set of frequent NDIs allows for computing the frequencies of all other frequent itemsets, and as such, forms a *concise representation* [62] of the frequent itemsets. To prove this result, we first need to show that when a set I is not derivable, then neither are its subsets.

Lemma 14 (Monotonicity) *Let $I \subseteq \mathcal{I}$ be an itemset, and $i \in \mathcal{I} - I$ an item. Then*

$$\begin{array}{rcl} 2|UB(I\cup\{i\}) - LB(I\cup\{i\})| & & \\ & \leq & 2\min(|freq(I) - LB(I)|, |freq(I) - UB(I)|) \\ & \leq & |UB(I) - LB(I)| \ . \end{array}$$

In particular, if I is a DI, then also $I \cup \{i\}$ is a DI.

Proof
The proof is based on the fact that $\mathcal{F}_J^I = \mathcal{F}_J^{I\cup\{i\}} + \mathcal{F}_{J\cup\{I\}}^{I\cup\{i\}}$. From Theorem 16 we know that $\mathcal{F}_J^I$ is the difference between the bound calculated by $\mathcal{R}_I(J)$ and the real frequency of I. Let now J be such that the rule $\mathcal{R}_I(J)$ calculates the bound that is closest to the frequency of I. Then, the width of the interval $[LB(I), UB(I)]$ is at least $2\mathcal{F}_J^I$. Furthermore, $\mathcal{R}_{I\cup\{i\}}(J)$ and $\mathcal{R}_{I\cup\{i\}}(J\cup\{i\})$ are a lower and an upper bound on the frequency of $I\cup\{i\}$ (if $|I\cup\{i\} - (J\cup\{i\})|$ is odd, then $|I\cup\{i\} - J|$ is even and vice versa), and these bounds on $I\cup\{i\}$ differ respectively $\mathcal{F}_J^{I\cup\{i\}}$ and $\mathcal{F}_{J\cup\{I\}}^{I\cup\{i\}}$ from the real frequency of $I\cup\{i\}$. When we combine all these observations, we get:

$$UB(I\cup\{i\}) - LB(I\cup\{i\}) \leq \mathcal{F}_J^{I\cup\{i\}} + \mathcal{F}_{J\cup\{I\}}^{I\cup\{i\}} = \mathcal{F}_J^I \leq \frac{1}{2}(UB(I) - LB(I)) \ .$$

□

This lemma gives us the following valuable insights.

Corollary 4 *The width of the intervals shrinks exponentially with the size of the itemsets. Hence, every set I with $|I| > {}^2\log(|\mathcal{D}|)+1$ must be derivable in $\mathcal{D}$.*

Proof
For the singleton sets the bounds are $[0, 1]$. Let $I = \{i_1, \ldots, i_n\}$. Because of Lemma 14,

$$\begin{aligned} |UB(I) - LB(I)| &\leq \frac{|UB(I - i_n) - LB(I - i_n)|}{2} \\ &\leq \ldots \\ &\leq \frac{UB(i_1) - LB(i_1)}{2^{n-1}} = \frac{1}{2^{n-1}} \ . \end{aligned}$$

Furthermore, all frequencies counted in the database are of the form $\frac{n}{|\mathcal{D}|}$ with n a natural number, and hence, since no divisions are used in the rules, also are the bounds. Therefore, for every set I of size larger than $^2\log(\mathcal{D}) + 1$, we get bounds $\left[\frac{L}{|\mathcal{D}|}, \frac{U}{|\mathcal{D}|}\right]$ with $\frac{U-L}{|\mathcal{D}|} < \frac{1}{|\mathcal{D}|}$, with U and L natural numbers. This is only possible if $L = U$, and thus I is derivable. □
This remarkable fact is a strong indication that the number of large NDIs will be very small. This reasoning will be supported empirically in Section 4.5.

Corollary 5 *If I is a NDI, and $\mathcal{R}_I(J)$ gives the exact frequency of I, then all supersets $I \cup \{i\}$ of I will be DI's, with rules $\mathcal{R}_{I\cup\{i\}}(J)$ and $\mathcal{R}_{I\cup\{i\}}(J \cup \{i\})$. That is, if $freq(I, \mathcal{D}) = \sigma(J, I)$, then*

$$freq(I \cup \{i\}, \mathcal{D}) = \sigma(I \cup \{i\}, J) = \sigma(I \cup \{i\}, J \cup \{i\}) \ .$$

Proof
This can easily be derived from the proof of Lemma 14. □
We will use Corollary 5 to avoid checking all possible rules for $I \cup \{i\}$. This avoidance can be done in the following way: whenever we calculate bounds on the frequency of an itemset I, we remember the lower and upper bound $LB(I), UB(I)$. If I is a NDI; that is, $LB(I) \neq UB(I)$, then we will have to count its frequency. After we counted the frequency, the tests $freq(I, \mathcal{D}) = LB(I)$ and $freq(I, \mathcal{D}) = UB(I)$ are performed. If one of these two equalities obtains, then all supersets of I are derivable.

From Lemma 14, we easily obtain the following theorem, saying that the set of NDIs is a concise representation.

Theorem 18 *For a database $\mathcal{D}$, and a threshold t, let $NDIRep(\mathcal{D}, t)$ be the following set:*

$$NDIRep(\mathcal{D}, t) \ =_{def} \ \{(I, freq(I, \mathcal{D})) \mid freq(I, \mathcal{D}) \geq t \wedge LB(I) \neq UB(I)\}.$$

$NDIRep(\mathcal{D}, t)$ is a concise representation for the frequent itemsets. That is, for each itemset J not in $NDIRep(\mathcal{D}, t)$, we can decide whether J is frequent, and if J is frequent, we can derive its frequency from the information in $NDIRep(\mathcal{D}, t)$.

Proof
We show by induction on the cardinality of J how we can calculate this information from the set of frequent NDIs.
Base case. $J = \{\}$ is trivial, since $freq(\{\}) = 1$ always holds.
General case. Suppose we know of each subset I of J whether it is frequent, and if I is frequent, we know $freq(I, \mathcal{D})$ exact. If one of the subsets is infrequent, J must be infrequent as well. If all subsets are frequent, then we know all the frequencies. These frequencies allow us to apply the deduction rules and to derive bounds $[l, u]$ on the frequency of J. If $l = u$, we know the frequency of J exactly. If $l \neq u$, then J is a NDI, and thus either J is in *NDIRep*, together with its frequency, or J is infrequent. □

Suppose that we want to build the entire set of frequent itemsets starting from the concise representation *NDIRep*. We can then use Corollary 5 to improve the performance of deducing all frequencies. Suppose we need to deduce the frequency of derivable sets I, J, with $J \subseteq I$. Instead of trying all rules to find the exact frequency for I, we first evaluated J. Since J is a DI, there is a rule $\mathcal{R}_J(K)$ that gives the exact frequency of J. Using 5, we know that $\mathcal{R}_I(K)$ gives the exact frequency of I. Hence, for I we only have to evaluate the rule $\mathcal{R}_I(K)$.

4.3 The NDI-Algorithm

Based on the results in the previous section, we propose a level-wise algorithm to find all frequent NDIs. Since derivability is monotone, we can prune an itemset if it is derivable. This gives the NDI-algorithm as shown in Figure 4.2. The correctness of the algorithm follows from the results in Lemma 14.

Since evaluating all rules can be very cumbersome (step 15), in the experiments we show what the effect is of only using a couple of rules. We will say that we use rules *up to depth* k if we only evaluate the rules $\mathcal{R}_I(J)$ for $|I - J| \leq k$. Experiments show that in most cases, the gain of evaluating rules up to depth k instead of up to depth $k - 1$ typically quickly decreases if k increases. Therefore, we can conclude that in practice most pruning is done by the rules of limited depth.

Input: Transaction database $\mathcal{D}$, threshold t.
Output: Set NDI of all frequent non-derivable sets in $\mathcal{D}$.

```
(1)  NDI(D,s)
(2)      i := 1; NDI := {}; C_1 := {{i} | i ∈ I};
(3)      for all I in C_1 do I.l := 0; I.u := 1;
(4)      while C_i not empty do
(5)          Count the frequencies of all candidates in C_i
             in one pass over D;
(6)          F_i := {I ∈ C_i | freq(I,D) ≥ t};.
(7)          NDI := NDI ∪ F_i;
(8)          Gen := {};
(9)          for all I ∈ F_i do
(10)             if freq(I,D) ≠ I.l and freq(I,D) ≠ I.u then
(11)                 Gen := Gen ∪ {I};
(12)         PreC_{i+1} := AprioriGenerate(Gen);
(13)         C_{i+1} := {};
(14)         for all I ∈ PreC_{i+1} do
(15)             Compute bounds [l,u] on frequency of I;
(16)             if l ≠ u then I.l := l; I.u := u; C_{i+1} := C_{i+1} ∪ {I};
(17)         i := i + 1
(18)     end while
(19)     return NDI;
```

Figure 4.2: The NDI-algorithm. *AprioriGenerate* is the standard procedure of the Apriori-algorithm to generate new candidates. In fact, the set $AprioriGenerate(Gen)$ equals $\{I \in \mathcal{I} \mid |I| = i+1, \forall i \in I : I - \{i\} \in Gen\}$.

4.4 Halving Intervals at Minimal Cost

One of the main disadvantages of the algorithm proposed in the last section is the fact that calculating the results of all rules can be very hard for large sets I. Indeed, the number of rules is exponential in the size of the set I. For each subset J of I, we need to evaluate the rule $\mathcal{R}_I(J)$. Also, the length of the rules increases dramatically. The number of terms in the rule $\mathcal{R}_I(J)$ is $2^{|I-J|} - 1$. Thus, if we evaluate all rules brute force, the cost will be as high as

$$\sum_{i=0}^{|I|} \binom{|I|}{i} (2^i - 1) = 3^{|I|} - 2^{|I|} = \mathcal{O}(3^{|I|}) \ .$$

Clearly, for large $|I|$ this cost is unacceptable.

To overcome this problem we develop different strategies. First, we could only evaluate rules of limited depth; that is, rules $\mathcal{R}_I(J)$, where $|I - J|$ is limited by a predefined constant k. Together with Corollary 5, we can keep the monotonicity of derivability. A disadvantage of this approach is that we lose the guaranty that the interval size halves in each step. For example, Corollary 4 no longer holds.

We show how we can maintain the halving of the interval sizes while only evaluating two rules per itemset. The procedure is based on the proof of Lemma 14. Since

$$|\mathit{freq}(I, \mathcal{D}) - \sigma(J, I)| = \mathcal{F}_J^I = \mathcal{F}_J^{I \cup \{i\}} + \mathcal{F}_{J \cup \{i\}}^{I \cup \{i\}}$$

and

$$\mathcal{F}_J^{I \cup \{i\}} = |\mathit{freq}(I \cup \{i\}, \mathcal{D}) - \sigma(J, I \cup \{i\})|,$$
$$\mathcal{F}_{J \cup \{i\}}^{I \cup \{i\}} = |\mathit{freq}(I \cup \{i\}, \mathcal{D}) - \sigma(J \cup \{i\}, I \cup \{i\})|,$$

it is true that

$$|\mathit{freq}(I, \mathcal{D}) - \sigma(J, I)| = |\sigma(J, I \cup \{i\}) - \sigma(J \cup \{i\}, I \cup \{i\})| \ .$$

Suppose now that for each subset $I - i$ of I we remember the deduction rule $\mathcal{R}_{I-i}(\mathit{best}(I-i))$ that came closest to the actual frequency of $I-i$. We select among the sets $I - i$ the one with the smallest difference between the actual support $\mathit{freq}(I - i, \mathcal{D})$ and the bound calculated by $\mathcal{R}_{I-i}(\mathit{best}(I - i))$. Let $I - i$ be this set. Lemma 14 then guarantees that the rules $\mathcal{R}_I(\mathit{best}(I - i))$ and $\mathcal{R}_I(\mathit{best}(I - i) \cup \{i\})$ compute an interval that is at most half the size of the intervals of its supersets.

In the algorithm this adaptation results in a modification to step (15). We replace step (15) with the following steps.

(15a) % *Compute bounds* $[l, u]$ *on frequency of* I*;*
(15b) Let $i := \text{minarg}_{i \in I}(|\mathit{freq}(I - i) - \sigma(\mathit{best}(I - i), I - i)|)$
(15c) Calculate l_I and u_I with the rules $\mathcal{R}_I(\mathit{best}(I - i))$ and $\mathcal{R}_I(\mathit{best}(I - i) \cup \{i\})$.
(15d) Let $I.\mathit{rules} = \{\mathit{best}(I - i), \mathit{best}(I - i) \cup \{i\}\}$;

After we counted the frequency of a set I we have to do some bookkeeping to set $\mathit{best}(I)$ to the right set. This can for example be done in the loop (9)-(11): we add the following lines in the loop (9)-(11), after step (11):

(11b) Let $\mathit{best}(I) := \text{minarg}_{J \in I.\mathit{rules}}(|\mathit{freq}(I) - \sigma(J, I)|)$;

4.5 Experiments

4.5.1 Data set

The data set we used to perform the experiments is derived from the census-data set as available in the UCI KDD-repository [47]. This data set is *in se* a relational table, with 68 numerical attributes. We transformed this data set into a transaction database in the following way: every (attribute,value)-pair was considered as a different item. Notice that therefore a value a in attribute A denotes another item as the same value a in another attribute B. Using this convention, every tuple was transformed into a transaction with 68 items. In order to speed-up the experiments, we only used a random sample of 10000 transactions. The data set contains 396 different items.

4.5.2 Results

In the experiments give an empirical answer to the following questions:

(a) *Pruning.* How much can we reduce the search space using the deduction rules? We tested both the situation in which the lower bound is above the threshold (the set is certainly frequent), and the one in which the upper bound is below the threshold (the set is certainly infrequent).

(b) *Interval Width.* How accurate are the bounds in predicting the actual support of an itemset? We report, grouped by the cardinality of the itemsets, the average width of the intervals $[l, u]$.

(c) *Size of the concise representation.* How big is the concise representation $NDIRep(\mathcal{D}, s)$ w.r.t. the complete set of frequent itemsets?

(d) *Strength of the deduction rules.* Which one of the rules contributes most to the bounds? Can we restrict ourselves to only evaluating a couple of rules?

Pruning In this test we want to see how much pruning is performed by the deduction rules. The results are given in Table 4.1. We mined the transaction database at different support levels, and we record in every pass of the Apriori-algorithm the following measures: (a) the number of candidate itemsets, (b) the number of frequent itemsets, (c) the number of itemsets for which the lower bound is above the support threshold, and (d) the number of itemsets for which the upper bound is below the support threshold.

Support = 90%, all 396 items

	(a)	(b)	(c)	(d)
1	396	20		
2	190	159	151	0
3	750	598	592	152
4	1512	1170	1170	342
5	1469	1186	1186	283
		...		

Support = 10%, all 396 items

	(a)	(b)	(c)	(d)
1	396	133		
2	8778	5444	3085	0
3	131258	121875	117089	2089
4	1853220	1809695	1802860	35491
		...		

Support = 90%, 100 items

	(a)	(b)	(c)	(d)
1	100	20		
2	190	159	151	0
3	750	598	592	152
4	1512	1170	1170	342
5	1469	1186	1186	283
6	710	622	622	88
7	170	165	165	5
8	16	16	16	0
9	1	1	1	0

Support = 10%, 20 items

	(a)	(b)	(c)	(d)
1	20	16		
2	120	101	72	0
3	355	348	347	2
4	759	754	752	5
5	1091	1050	1050	41
6	985	974	974	11
7	623	621	621	2
8	278	278	278	0
9	82	82	82	0
10	14	14	14	0
11	1	1	1	0

(a) number of candidate itemsets,
(b) number of frequent itemsets,
(c) number of itemsets with $l \geq ts$, and
(d) number of itemsets with $u < ts$.

Table 4.1: Experiments w.r.t. pruning

It is important to remark that in these counts only the itemsets that are not pruned by the monotonicity rule are evaluated with the deduction rules. Thus, the numbers we give represent pruning *additional* to the monotonicity rule. From these tests it is clear that the amount of pruning done by the monotonicity rule can be improved dramatically. For example, in all tests, from pass 4 on, for almost all sets, we know in advance whether or not they are frequent.

Interval width In Table 4.2, we report the mean width of the intervals per iteration. We pay special attention to derivable sets. We mined for frequent itemsets at different support levels and we report for each loop of the Apriori-algorithm the following measures: (a) the number of frequent sets, (b) the mean interval width, (c) the number of candidate itemsets for which $l = u$, and (d) and (e) the number of candidate itemsets with interval width at most 0.1%, respectively 0.05%.

From the tests we see that the width decreases very fast. After pass 4, in all our tests, we know *exactly* the frequencies of all sets that follow.

Concise representations In this test we measure how large a concise representation of the set of frequent itemsets would be. Table 4.3 gives $|NDIRep|$ and the number of frequent sets. In the tests, the concise representation is much smaller than the actual set of frequent itemsets.

Strength of deduction rules We study how much the different rules contribute to the bounds. The amount of work one has to do to evaluate $\sigma(J, I)$ is exponential in $|I - J|$. Therefore, at a certain depth there must be a trade-off between on the one hand evaluating more rules and as such getting better bounds, and on the other hand evaluating less rules, and possibly counting too many itemsets. In Fig. 4.3, we illustrate the trade-off. We performed 9 experiments: in the ith experiment we only evaluate rules up to depth i. This implies that for $i < j$, we have more pruning in the jth experiment than in the ith, but at the cost of evaluating more rules. The horizontal axis contains the depth up to which we evaluate the rules. On the left axes we present the number of itemsets counted by the NDI-algorithm, and on the right axes the time needed for mining these itemsets. The figure shows that the number of itemsets we need to count decreases when we increase the depth. However, the evaluation time for the rules also increases. In the experiment evaluating rules of more than depth 2 does not give much extra reduction in the number of counts, while the evaluation time of the rules grows.

Support = 90%, all 396 items

	(a)	(b)	(c)	(d)	(e)
1	396				
2	190	2.16%	0	0	19
3	750	0.029%	313	625	697
4	1512	$\approx 0\%$	1494	1512	1512
5	1469	0%	1469	1469	1469
			...		

Support = 10%, 20 items

	(a)	(b)	(c)	(d)	(e)
1	20				
2	120	7.5%	0	0	0
3	355	0.21%	71	170	201
4	759	$\approx 0\%$	590	746	756
5	1091	$\approx 0\%$	1087	1091	1091
6	985	0%	985	985	985
7	623	0%	623	623	623
8	278	0%	278	278	278
9	82	0%	82	82	82
10	14	0%	14	14	14
11	1	0%	1	1	1

(a) number of frequent sets,
(b) mean interval width,
(c) number of sets with $l = u$,
(d) number of sets with width $\leq 0.1\%$, and
(e) number of sets with width $\leq 0.05\%$.

Table 4.2: Experiments w.r.t. interval width

Support	$\lvert\mathcal{I}\rvert$	#Freq	$\lvert NDIRep\rvert$
90%	100	3937	634
10%	20	4239	569
1%	10	255	113

Table 4.3: Experiments w.r.t. size of *NDIRep*

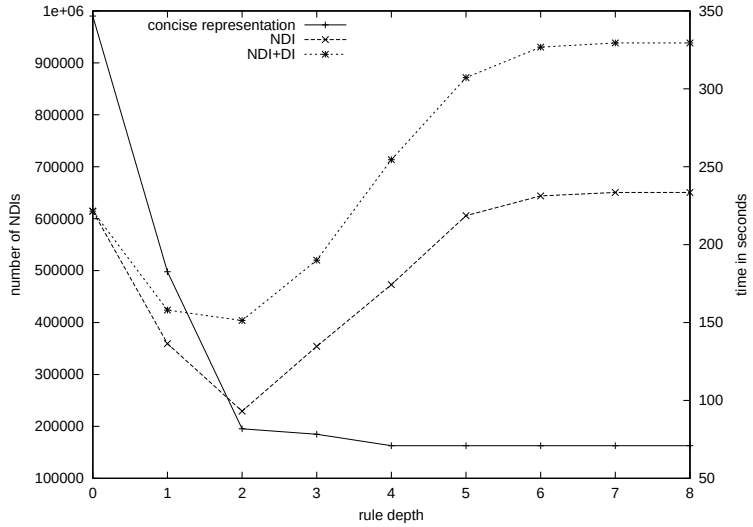


Figure 4.3: Strength of deduction rules.

For more experiments we refer to the paper [15].

4.6 Support versus Frequency

In the special case presented in this chapter, we could as well have used support instead of frequency. Indeed, none of the proofs relies on the fact that we use frequencies instead of support. This implies that *in this special case*,

$$\{freq(J) = f_J \mid J \subset I\} \models_{tight} freq(I) \in [l, u]$$

is true if and only if

$$\{support(J) = d \cdot f_J \mid J \subset I\} \models_{tight} freq(I) \in [d \cdot l, d \cdot u] \ ,$$

with d the least common multiplier of the denominators of the numbers in $\{f_J \mid J \subset I\}$. In [15], the results in this chapter are presented with supports. The equivalence between support and frequency in this case can also be proven without replacing all occurrences of $freq(.)$ by $support(.)$ in this chapter. Since none of the rules $\mathcal{R}_I J$ does use division, the bounds can always be expressed with d as denominator. Furthermore, the fractions in a database can be calculated from the frequencies, using only addition and substraction (see for example the system (4.3) in the proof of Theorem 17). Hence, the proof databases can always be constructed in such a way that they contain exactly d transactions.

5

Generic Construction of Axioms

In this chapter we show how we can construct an axiomatization for special cases of FREQSAT and T-FREQENT. This generic construction is based on the Fourier-Motzkin elimination method [22, pp. 84] for linear systems of inequalities. First we give a new and simpler system of inequalities based on the theory in Chapter 4. This system of inequalities has a solution if and only if there exists a transaction database that fulfills certain conditions. In this system of inequalities we then eliminate all spurious variables. The result is an axiomatization for the special case.

5.1 New Existence Condition

Let for each $I \subseteq \mathcal{I}$, f_I be a rational number. Theorem 17 states that there exists a database with $freq(I, \mathcal{D}) = f_I$ if and only if $f_{\{\}} = 1$, and for each $I \subseteq \mathcal{I}$, the rules $\mathcal{R}_{\mathcal{I}}(I)$ are satisfied. From this we can derive the following corollary.

Corollary 6 *The set of frequency constraints*

$$\mathcal{C} = \{freq(I_1) \in [l_1, u_1], \ldots, freq(I_n) \in [l_n, u_n]\}$$

is satisfiable if and only if there exist rational numbers f_I, $I \subseteq \mathcal{I}$ such that

(1) $f_{I_j} \in [l_j, u_j], \quad \forall j = 1, \ldots, n$, *and*

(2) $freq(I) = f_I$ *for all* $I \subseteq \mathcal{I}$, *respects the rules* $\mathcal{R}_{\mathcal{I}}(J)$, $J \subseteq \mathcal{I}$, *and* $f_{\{\}} = 1$.

Proof
The second condition makes sure that there exists a database $\mathcal{D}$ such that for all $J \subseteq \mathcal{I}$, $freq(J, \mathcal{D}) = f_J$. The first condition assures that $freq(I_j, \mathcal{D}) = f_{I_j}$ is in the interval $[l_j, u_j]$, for all $j = 1, \ldots, n$. □

Example 14 *The set of frequency constraints*

$$\mathcal{C} \;=\; \left\{ freq(a) \in \left[0, \frac{1}{2}\right], freq(ab) \in \left[\frac{1}{2}, \frac{3}{4}\right], freq(abc) \in \left[\frac{1}{2}, \frac{3}{4}\right] \right\}$$

has a solution if and only if the following system of inequalities has a solution:

$$\begin{cases} f_{abc} \geq 0 \\ f_{ab} \geq f_{abc} \\ f_{ac} \geq f_{abc} \\ f_{bc} \geq f_{abc} \\ f_{abc} \geq f_{ab} + f_{ac} - f_a \\ f_{abc} \geq f_{ab} + f_{bc} - f_b \\ f_{abc} \geq f_{ac} + f_{bc} - f_c \\ f_{ab} + f_{ac} + f_{bc} - f_a - f_b - f_c + 1 \geq f_{abc} \\ \\ f_a \geq 0 \\ \frac{1}{2} \geq f_a \\ f_b \geq \frac{1}{2} \\ \frac{3}{4} \geq f_a \\ f_{ab} \geq \frac{1}{2} \\ \frac{3}{4} \geq f_{ab} \end{cases}$$

□

5.2 Fourier-Motzkin Elimination

The *Fourier-Motzkin elimination method* [22, pp. 84] for systems of linear inequalities works as follows.

Let

$$\begin{cases} a_{11}x_1 & + & \ldots & + & a_{n1}x_n & \leq & b_1 \\ \ldots & & & & & & \\ a_{1m}x_1 & + & \ldots & + & a_{nm}x_n & \leq & b_m \end{cases}$$

be a system of inequalities. The Fourier-Motzkin elimination method allows for eliminating x_1, such that we get a system of inequalities in the variables $x_2, \ldots, x_n$ that has a solution if and only if the original system has a solution.

First, we rewrite each of the inequalities by isolating x_1. In this way we get three types of inequalities:

(1) $x_1 \geq \ldots$

(2) $x_1 \leq \ldots$

(3) Inequalities that do not contain x_1.

Let

$$L = \left\{ -\frac{a_{i2}}{a_{i1}}x_2 - \ldots - \frac{a_{in}}{a_{i1}}x_n + \frac{b_i}{a_{i1}} \;\middle|\; a_{i1} > 0 \right\}$$

$$U = \left\{ -\frac{a_{i2}}{a_{i1}}x_2 - \ldots - \frac{a_{in}}{a_{i1}}x_n + \frac{b_i}{a_{i1}} \;\middle|\; a_{i1} < 0 \right\}$$

The equalities of type (1) are exactly the inequalities $x_1 \geq \alpha_2 x_2 + \ldots + \alpha_n x_n + \beta$ such that $\alpha_2 x_2 + \ldots + \alpha_n x_n + \beta$ is in L. Similarly, the equalities of type (2) can be formed using the terms in U.

We now construct a new system that is equivalent to the original system, but that does not contain x_1. By equivalent we mean that the new system has a solution if and only if the original system has one. The new system is based on the observation that if we have an inequality $x_1 \geq l$, and an inequality $x_1 \leq u$, then there exists an x_1 fulfilling these two inequalities if and only if $l \leq u$ has a solution. Indeed, if there is a solution such that $l \leq u$, we only need to pick a x_1 in between the value of l and the value of u. This assignment makes both inequalities true. Also, if there does not exist a solution such that $l \leq u$, there is no hope in making both inequalities true, since, for this case, $x_1 \geq l$ automatically implies $x_1 > u$.

The new system thus consists of the inequalities of type (3) in the original system, plus all inequalities $l \leq u$, for all $l \in L$ and $u \in U$.

For example, eliminating x_1 in

$$\begin{aligned} x_1 + 3x_2 &\leq 5 \\ x_1 + x_4 - 7x_5 &\leq 3 \\ -x_1 + x_3 &\leq 5 \\ x_3 + x_5 &\leq 4 \end{aligned}$$

goes as follows: first we rewrite every formula such that we isolate x_1.

$$\begin{aligned} x_1 &\leq 5 - 3x_2 \\ x_1 &\leq 3 - x_4 + 7x_5 \\ -5 + x_3 &\leq x_1 \\ \\ x_3 + x_5 &\leq 4 \end{aligned}$$

Hence, $L = \{-5 + x_3\}$, and $U = \{5 - 3x_2, 3 - x_4 + 7x_5\}$ The new system is thus:

$$\begin{array}{rcl} -5 + x_3 & \leq & 5 - 3x_2 \\ -5 + x_3 & \leq & 3 - x_4 + 7x_5 \\ \\ x_3 + x_5 & \leq & 4 \end{array}$$

or, equivalent,

$$\begin{array}{rcl} 3x_2 + x_3 & \leq & 10 \\ x_3 + x_4 - 7x_5 & \leq & 8 \\ x_3 + x_5 & \leq & 4 \end{array}$$

This new system does no longer contain variable x_1, and has a solution if and only if the original system has one.

5.3 Construction of Axioms

Suppose now that we want to make axioms for the specific case that we have 3 items, and we know bounds on the sets a, b, and abc. We denote the hypothetical bounds on a set I by $[l_I, u_I]$. Using the existence conditions of Corollary 6, we get the following system.

$$\begin{cases} f_{abc} \geq 0 \\ f_{ab} \geq f_{abc} \\ f_{ac} \geq f_{abc} \\ f_{bc} \geq f_{abc} \\ f_{abc} \geq f_{ab} + f_{ac} - f_a \\ f_{abc} \geq f_{ab} + f_{bc} - f_b \\ f_{abc} \geq f_{ac} + f_{bc} - f_c \\ f_{ab} + f_{ac} + f_{bc} - f_a - f_b - f_c + 1 \geq f_{abc} \\ \\ f_a \geq l_a \\ u_a \geq f_a \\ f_b \geq l_b \\ u_b \geq f_a \\ f_{abc} \geq l_{abc} \\ u_{abc} \geq f_{abc} \end{cases}$$

Thus, given bounds on a, b, and abc, there exists a database that fulfils 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 f_I. For this, we

can use the Fourier-Motzkin elimination method to eliminate all unwanted variables.

First we eliminate f_a. The inequalities involving f_a are:

$$\begin{array}{rcl} l_a & \leq & f_a \\ f_{ab} + f_{ac} - f_{abc} & \leq & f_a \end{array}$$

$$\begin{array}{rcl} f_a & \leq & u_a \\ f_a & \leq & f_{ab} + f_{ac} + f_{bc} - f_b - f_c + 1 - f_{abc} \end{array}$$

This gives the following system:

$$\left\{ \begin{array}{l} f_{abc} \geq 0 \\ f_{ab} \geq f_{abc} \\ f_{ac} \geq f_{abc} \\ f_{bc} \geq f_{abc} \\ f_{abc} \geq f_{ab} + f_{bc} - f_b \\ f_{abc} \geq f_{ac} + f_{bc} - f_c \\ f_b \geq l_b \\ u_b \geq f_b \\ f_{abc} \geq l_{abc} \\ u_{abc} \geq f_{abc} \\ \\ u_a \geq l_a \\ f_{ab} + f_{ac} + f_{bc} - f_b - f_c + 1 - f_{abc} \geq l_a \\ u_a \geq f_{ab} + f_{ac} - f_{abc} \\ f_{bc} - f_b - f_c + 1 \geq 0 \end{array} \right.$$

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

$$\left\{ \begin{array}{l} 0 \leq u_a \\ 0 \leq u_b \\ 0 \leq u_{abc} \\ l_a \leq 1 \\ l_b \leq 1 \\ l_{abc} \leq 1 \\ l_a \leq u_a \\ l_b \leq u_b \\ l_{abc} \leq u_{abc} \\ \\ l_{abc} \leq u_a \\ l_{abc} \leq u_b \end{array} \right.$$

The first 9 conditions 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 $l \leq u$, $l \leq 1$, $u \geq 0$. These conditions look a little strange because we did not explicitly require that $l, u \in [0, 1]$. The last three conditions state a form of the monotonicity rule; the lower bound on *abc* must always be smaller than the upper bounds of a and b. Thus, these conditions together with the implicit assumption that $l, u \in [0, 1]$ for all bounds, gives the following 5 axioms for the special case in which bounds on a, b, and *abc* have been given:

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

5.4 Entailment

We can solve entailment problems with a slight variation on the method in last section.

Suppose that we want to entail formulas that give tight bounds on the frequency of *abc* 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:

$$\begin{cases} f_{abc} \geq 0 \\ f_{ab} \geq f_{abc} \\ f_{ac} \geq f_{abc} \\ f_{bc} \geq f_{abc} \\ f_{abc} \geq f_{ab} + f_{ac} - f_a \\ f_{abc} \geq f_{ab} + f_{bc} - f_b \\ f_{abc} \geq f_{ac} + f_{bc} - f_c \\ f_{ab} + f_{ac} + f_{bc} - f_a - f_b - f_c + 1 \geq f_{abc} \\ \\ f_a \geq l_a \\ u_a \geq f_a \\ f_b \geq l_b \\ u_b \geq f_a \end{cases}$$

In this system we eliminate all f_I's except for f_{abc}. This gives the following,

equivalent system:

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

The first 4 conditions are again existence conditions. The last 4 conditions show that

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

5.5 Examples

In this section we give some examples of special cases denoted by "$k \to l$, n items". We only concentrate on entailment problems. In the case "$k \to l$, n items" we assume that bounds have been given for all itemsets of cardinality k. We then present rules for the frequency of a set of cardinality l. For "$2 \to 3$, 4 items", for example, we assume that bounds on ab, ac, ad, bc, bd, and cd have been given, and we want to entail tight bounds for abc. We only present rules that involve the set of cardinality l, since only these rules are important for the entailment.

5.5.1 $2 \to 3$, 3 items

$$\begin{array}{lcl} 0 & \leq & f_{abc} \\ f_{abc} & \leq & u_{ab} \\ f_{abc} & \leq & u_{ac} \\ f_{abc} & \leq & u_{bc} \\ f_{abc} & \leq & 1 \end{array}$$

$$\begin{array}{lcl} l_{ac} + l_{bc} - 1 & \leq & f_{abc} \\ l_{ab} + l_{bc} - 1 & \leq & f_{abc} \\ l_{ab} + l_{ac} - 1 & \leq & f_{abc} \end{array}$$

$$\frac{l_{ab} + l_{ac} + l_{bc}}{2} \leq f_{abc}$$

5.5.2 $2 \to 3$, 4 items

$$
\begin{aligned}
f_{abc} &\leq u_{ab} \\
f_{abc} &\leq u_{ac} \\
f_{abc} &\leq u_{bc}
\end{aligned}
$$

$$
f_{abc} \leq 1
$$

$$
\begin{aligned}
f_{abc} &\leq 1 + u_{ad} - l_{cd} \\
f_{abc} &\leq 1 + u_{ad} - l_{bd} \\
f_{abc} &\leq 1 + u_{bd} - l_{cd} \\
f_{abc} &\leq 1 - l_{bd} + u_{cd} \\
f_{abc} &\leq 1 - l_{ad} + u_{bd} \\
f_{abc} &\leq 1 - l_{ad} + u_{cd}
\end{aligned}
$$

$$
\begin{aligned}
l_{ad} + l_{bc} - 1 &\leq f_{abc} \\
l_{ac} + l_{bd} - 1 &\leq f_{abc} \\
l_{ac} + l_{bc} - 1 &\leq f_{abc} \\
l_{ab} + l_{cd} - 1 &\leq f_{abc} \\
l_{ab} + l_{bc} - 1 &\leq f_{abc} \\
l_{ab} + l_{ac} - 1 &\leq f_{abc}
\end{aligned}
$$

$$
\begin{aligned}
l_{ac} + l_{bc} + l_{bd} - u_{cd} - 1 &\leq f_{abc} \\
l_{ac} + l_{ad} + l_{bc} - u_{cd} - 1 &\leq f_{abc} \\
l_{ab} + l_{bc} - u_{bd} + l_{cd} - 1 &\leq f_{abc} \\
l_{ab} + l_{ad} + l_{bc} - u_{bd} - 1 &\leq f_{abc} \\
l_{ab} + l_{ac} - u_{ad} + l_{cd} - 1 &\leq f_{abc} \\
l_{ab} + l_{ac} - u_{ad} + l_{bd} - 1 &\leq f_{abc}
\end{aligned}
$$

$$
f_{abc} \leq 1 + u_{ab} + u_{ac} - l_{ad} + u_{bc} - l_{bd} - l_{cd}
$$

$$
\begin{aligned}
-u_{ab} + l_{ac} + l_{ad} + l_{bc} + l_{bd} - u_{cd} - 1 &\leq f_{abc} \\
l_{ab} - u_{ac} + l_{ad} + l_{bc} - u_{bd} + l_{cd} - 1 &\leq f_{abc} \\
l_{ab} + l_{ac} - u_{ad} - u_{bc} + l_{bd} + l_{cd} - 1 &\leq f_{abc}
\end{aligned}
$$

$$
\frac{l_{ab} + l_{ac} + l_{bc} - 1}{2} \leq f_{abc}
$$

$$
\begin{aligned}
f_{abc} &\leq \frac{1 + u_{ab} - l_{ad} - l_{bd} + 2u_{cd}}{2} \\
f_{abc} &\leq \frac{1 + u_{ac} - l_{ad} + 2u_{bd} - l_{cd}}{2} \\
f_{abc} &\leq \frac{1 + 2u_{ad} + u_{bc} - l_{bd} - l_{cd}}{2}
\end{aligned}
$$

$$
\begin{array}{rcl}
f_{abc} & \leq & 1 + u_{ab} - l_{ad} - l_{bc} - l_{bd} + 2u_{cd} \\
f_{abc} & \leq & 1 + u_{ab} - l_{ac} - l_{ad} - l_{bd} + 2u_{cd} \\
f_{abc} & \leq & 1 + u_{ac} - l_{ad} - l_{bc} + 2u_{bd} - l_{cd} \\
f_{abc} & \leq & 1 - l_{ac} + 2u_{ad} + u_{bc} - l_{bd} - l_{cd} \\
f_{abc} & \leq & 1 - l_{ab} + u_{ac} - l_{ad} + 2u_{bd} - l_{cd} \\
f_{abc} & \leq & 1 - l_{ab} + 2u_{ad} + u_{bc} - l_{bd} - l_{cd} \\
l_{ad} + l_{bd} + l_{cd} - 2 & \leq & f_{abc} \\
\dfrac{2l_{ac} + l_{ad} + 2l_{bc} + l_{bd} - u_{cd} - 2}{3} & \leq & f_{abc} \\
\dfrac{2l_{ab} + l_{ad} + 2l_{bc} - u_{bd} + l_{cd} - 2}{3} & \leq & f_{abc} \\
\dfrac{2l_{ab} + 2l_{ac} - u_{ad} + l_{bd} + l_{cd} - 2}{3} & \leq & f_{abc}
\end{array}
$$

5.5.3 $2 \to 4$, 4 items

$$
\begin{array}{rcl}
0 & \leq & f_{abcd} \\
\\
f_{abcd} & \leq & u_{ab} \\
f_{abcd} & \leq & u_{ac} \\
f_{abcd} & \leq & u_{ad} \\
f_{abcd} & \leq & u_{bc} \\
f_{abcd} & \leq & u_{bd} \\
f_{abcd} & \leq & u_{cd} \\
f_{abcd} & \leq & 1 \\
\\
l_{ad} + l_{bc} - 1 & \leq & f_{abcd} \\
l_{ac} + l_{bd} - 1 & \leq & f_{abcd} \\
l_{ab} + l_{cd} - 1 & \leq & f_{abcd} \\
u_{ab} + l_{ac} + l_{ad} + l_{bc} + l_{bd} - u_{cd} - 1 & \leq & f_{abcd} \\
l_{ab} - u_{ac} + l_{ad} + l_{bc} - u_{bd} + l_{cd} - 1 & \leq & f_{abcd} \\
l_{ab} + l_{ac} - u_{ad} - u_{bc} + l_{bd} + l_{cd} - 1 & \leq & f_{abcd} \\
\\
f_{abcd} & \leq & 1 + u_{ab} + u_{ac} - l_{ad} + u_{bc} - l_{bd} - l_{cd} \\
f_{abcd} & \leq & 1 + u_{ab} - l_{ac} + u_{ad} - l_{bc} + u_{bd} - l_{cd} \\
f_{abcd} & \leq & 1 - l_{ab} + u_{ac} + u_{ad} - l_{bc} - l_{bd} + u_{cd} \\
f_{abcd} & \leq & 1 - l_{ab} - l_{ac} - l_{ad} + u_{bc} + u_{bd} + u_{cd} \\
\\
l_{ad} + l_{bd} + l_{cd} - 2 & \leq & f_{abcd} \\
l_{ac} + l_{bc} + l_{cd} - 2 & \leq & f_{abcd} \\
l_{ab} + l_{bc} + l_{bd} - 2 & \leq & f_{abcd} \\
l_{ab} + l_{ac} + l_{ad} - 2 & \leq & f_{abcd}
\end{array}
$$

6

Concise Representations

Until recently, most research in itemset mining concentrated on extracting *all* frequent sets as efficiently as possible. In this context, levelwise search [63] based on the monotonicity of frequency [2], sampling [78] and efficient structures for counting [43] have been studied. However, often the result of the mining operation itself is so large, that even enumerating all frequent sets is impossible. This blow-up happens for example when we set the frequency threshold too low, or when the data is heavily correlated. In the worst case, the number of itemsets can even be exponential in the number of items. Clearly, even the most efficient algorithms cannot enumerate such huge numbers of itemsets.

Recently *concise representations* [62] were proposed to address this problem. Instead of mining all frequent sets, a lossless representation is mined. A concise representation typically is a subset of all frequent sets, together with their frequency [1]. From this reduced set, the complete collection of frequent sets with their frequencies can be reconstructed. The representation then serves as a basis for further exploration of the data. Also in the domain of inductive databases [10] and in the context of concept lattices [76], concise representations are very useful.

In this chapter, an overview of different approaches to construct a concise representation is given. We discuss in-depth the following concise representations: free sets [9], closed sets [72, 73, 80, 75], disjunction-free sets [12, 56], generalized disjunction-free sets [58, 57], and non-derivable itemsets [15]. We extend the representation introduced in Chapter 4 and that is based on non-derivable itemsets by using additional assumptions.

Based on a careful analysis of the strategies used in the different representations, we present a unifying framework for the existing representations and

[1] As we will show later on, in some proposals there might be infrequent sets in the representation, as well as sets without their frequency. However, the ideas behind these representations are similar in spirit.

the ones we introduce. This framework gives an alternative representation based on the deduction rules presented in Chapter 4.

6.1 Definition

In this chapter, we will implicitly assume that we are working over a transaction database $\mathcal{D}$, and with a frequency threshold t. For example, we will use FSET to denote FSET$(\mathcal{D}, t)$.

A *Concise Representation of frequent sets* is, loosely speaking, a subset of FSET, completed with the frequencies, that allows for reconstructing FSET. Therefore, based on the representation, for each itemset I, we must be able to (a) decide whether I is frequent, and (b) if I is frequent, produce its frequency. Clearly, from this point of view, a concise representation needs to be defined with respect to a constructive procedure that performs extraction of frequencies from representations. *Mannila et al.* introduced in [62] the notion of a concise representation in a more general context. Our definition resembles theirs, but for reasons of simplicity, we only concentrate on representations that are exact, and for frequent itemsets.

Definition 17 *Let R be a function that takes a transaction database and a frequency threshold as input. R is a* representation of the frequent itemsets *if there exists a constructive procedure ϕ such that for each database $\mathcal{D}$, frequency threshold t, and itemset I, ϕ will, based on $R(\mathcal{D}, t)$,*

(a) decide whether I is t-frequent, and

(b) if I is t-frequent, compute the frequency of I.

ϕ will be called the evaluation function of R. □

In the definition we are deliberately vague about the image of R. In general, any set of structures is allowed as image. We will however only concentrate on structures that are based on collections of itemsets with associated frequency counts. For such representations, the term *concise* will refer to the space-efficiency of R; that is, R is called *more concise* than R' if for every database $\mathcal{D}$ and frequency threshold t, $R(\mathcal{D}, t)$ is smaller than $R'(\mathcal{D}, t)$. Notice however, that there necessarily is a trade-off between the space-efficiency of R and the complexity of the evaluation function of R. In general, the more concise R is, the more complex the evaluation function will be.

Example 15 *Suppose that in* FSET *we have three sets $I \subset J \subset K$, with $freq(I) = freq(K)$. Since $freq(I) \geq freq(J) \geq freq(K)$, there is no need to store the frequency of J. Based on this observation, we define R be as follows:*

$$R(\mathcal{D}, t) \;=\; \left\{ \begin{array}{l} (J, freq(J)) \mid J \in \text{FSET} \wedge \\ \quad \nexists I \subset J \subset K : freq(I) = freq(K) \end{array} \right\},$$

that is, R gives the set of all frequent itemsets, together with their frequencies, except for the itemsets I that have a sub- and super-set with the same frequency. R is a representation; a set J is in FSET *if and only if there is a superset of J in the representation. If J is frequent, then either J is in the representation together with its frequency, or there are sets I, K with $I \subset J \subset K$ and $freq(I, \mathcal{D}) = freq(K, \mathcal{D})$ in the representation.* □

Let f_I be the frequency of I in $\mathcal{D}$. Notice that it is *not* necessary that for each frequent set I,

$$R(\mathcal{D}, t) \models_{tight} freq(I) = f_I \;\; .$$

Indeed, as we will see, the procedure ϕ can be any computation and does not need to be based on logical implication.

6.2 Overview

We give an overview of the different existing concise representations. To illustrate the representations, we will use the database $\mathcal{D}$ over the items a, b, c, d given in Figure 6.1. All representations we discuss maintain the itemset-frequency semantics. We will thus deal a lot with sets of itemsets and sets of pairs $(I, freq(I, \mathcal{D}))$. For notational convenience we introduce Π_{Sets} and $\bowtie Freq$. Let S be a set of itemsets.

$$S \bowtie Freq \;=_{def}\; \{(I, freq(I, \mathcal{D})) \mid I \in S\} \;\; .$$

Let $\mathcal{R}$ be a set of itemset-support pairs.

$$\Pi_{Sets}\mathcal{R} \;=_{def}\; \{I \mid (I, f_I) \in \mathcal{R}\} \;\; .$$

The representations we consider in this chapter are n-tuples with each of the components either subsets of $2^{\mathcal{I}}$ or $2^{\mathcal{I}} \bowtie Freq$. Let now

$$\begin{array}{rcl} \mathcal{R}_1(\mathcal{D}, t) & = & (S_1, \ldots, S_k, T_{k+1}, \ldots, T_n) \;\text{ and} \\ \mathcal{R}_2(\mathcal{D}, t) & = & (S'_1, \ldots, S'_l, T'_{l+1}, \ldots, T_m) \end{array}$$

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

$t = 3/5$

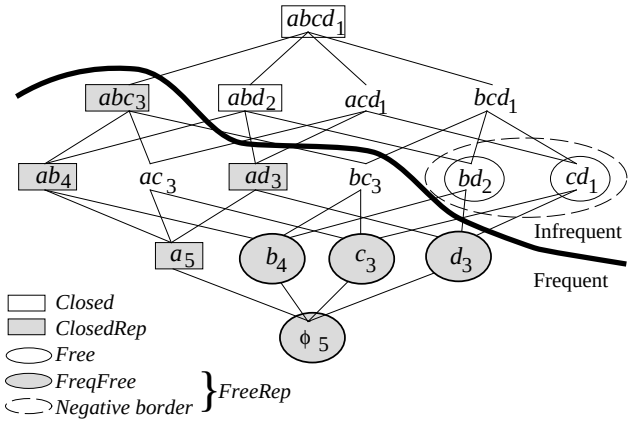


Figure 6.1: Free and Closed sets representations

be two representations, with then S_i's and S'_i's subsets of $2^{\mathcal{I}} \bowtie \mathit{Freq}$, and the T_i's and T'_i's subsets of $2^{\mathcal{I}}$. We assume that both $\Pi_{\mathit{Sets}} S_1$, ..., $\Pi_{\mathit{Sets}} S_k$, T_{k+1}, ..., T_n and $\Pi_{\mathit{Sets}} S'_1$, ..., $\Pi_{\mathit{Sets}} S'_l$, T'_{l+1}, ..., T'_m are collections of disjoint sets. In the representations we study, this will always be the case. We say that $\mathcal{R}_1$ is more concise than $\mathcal{R}_2$, denoted $\mathcal{R}_1 \sqsubseteq \mathcal{R}_2$, if, for all $\mathcal{D}$ and t,

$$\bigcup_{i=1...k} \Pi_{\mathit{Sets}} S_i \subseteq \bigcup_{j=1...l} \Pi_{\mathit{Sets}} S'_j \ ,$$

and

$$\bigcup_{i=k+1...n} T_i \subseteq \bigcup_{j=1...l} \Pi_{\mathit{Sets}} S'_j \cup \bigcup_{j=l+1...m} T'_j \ .$$

that is, every set in $\mathcal{R}_1$ must also be in $\mathcal{R}_2$, and every set in $\mathcal{R}_1$ that is stored together with its frequency, must also be in $\mathcal{R}_2$ with its frequency. If $\mathcal{R}_1 \sqsubseteq \mathcal{R}_2$ and $\mathcal{R}_2 \sqsubseteq \mathcal{R}_1$, then we say that $\mathcal{R}_1$ is equivalent with $\mathcal{R}_2$, and denote this with $\mathcal{R}_1 \equiv \mathcal{R}_2$.

6.2.1 Free Sets Representations

A *free set* [9] or *generator* [56] is an itemset such that its frequency does not equal the frequency of any of its subsets. In Figure 6.1, an example is given. All free sets are encircled with a solid line. For example, the set bc is not free, since its frequency equals the frequency of c. We will denote the collection of all free sets by *Free*.

The free sets representation is based on the following observation.

Lemma 15 *The frequency of I can be derived from*

$$\{J \subseteq I \mid J \in \mathit{Free}\} \bowtie \mathit{Freq}$$

as follows:

$$\mathit{freq}(I) \;=\; \min\{\mathit{freq}(J) \mid J \subseteq I, J \in Free\}$$

Proof
If I is free, then the result is straightforward. Otherwise, there exists a free subset K of I, such that $\mathit{freq}(I) = \mathit{freq}(K)$. Because of the monotonicity principle, it is also true that the frequency of I is smaller than or equal to the frequency of its subsets, and thus,

$$\begin{array}{c}\min\{\mathit{freq}(J) \mid J \subseteq I \wedge J \in \mathit{Free}\} \\ \leq \mathit{freq}(K) = \mathit{freq}(I) \leq \\ \min\{\mathit{freq}(J) \mid J \subseteq I \wedge J \in \mathit{Free}\}\end{array}$$

□

Another nice observation is that freeness of itemsets is anti-monotone; that is:

Lemma 16 *If an itemset is free, then its subsets are free as well.*

Proof
Suppose that $\mathit{freq}(I) = \mathit{freq}(I \cup \{a\})$, then we can conclude that every transaction that contains all items in I, must also contain the item a, otherwise the frequency of $I \cup \{a\}$ would be lower than the frequency if I. Let now J be a superset of $I \cup \{a\}$. Now, $\mathit{freq}(J - \{a\}) = \mathit{freq}(J)$, since $I \subseteq J - \{a\}$, and hence every transaction that contains $J - \{a\}$, also contains a. Therefore, if a set is not free, then neither is any of its supersets. □

We will now show how we can construct a representation based on the free sets. Let

$$\mathit{FreqFree} \;=_{def}\; (\mathit{Free} \cap \mathrm{FSET}) \;.$$

In spite of Lemma 15, $\mathit{FreqFree} \bowtie \mathit{Freq}$ is not a concise representation, since it does not allow to decide whether a set is frequent or not: consider the example database given in Figure 6.1; the frequent free sets are indicated with shaded circles. bc is not in *FreqFree* because it is not free, and bd is not

in *FreqFree* because it is not frequent. It is however impossible to differentiate between these two sets based on $FreqFree \bowtie Freq$ alone;

$$FreqFree \bowtie Freq \;=\; \left\{(\phi, 1), \left(b, \frac{4}{5}\right), \left(c, \frac{3}{5}\right), \left(d, \frac{3}{5}\right)\right\}$$

is symmetric in c and d. To overcome this problem, in [9, 56], it is proposed to add a part of the *cover* of *FreqFree* to the representation. The cover of a collection of itemsets $\mathcal{S}$ is

$$cover(\mathcal{S}) \;=_{def}\; \{I \mid I \notin \mathcal{S} \wedge \forall J \subset I : J \in \mathcal{S}\}\ .$$

In Figure 6.1, for example, $cover(FreqFree) = \{a, bc, bd, cd\}$. The free sets representation now is the pair of sets

$$FreeRep \;=_{def}\; \begin{array}{l}(FreqFree \bowtie Freq),\\ (cover(FreqFree) \cap Free)\end{array}$$

In Figure 6.1, the sets in $cover(FreqFree) \cap Free$ are indicated with dotted circles. It is proven in [9] that *FreeRep* is a representation. We will give a shorter and less involved proof than in [9], using the following lemma.

Lemma 17

$$(cover(FreqFree) \cap Free) = cover(\text{FSET})$$

Proof
Every itemset I in $cover$(FSET), is in *Free*, since every subset of I has frequency higher than the frequency threshold, and I has frequency below the threshold. Because of the anti-monotonicity of freeness, all subsets of I must be free as well. Thus, all subsets of I are in *FreqFree*. Therefore, I is in $(cover(FreqFree))$. It is also true that every itemset in $(cover(FreqFree) \cap Free)$ is in $cover$(FSET): every set in $(cover(FreqFree) \cap Free)$ is infrequent, otherwise it would be in *FreqFree*, and thus not in $cover(FreqFree)$. It is in the $cover$(FSET), because every subset is in *FreqFree* and thus frequent. Hence, $cover(FreqFree) \cap Free \;=\; cover(\text{FSET})$. □

The set $cover$(FSET) is called the *negative border*. Based on the negative border it is easy to decide whether a set is frequent or not; if it is below the negative border, then it is, when it is on or above the negative border, it is not. In Figure 6.1, the sets in the negative border are indicated with a dashed circle.

Theorem 19 *FreeRep is a representation.*

Proof
Based on Lemma 17, the proof is immediate. We can use the negative border to decide whether a set is frequent or not. If the set is frequent, we use Lemma 15 to derive its support from $\mathit{FreqFree} \bowtie \mathit{Freq}$. □

Notice that *FreeRep* contains infrequent sets. Therefore, it is possible that in some cases, *FreeRep* is larger than FSET $\bowtie$ *Freq*. We propose another representation, *FreeRep*$'$, that does not have this undesirable property.

$$\mathit{FreeRep}' =_{def} \begin{pmatrix} (\mathit{FreqFree} \bowtie \mathit{Freq}), \\ (\mathit{cover}(\mathit{FreqFree}) \cap \mathrm{FSET}) \end{pmatrix}$$

Theorem 20 *FreeRep*$'$ *is a representation and all sets in it are always frequent.*

Proof
The sets in $\mathit{cover}(\mathit{FreqFree})$ are either frequent, or free, but not both. Therefore,

$$\mathit{cover}(\mathit{FreqFree}) \cap \mathit{Free} = \mathit{cover}(\mathit{FreqFree}) - \mathit{cover}(\mathit{FreqFree}) \cap \mathrm{FSET} \ .$$

Thus, we can derive *FreeRep* from *FreeRep*$'$. □

6.2.2 Closed Sets Representation

In short, a *closed itemset* [72] is an itemset such that its frequency does not equal the frequency of any of its supersets. In [72], a closure operator $cl(.)$ on itemsets is introduced as follows: let I be a set of items. Recall that the *extention* of I, denoted $\mathit{ext}(I)$, is the set of all transactions that contain I. Given a set of transactions $\mathcal{T}$, the *intention* of $\mathcal{T}$, denoted $\mathit{int}(\mathcal{T})$, is the largest set of items that is contained in all transactions in $\mathcal{T}$. Notice that the extension is always defined relative to a transaction database. Based on these definitions, the closure of an itemset I is defined as

$$cl(I) =_{def} \mathit{int}(\mathit{ext}(I)) \ .$$

For example, in the database given in Figure 6.1, the extention of ab consists of the transactions with identifiers 1, 2, and 5. The intention of these transactions is the intersection of their sets of items: abc. Hence, $cl(ab) = abc$.

The closure of an itemset I is the largest itemset that has the same set of transactions as I. Therefore, $freq(cl(I), \mathcal{D}) = freq(I, \mathcal{D})$, and $cl(I)$ is the largest such set. An itemset I is closed if $cl(I) = I$. Notice that the notion of "the largest set" is well-defined. Suppose for the sake of contradiction that this largest set is not unique; that is, there are two sets $J_1, J_2 \supset I$ with $freq(J_i) = freq(I)$, for $i = 1, 2$. This equality implies that every transaction that contains I also contains $J_i - I$. Therefore, every transaction that contains I also contains $J_1 \cup J_2$, and thus $freq(J_1 \cup J_2) = freq(I)$, what contradicts the assumed maximality of J_1 and J_2. We will denote the set of all closed itemsets by *Closed*.

In Figure 6.1, the closed itemsets are indicated with rectangles. For example, the set ab is closed since none of its supersets has a frequency of $\frac{4}{5}$. b is not closed, since ab has the same frequency. The closure of b is ab. Typically, in realistic datasets, the number of closed itemsets is much smaller than the total number of itemsets.

If we look at the definition of free sets as the minimal subsets sharing the same frequency, and closed sets as the maximal such supersets, it is not surprising that there is a strong connection between the two.

Lemma 18

$$Closed = \{cl(I) \mid I \in Free\}$$

Proof
Let I be a closed set. Let K be one of the minimal elements (with respect to the inclusion ordering) in set

$$\{J \subseteq I \mid freq(J) = freq(I)\} \ .$$

It is straightforward that K is free and that $cl(K) = I$. □
This connection is the reason why free sets are also called generators; they generate the closed sets via the $cl(.)$ -operator.

Lemma 19 *The frequency of I can be derived from*

$$(\text{minimal}\{J \in Closed \mid I \subseteq J\}) \bowtie Freq$$

as follows:

$$freq(I) = \max\{freq(J) \mid J \in \text{minimal}\{J \in Closed \mid I \subseteq J\}\} \ .$$

$(\text{minimal}(\mathcal{S}) := \{J \in \mathcal{S} \mid \nexists J' \in \mathcal{S} : J' \subset J\})$

Proof
Via similar reasoning as in the proof of Lemma 15. □

The positive border of the frequent sets consists of all sets that are frequent, but that do not have frequent supersets. In Figure 6.1, the positive border is $\{ad, abc\}$. Clearly, the positive border characterizes FSET; a set is frequent if it has a superset in the positive border. Furthermore, all elements in the positive border must be closed: suppose that a frequent set I is not closed, then $cl(I)$ is a superset of I that is frequent, and thus I is not in the positive border. This statement directly implies that the frequent closed itemsets allow for deciding whether a set is frequent or not. Thus, unlike in the free-set representation, we will not need to add the negative border. Let now

$$\mathit{ClosedRep} \ =_{def} \ (\mathrm{FSET} \cap \mathit{Closed}) \bowtie \mathit{Freq} \ .$$

In Figure 6.1, *ClosedRep* is denoted with shaded rectangles.

Theorem 21 *ClosedRep is a representation.*
Proof
We can decide whether or not a set is frequent using the positive border of the frequent sets, and if a set is frequent, then we can derive its actual frequency using Lemma 19. □

In [72], it is also shown that *Closed* forms a lattice. This lattice is known as the *concept lattice* [32]. In [80], the ChARM algorithm exploits this lattice structure. Other algorithms, such as A-Close [72] and CLOSET [75] rely mainly on Lemma 18 to find the closed sets.

6.2.3 Disjunction-Free Sets Representations

A *disjunctive rule* [12] is an expression $X \rightarrow (a \vee b)$, with X an itemset, and a and b items. The disjunctive rule $X \rightarrow (a \vee b)$ is said to *hold in a transaction database* $\mathcal{D}$ if every transaction that contains X also contains either a or b.

Lemma 20 *$X \rightarrow (a \vee b)$ holds if and only if*

$$\mathit{freq}(X \cup \{a, b\}) = \mathit{freq}(X \cup \{a\}) + \mathit{freq}(X \cup \{b\}) - \mathit{freq}(X) \ .$$

Furthermore, if $a \neq b$, then

$$\mathit{freq}(X \cup \{a, b\}) \geq \mathit{freq}(X \cup \{a\}) + \mathit{freq}(X \cup \{b\}) - \mathit{freq}(X) \ .$$

TID	Items
1	a, b, c, d, e
2	a, b, d, e
3	a, b, d, e
4	b, c, d, e
5	b, c, d, e
6	a, b, e
7	a, c, d
8	a, c, e
9	b, c, d
10	b, c, e
11	c, d, e
12	b, c
13	b, d
14	c, d
15	d, e
16	b

$t = 3/16$

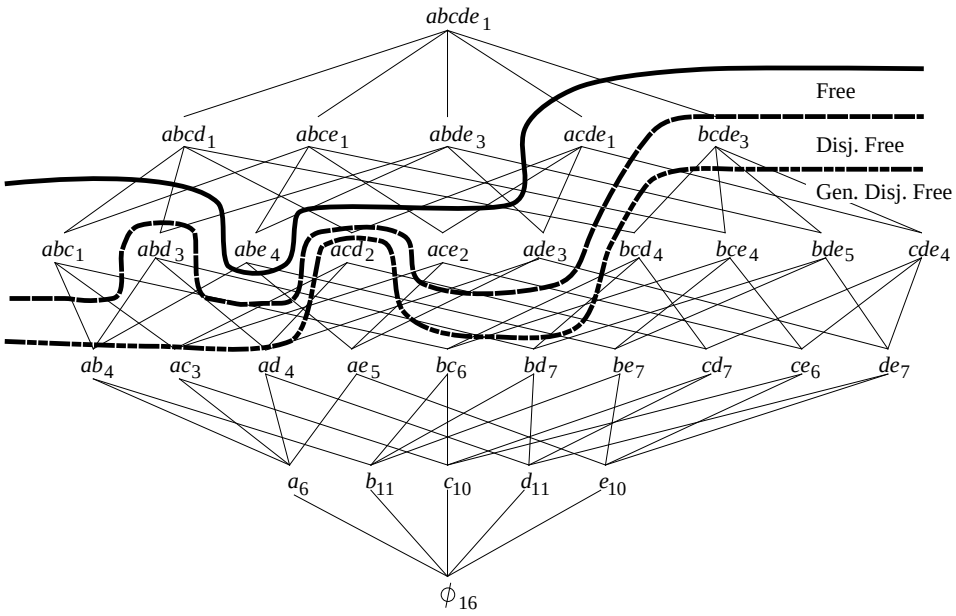


Figure 6.2: Free, disjunction-free and generalized disjunction-free sets

Proof
The disjunctive rule $X \rightarrow (A \vee B)$ holds if and only if $\mathcal{F}_X^{X\cup\{a,b\}} = 0$. Indeed; $\mathcal{F}_X^{X\cup\{a,b\}} = 0$ if and only if there are no transactions that contain X and do not contain a nor b. It is also true that

$$
\begin{aligned}
freq(X) &= \mathcal{F}_X^{X\cup\{a,b\}} + \mathcal{F}_{X\cup\{a\}}^{X\cup\{a,b\}} + \mathcal{F}_{X\cup\{b\}}^{X\cup\{a,b\}} + \mathcal{F}_{X\cup\{a,b\}}^{X\cup\{a,b\}} \ , \\
freq(X\cup\{A\}) &= \mathcal{F}_{X\cup\{a\}}^{X\cup\{a,b\}} + \mathcal{F}_{X\cup\{a,b\}}^{X\cup\{a,b\}} \ , \\
freq(X\cup\{B\}) &= \mathcal{F}_{X\cup\{b\}}^{X\cup\{a,b\}} + \mathcal{F}_{X\cup\{a,b\}}^{X\cup\{a,b\}} \ , \\
freq(X\cup\{A,B\}) &= \mathcal{F}_{X\cup\{a,b\}}^{X\cup\{a,b\}} \ .
\end{aligned}
$$

From these equalities, we easily derive the lemma. □

Lemma 20 will be used in both directions. On one hand we will use it to check whether a disjunctive rule $X \rightarrow (a \vee b)$ holds, and on the other hand we can use it to derive the frequency of the itemset $X \cup \{a, b\}$ in the case that we know the rule holds. Notice also that we can use the second part of the lemma to derive lower bounds on the frequency of itemsets.

A set I will be called *disjunction-free* if there do not exist items $a, b \in I$ such that $(I - \{a, b\}) \rightarrow (a \vee b)$ holds in the transaction database [2]. We denote the set of all disjunction-free itemsets by *DFree*. Notice that a set I is a free set if and only if there exists an item $a \in I$ such that the rule $I \rightarrow (a \vee a)$ holds. Therefore, every disjunction-free set is also free. The opposite direction does not hold, as can be seen in Figure 6.1; the set *bd* is free, but not disjunction-free (rule $\phi \rightarrow (b \vee d)$ holds).

In Figure 6.2, a database is given, together with the disjunction-free sets. The disjunction-free sets are the ones below the second line. *acde* is not disjunction-free since for example the rule $ac \rightarrow (d \vee e)$ holds. Indeed, the only transactions that contain *ac* are the ones with identifiers 1, 7, and 8. 1 contains both d and e, 7 contains d, and 8 contains e.

Lemma 21 *Disjunction-freeness is anti-monotone, that is: if an itemset I is disjunction-free, then all subsets are disjunction-free as well.*

Proof
Let $I \subset J$. Suppose $(I - \{a, b\}) \rightarrow (a \vee b)$ holds. Then also $(J - \{a, b\}) \rightarrow (a \vee b)$ holds, since every transaction that contains $J - \{a, b\}$ also contains $I - \{a, b\}$. We can thus conclude that if I is not disjunction free, then neither

[2] Notice that we do *not* require that a and b are different items.

is J. □

Notice that from the proof of Lemma 21, we can derive that if we know a disjunctive rule for I that holds, we automatically know disjunctive rules for all supersets of I. For example, consider the database in Figure 6.2. Suppose that we know the following frequencies: $freq(a) = \frac{6}{16}$, $freq(ac) = \frac{3}{16}$, $freq(ae) = \frac{5}{16}$, and $freq(ace) = \frac{2}{16}$. Using Lemma 20, we can derive that the rule $a \rightarrow (c \vee e)$ holds. Using a similar reasoning as in the proof of Lemma 21, we can derive that hence also the rule $ab \rightarrow (c \vee e)$ holds, and thus that $freq(abce) = freq(abc) + freq(abe) - freq(ab)$. The disjunction-free sets representation is based on this observation.

The frequent disjunction-free sets, which we will denote by *FreqDFree*, do not form a representation on their own. Let *DFreeRep* be the following collection

$$DFreeRep \ =_{def} \ (FreqDFree \cup cover(FreqDFree)) \bowtie Freq \ .$$

In [12], the following theorem is proved.

Theorem 22 *DFreeRep is a representation.*

Proof

Because we have added the cover to the representation, for each set J that is not in the representation, there exists a disjunction-free subset I in the representation and a subset $I \cup \{a\}$ in the border. The set $I \cup \{a\}$ is either infrequent or not disjunction-free. If it is infrequent, then J must be infrequent as well. If it is not disjunction-free, then neither is J. Using the reasoning given above, we can derive which disjunctive rule holds for J, based on the rule that holds for $I \cup \{a\}$. Thus, based on the frequencies of the subsets of J, we can derive the frequency of J. Now, the theorem can easily be proved by induction on the size of J. □

In Figure 6.2,

$$DFreeRep \ = \ \left(\left\{ \begin{array}{l} \phi, a, b, c, d, e, \\ ab, ac, ad, ae, bc, \\ bd, be, cd, ce, de, \\ abd, acd, bcd, bce, \\ bde, cde, bcde \end{array} \right\} \cup \left\{ \begin{array}{l} abc, abe, \\ ace, ade \end{array} \right\} \right) \bowtie Freq$$

Suppose we want to know whether *abcde* is frequent. We first check whether there are infrequent subsets of *abcde* in the cover. This is the case, for

example, *ace*. Hence, based on the representation we can conclude that *abcde* is infrequent. If we want to know whether *abde* is frequent, we perform the same check. None of the subsets of *abde* in the cover is infrequent. Because for example *ade* is in the cover, and is frequent, we know that it is not disjunction-free. We can test which disjunctive rule holds with the frequencies in *DFreeRep*. In this way we find that

$$freq(ade) = freq(ad) + freq(ae) - freq(a) \ .$$

Therefore, the rule $a \rightarrow (d \vee e)$ must hold. But then also rule $ab \rightarrow (d \vee e)$ holds, and thus

$$freq(abde) = freq(abd) + freq(abe) - freq(ab) = \frac{3}{16} \ .$$

As is observed in [56], it is not necessary to store all sets in the border. Instead, it is argued that it suffices to only store the *generators* in the border of *FreqDFree*. Furthermore, for the infrequent sets we do not need to store the actual frequency. In this way we get the following representation *DFreeGenRep*, which we will call the *disjunction-free generators representation*:

$$DFreeGenRep \ =_{def} \ \left(\begin{array}{l} FreqDFree \bowtie Freq \ , \\ cover(FreqDFree) \cap Free \cap \mathrm{FSET} \bowtie Freq \ , \\ cover(FreqDFree) \cap Free \cap Infrequent \end{array} \right)$$

Theorem 23 *DFreeGenRep is a representation of the frequent sets.*

Proof

If a set J is in *DFreeRep*, but not in *DFreeGenRep*, then, by definition, J is not free. All subsets of J however are stored in *DFreeGenRep*, together with their frequencies. We can now use Lemma 15 to derive the frequency of J. □

In Figure 6.2, *DFreeGenRep* is

$$\left(\left\{ \begin{array}{l} \phi, a, b, c, d, e, \\ ab, ac, ad, ae, bc, \\ bd, be, cd, ce, de, \\ abd, acd, bcd, bce, \\ bde, cde, bcde \end{array} \right\} \bowtie Freq, \left\{ \ ade \ \right\} \bowtie Freq, \left\{ \ abc, ace \ \right\} \right)$$

Because *abe* is in the cover of the disjunction-free sets, but not in the representation, we know that *abe* is not free. Therefore, the frequency of *abe* is the minimum of the frequencies of its subsets, which is $\frac{4}{16}$.

6.2.4 Generalized Disjunction-Free Representation

In [58, 57], generalized disjunction-free generators are introduced as a natural extension of disjunction-free generators. The construction of the representation based on generalized disjunction-free itemsets is very similar to the construction of the disjunction-free representations.

A *generalized disjunctive rule* is an expression $X \rightarrow \bigvee Y$, with X and Y disjunctive itemsets. A rule $X \rightarrow \bigvee Y$ is said to *hold in a transaction database* if every transaction that contains X also contains at least one of the items in Y. An itemset I is *generalized disjunction-free* if there does not exist a subset X of I such that $(I - X) \rightarrow \bigvee X$ holds. We denote the set of all generalized disjunction-free sets by *GDFree*.

For example, in Figure 6.2, *bcde* is not generalized disjunction-free because the rule $c \rightarrow (b \vee d \vee e)$ holds.

Lemma 22 *Generalized disjunction-freeness is anti-monotone; that is, if a set is generalized disjunction-free, than its subsets are as well.*

Proof
The proof of this lemma is very similar to the proof of Lemma 21. Suppose an itemset I is generalized disjunction-free based on rule $(I - X) \rightarrow \bigvee X$, then every superset J of I is generalized disjunction-free because rule $(J - X) \rightarrow \bigvee X$ must hold as well. □

In Figure 6.2, *bcde* is not generalized disjunction-free because of rule $c \rightarrow (b \vee d \vee e)$. Therefore, also *abcde* is not generalized disjunction-free, with rule $ac \rightarrow (b \vee d \vee e)$.

Notice that the situation here is similar as with the disjunction-free sets. If we know the rule for a non generalized disjunction-free set, then we also know rules for all supersets. We still need to show how, based on a generalized disjunctive rule that holds, and the frequencies of subsets, we can derive the frequency of a non-generalized disjunctive-free itemset.

Lemma 23 *Let I be an itemset, X a subset of I. The rule $(I - X) \rightarrow \bigvee X$ holds if and only if*

$$freq(I) = \sum_{I - X \subseteq J \subset I} (-1)^{|I-J|+1} freq(J) \ .$$

Proof
For a direct proof of this result, we refer the reader to [57]. This result also follows from

$$\sigma(I-X,I) = \sum_{I-X\subseteq J\subset I} (-1)^{|I-J|+1} freq(J) \ ,$$

and

$$|freq(I) - \sigma(I-X,I)| = \mathcal{F}^I_{I-X} \ .$$

Furthermore, $(I-X) \rightarrow \bigvee X$ holds if there are no transactions that contain $I-X$, but none of X, hence whenever $\mathcal{F}^I_X = 0$. □

This lemma allows us to check, based on the frequencies of an itemset I and its subsets, whether there is a rule based on I that holds. If there is such a rule, then we know by Lemma 22 that, for all supersets of I, there are rules that hold. Using Lemma 23 in the only if direction, we can then derive the frequencies of these supersets, based on the frequency of the subsets. For example, in Figure 6.2, we find that the rule $c \rightarrow (b \vee d \vee e)$ holds because

$$\begin{aligned} freq(bcde) &= freq(bcd) + freq(bce) + freq(cde) \\ &\quad -freq(bc) - freq(cd) - freq(ce) + freq(c) \\ &= \tfrac{3}{16} \ . \end{aligned}$$

Because rule $c \rightarrow (b \vee d \vee e)$ holds, also rule $ac \rightarrow (b \vee d \vee e)$ holds, and hence,

$$\begin{aligned} freq(abcde) &= freq(abcd) + freq(abce) + freq(acde) \\ &\quad -freq(abc) - freq(acd) - freq(ace) + freq(ac) \\ &= \tfrac{1}{16} \ . \end{aligned}$$

In a representation, we will not store the supersets of a non-generalized disjunction-free set. The sets in the border of the frequent generalized disjunction-free sets we will have to store however; for those sets, we have no indication of which rule holds. We will denote the frequent generalized disjunction-free sets by *FreqGenDFree*. Let now *GDFreeRep* be

$$(FreqGenDFree \cup cover(FreqGenDFree)) \bowtie Freq \ .$$

The next theorem is now immediate.

Theorem 24 *GDFreeRep is a representation.*

We can again apply the same technique as with the disjunction-free representation. Instead of storing the whole cover, we only store the free sets in the cover, and we only store the frequencies of the frequent sets.

Let *GDFreeGenRep* be the following collection:

$$\left(\begin{array}{l} \mathit{FreqGenDFree} \bowtie \mathit{Freq} \ , \\ \mathit{cover}(\mathit{FreqGenDFree}) \cap \mathit{Free} \cap \mathrm{FSET} \bowtie \mathit{Freq} \ , \\ \mathit{cover}(\mathit{FreqGenDFree}) \cap \mathit{Free} \cap \mathit{Infrequent} \end{array} \right)$$

Theorem 25 *GDFreeGenRep is a representation of the frequent sets.*
Proof
The proof is similar to the proof of Theorem 23. □

In Figure 6.2, *GDFreeGenRep* is the following collection:

$$\left(\left\{ \begin{array}{l} \phi, a, b, c, d, e, \\ ab, ac, ad, ae, bc, \\ bd, be, cd, ce, de, \\ acd, bcd, bce, bde, \\ cde \end{array} \right\} \bowtie \mathit{Freq}, \left\{ \ ade, bcde \ \right\} \bowtie \mathit{Freq}, \left\{ \ abc, ace \ \right\} \right)$$

6.2.5 Non-Derivable Itemsets Representation

The non-derivable itemsets were already introduced in Chapter 4. Here we only repeat the most important properties.

Properties Let

$$\sigma(J, I) \ = \ \sum_{J \subseteq I' \subset I} (-1)^{|I - I'|+1} \mathit{freq}(I', \mathcal{D})$$

$$\mathcal{R}_I(J) \ \equiv \ \left\{ \begin{array}{ll} \mathit{freq}(I) \leq \sigma(J, I) & |I - J| \text{ odd} \\ \mathit{freq}(I) \geq \sigma(J, I) & |I - J| \text{ even} \end{array} \right.$$

A set I is called a derivable itemset in a database $\mathcal{D}$ if and only if

$$\{\mathit{freq}(I) = \mathit{freq}(I, \mathcal{D}) \mid I \subset J\} \models_{\mathit{tight}} \mathit{freq}(I) = \mathit{freq}(I, \mathcal{D}) \ .$$

- Let $\mathcal{C} = \{\mathit{freq}(I) = \mathit{freq}(I, \mathcal{D}) \mid I \subset J\}$.

$$\mathcal{C} \models_{\mathit{tight}} \mathit{freq}(I) \in \ \begin{array}{l} [\ \max \{\sigma(J, I) \mid |I - J| \text{ even}\} \ , \\ \min \{\sigma(J, I) \mid |I - J| \text{ odd}\} \ ] \end{array}$$

- The error on $\mathcal{R}_I(J)$ is $\mathcal{F}^I_J$; that is,

$$|\sigma(J, I) - \mathit{freq}(I, \mathcal{D})| \ = \ \mathcal{F}^I_J \ .$$

- Derivability is monotone, that is, if $I \subseteq J$, and I is a derivable itemset, then also is J.

- If $\sigma(J, I) = \mathit{freq}(I, \mathcal{D})$, then also

$$\sigma(J, I \cup \{i\}) = \sigma(J \cup \{i\}, I \cup \{i\}) = \mathit{freq}(I \cup \{i\}) \ .$$

Let J be an itemset. We denote the bounds which we can calculate on the frequency of I with the deduction rules $\mathcal{R}_I(J)$, by $LB(I)$ and $UB(I)$. That is,

$$\begin{array}{rcl} LB(I) & =_{def} & \max\{\sigma(J, I) \mid |I - J| \text{ even}\} \ , \text{ and} \\ UB(I) & =_{def} & \min\{\sigma(J, I) \mid |I - J| \text{ odd}\} \ . \end{array}$$

Notice that in a concise representation, we do not need to store the supports of the derivable itemsets, since their support can be derived from the supports of its subsets.

The NDI-based representation defined in Chapter 4 is:

$$\mathit{NDIRep} \ =_{def} \ (\mathit{NDI} \cap \text{FSET}) \bowtie \mathit{Freq} \ .$$

We already showed in Theorem 18 that *NDIRep* is a representation.

Notice that, unlike the other representations, *NDIRep* is based on *logical implication*. That is, for every frequent set that is not in the representation, the frequency is uniquely determined by the frequencies of the sets in *NDIRep*. We do not need any additional conventions; the fact that an itemset I in the representation has frequency $\mathit{freq}(I, \mathcal{D})$ is *logically implied* by *NDIRep*. For the other representations this is not the case. In the other representations we rely heavily on a couple of assumptions. For example, the free sets representation relies on the fact that if a set is frequent and not in the representation, then its frequency equals the frequency of one of its subsets. With this assumption, the frequency is uniquely determined, but the frequencies of the itemsets in the representation alone do not necessarily logically imply the frequency of this set. In the next section we show how we can improve the NDI-based representation by adding similar assumptions.

6.3 Extending the NDI-Representation

In this section we will extend the NDI-representations in two ways. First, we restrict the depth of the rules. Second, we add assumptions to the NDI-based representations.

6.3.1 Rules of Limited Depth

Recall form the last section that the rule $\mathcal{R}_I(J)$ with $J \subseteq I$ calculates a lower bound on the support of I if and only if $|I - J|$ is even. In the case $|I - J|$ is odd, $\mathcal{R}_I(J)$ calculates an upper bound. Notice also that the complexity of the rules becomes higher as $|I - J|$ increases. We will call rules $\mathcal{R}_I(J)$ with $|I - J| = k$, *rules of depth* k. For example, the monotonicity of frequency is stated by the rules of depth 1. Because of the high complexity of the rules of high depth, we will often evaluate only the rules of limited depth. By restricting ourselves to only these rules, the resulting bounds are no longer tight, but they are more efficiently computable. We will denote the lower and upper bound that we can compute on the frequency of I with rules up to depth k by respectively $LB_k(I)$ and $UB_k(I)$. That is,

$$\begin{array}{lll} LB_k(I) & =_{def} & \max\{\sigma(J,I) \mid J \subseteq I, |I-J| \leq k, \text{even}\} \ , \\ UB_k(I) & =_{def} & \min\{\sigma(J,I) \mid J \subseteq I, |I-J| \leq k, \text{odd}\} \ . \end{array}$$

If $k = |I|$, that is, we are evaluating all rules, we will omit the subscript k. Clearly,

$$\begin{array}{l} LB_0(I) = LB_1(I) \leq LB_2(I) = LB_3(I) \leq \ldots \leq LB(I) \leq \mathit{freq}(I,\mathcal{D}) \ , \\ \mathit{freq}(I,\mathcal{D}) \leq UB(I) \leq UB_{|I|}(I) \leq \ldots \leq UB_2(I) = UB_1(I) \ , \end{array}$$

and,

$$[LB, UB] \ \subseteq \ [LB_{|I|-1}, UB_{|I|-1}] \ \subseteq \ \ldots \ \subseteq \ [LB_1, UB_1] \ \subseteq \ [LB_0, UB_0] \ .$$

Since there are no rules for upper bounds of depth 0, we let $UB_0(I) = \infty$.

Example 16 *Consider the database that has been given in Figure 6.2. For bcd, we can calculate the following bounds:*

$$\begin{array}{llcll} (0) & \mathit{freq}(bcd) & \geq & \sigma(bcd,bcd) & = & 0 \\ \\ (1) & & \leq & \sigma(bc,bcd) & = & f_{bc} = \frac{6}{16} \\ & & \leq & \sigma(bd,bcd) & = & f_{bd} = \frac{7}{16} \\ & & \leq & \sigma(cd,bcd) & = & f_{cd} = \frac{7}{16} \\ \\ (2) & & \geq & \sigma(b,bcd) & = & f_{bc} + f_{bd} - f_b = \frac{2}{16} \\ & & \geq & \sigma(c,bcd) & = & f_{bc} + f_{cd} - f_c = \frac{3}{16} \\ & & \geq & \sigma(d,bcd) & = & f_{bd} + f_{cd} - f_d = \frac{3}{16} \\ \\ (3) & & \leq & \sigma(\phi,bcd) & = & f_{bc} + f_{bd} + f_{cd} - f_b - f_c - f_d + f_\phi = \frac{4}{16} \end{array}$$

Hence, $LB_0(bcd) = 0$, $LB_2(bcd) = \frac{2}{16}$, $UB_1(bcd) = \frac{6}{16}$, $UB_3(bcd) = \frac{4}{16}$. *Thus, for depth going from* 0 *to* 3*, we get respectively the following intervals:*

$$[0, \infty] \supseteq \left[0, \frac{6}{16}\right] \supseteq \left[\frac{2}{16}, \frac{6}{16}\right] \supseteq \left[\frac{2}{16}, \frac{4}{16}\right] .$$

□

From the theory developed for non-derivable itemsets, we easily derive the following properties of the LB_k and UB_k notations.

Theorem 26 *Let* $f_I = freq(I, \mathcal{D})$*, for all* $I \subseteq \mathcal{I}$.

- $\{freq(I) = f_I \mid I \subset J\} \models_{tight} freq(I) \in [LB(I), UB(I)]$,
- $\min\{|LB_k(I) - freq(I)|, |UB_k(I) - freq(I)|\} = \min\{\mathcal{F}_J^I \mid |I - J| \leq k\}$,
- *Let* $I \subset J$*. If* $freq(I) = LB_{2k}(I)$*, then also* $freq(J) = LB_{2k}(J)$*, and* $freq(J) = UB_{2k+1}(J)$.
  If $freq(I) = UB_{2k-1}(I)$*, then also* $freq(J) = UB_{2k-1}(J)$*, and* $freq(J) = LB_{2k}(J)$.

Proof
These properties are the respective counterparts of Theorem 17, the fact that $|\sigma(J, I) - freq(I)| = \mathcal{F}_J^I$, and Corollary 5. □

6.3.2 NDI-representations of Limited Depth

We first extend the notion of NDI-based representation to also include representations that are only based on rules up to depth k.

$$NDI_k =_{def} \{I \mid LB_k(I) \neq UB_k(I)\} ,$$
$$NDIRep_k =_{def} (NDI_k \cap \text{FSET}) \bowtie Freq .$$

Based on Theorem 26 (the third part), we obtain the following theorem.

Theorem 27 $NDIRep_k$ *is a concise representation of the frequent itemsets.*

It is clear that there is a hierarchy between these different representations:

Lemma 24

$$NDIRep_1 \sqsupseteq NDIRep_2 \sqsupseteq \ldots \sqsupseteq NDIRep .$$

6.3.3 Adding Assumptions to NDI-Representations

As we mentioned earlier, most representations rely not only on logical implication, but also on some additional assumptions. We will now show how we can incorporate additional assumptions to make the NDI-based representations more concise.

Frequency Equals Lower Bound Suppose that we have a frequent itemset for which $freq(I) = LB_k(I)$, but not $freq(I) = UB_k(I)$. In that case we can leave I out of the representation. If we want to restore the set of frequent itemsets later on we will be able to recognize a set I that was left out because of the equality $freq(I) = LB_k(I)$ as follows. When we calculate the bounds on I, we see that I is frequent (the lower bound is above the frequency threshold), and not in NDI_k. Therefore, I must have been left out of the representation because $freq(I) = LB_k(I)$. We define the resulting representations as follows:

$$\begin{aligned} lbNDI_k &=_{def} \{I \mid LB_k(I) \neq freq(I, \mathcal{D})\} \ , \\ lbNDIRep_k &=_{def} (lbNDI_k \cap \text{FSET}) \bowtie Freq \ . \end{aligned}$$

lbNDI and *lbNDIRep* are defined in a similar way, with LB and UB. Thus, starting from $lbNDIRep_k$, we restore FSET as follows: we work bottom-up, that is, we start with the smallest sets. As such, for each set we need to consider, we know for all subsets whether they are frequent of not, and if they are frequent, then we know the exact frequency. We summarize the handling of a set I:

(1) I has an infrequent subset: I is infrequent due to monotonicity.

(2) I does not have an infrequent subset.

   (a) I is in $lbNDIRep_k$: I is frequent, and we get the frequency directly from $lbNDIRep_k$.

   (b) I is not in $lbNDIRep_k$: Since I has no infrequent subsets, and we are working bottom-up, we know the frequency of all subsets of I. Hence we can calculate the bound $LB_k(I)$. t denotes the threshold for the frequency.

      (i) $t \leq LB_k(I)$: I must be frequent, and hence $freq(I) = LB_k(I)$, because otherwise the set I would have been in $lbNDIRep_k$.

      (ii) $LB_k(I) < t$: Since I is not in $lbNDIRep_k$, I is either infrequent, or $freq(I) = LB_k(I) < t$, in which case I is infrequent as well. Therefore we can conclude that I is infrequent.

Theorem 28 *$lbNDIRep_k$ is a concise representation of the frequent itemsets, and for each k, $lbNDIRep_k \sqsubseteq NDIRep_k$. For all $k \geq 0$, $lbNDIRep_{k+1} \sqsubseteq lbNDIRep_k$*

Proof
It is clear that $lbNDIRep_k$ is a representation.

It is straightforward that $NDIRep_k \sqsubseteq lbNDIRep_k$; whenever $LB_k(I) = UB_k(I)$, also $freq(I) = LB_k(I)$.

$lbNDIRep_{k+1} \sqsubseteq lbNDIRep_k$ follows directly from $lbNDI_{k+1} \sqsubseteq lbNDI_k$. □

Notice that it is not always true that $NDIRep \sqsubseteq lbNDIRep_k$. As a concrete example consider the following database:

TID	Items
1	a
2	b
3	a, b

$$freq(\{\}) = 1$$
$$freq(a) = \tfrac{2}{3}$$
$$freq(b) = \tfrac{2}{3}$$
$$freq(ab) = \tfrac{1}{3}$$

In this database,

$$lbNDIRep_2 = \left\{ (\phi, 1), \left(a, \frac{2}{3}\right), \left(b, \frac{2}{3}\right) \right\} \ ,$$

and

$$NDIRep = \left\{ (\phi, 1), \left(a, \frac{2}{3}\right), \left(b, \frac{2}{3}\right), \left(ab, \frac{1}{3}\right) \right\} \ .$$

Frequency Equals Upper Bound One might now wonder whether we can do something similar for upper bounds. The answer however is negative; for the upper bounds the construction is far more involved. Let

$$ubNDI_k \ =_{def} \ \{I \mid UB_k(I) \neq freq(I)\} \ .$$

If we only store $(ubNDI_k \cap \text{FSET}) \bowtie Freq$, we will be unable to distinguish between a frequent itemset I with $LB_k(I) < t \leq freq(I) = UB_k(I)$, and an infrequent itemset J with $LB_k(J) \leq freq(J) < t \leq UB_k(J)$, where t is the frequency threshold. Using a similar schema as for $lbNDIRep_k$, for a set I, we would get the following (flawed) decision procedure.

(1) I has an infrequent subset: I is infrequent due to monotonicity.

(2) I does not have an infrequent subset

(a) I is in $ubNDI_k \cap \text{FSET}$: I is frequent, and we get the frequency directly from $(ubNDI_k \cap \text{FSET}) \bowtie \mathit{Freq}$.

(b) I is not in $ubNDI_k \cap \text{FSET}$: Since I has no infrequent subsets, and we are working bottom-up, we know the frequency of all subsets of I. Hence we can calculate the bounds $LB_k(I)$ and $UB_k(I)$. t denotes the threshold for the frequency.

(i) $t \leq LB_k(I)$: I is certainly frequent. Therefore, $\mathit{freq}(I) = UB_k(I)$, because otherwise I would be in $ubNDI_k \cap \text{FSET}$.

(ii) $UB_k(I) < t$: I must be infrequent.

(iii) $LB_k(I) < t \leq UB_k(I)$: Here we have a problem. There can be two reasons for I not being in $ubNDI_k \cap \text{FSET}$. Either I is infrequent, or $\mathit{freq}(I) = UB_k(I)$ and hence I is frequent. Based on $LB_k(I) < t \leq UB_k(I)$, these two possibilities are indistinguishable. There is however one more thing we can do:

(α) There is a subset J of I such that $UB_k(J) = \mathit{freq}(J)$. Therefore, I is frequent because also $\mathit{freq}(I) = UB_k(I) \geq t$.

(β) No subset J of I has $UB_k(J) = \mathit{freq}(J)$. Thus, every subset of I is in $ubNDI_k \cap \text{FSET}$, and I itself is not. The sets for which we cannot solve the problem are thus exactly the ones in $\mathit{cover}(ubNDI_k \cap \text{FSET})$.

Notice that we only have problems with sets in the cover of $ubNDI_k \cap \text{FSET}$ (case 2.b.iii.β). To resolve this issue we will add part of the cover of $ubNDI_k \cap \text{FSET}$ to the representation. It is clear that we do not need to add itemsets I for which $t \leq LB_k(I)$ or $UB_k(I) < t$, since these are handled in respectively case 2.b.i and 2.b.ii. We can now choose to either store the infrequent sets J in the cover that have $UB_k(J) \geq t$, or to store the frequent sets I with $LB_k(I) \leq t$. These two options give the following two representations.

$$ubNDIRep_k =_{def} \left((ubNDI_k \cap \text{FSET}) \bowtie \mathit{Freq} \ , \left(\begin{array}{l} \mathit{cover}(ubNDI_k \cap \text{FSET}) \\ \cap \mathit{Infrequent} \\ \cap \{I \mid UB_k(I) \geq t\} \end{array} \right) \right)$$

$$ubNDIRep'_k =_{def} \left((ubNDI_k \cap \text{FSET}) \bowtie Freq \ , \left(\begin{array}{l} cover(ubNDI_k \cap \text{FSET}) \\ \cap \text{FSET} \\ \cap \{I \mid LB_k(I) < t\} \end{array} \right) \right)$$

For $ubNDIRep_k$, case 2.b.iii.β becomes:

2.b.iii.β Let C denote $cover(ubNDI_k \cap \text{FSET}) \cap Infrequent \cap \{I \mid UB_k(I) \geq t\}$.

- β_1 I is in C: I is infrequent.
- β_2 I is not in C: I is frequent, and thus $freq(I) = UB_k(I)$, because otherwise I would have been in $ubNDI_k \cap \text{FSET}$.

For $ubNDIRep'_k$, we get:

2.b.iii.β' Let $\overline{C}$ denote $cover(ubNDI_k \cap \text{FSET}) \cap \text{FSET} \cap \{I \mid LB_k(I) < t\}$.

- β'_1 I is in $\overline{C}$: I is frequent, and thus $freq(I) = UB_k(I)$, because otherwise I would have been in $ubNDI_k \cap \text{FSET}$.
- β'_2 I is not in $\overline{C}$: I is infrequent.

Theorem 29 *$ubNDIRep_k$ and $ubNDIRep'_k$ are both concise representations of the frequent sets. Furthermore, $ubNDIRep'_k \sqsubseteq NDIRep_k$. For all $k \geq 1$ it holds that $ubNDIRep_{k+1} \sqsubseteq ubNDIRep_k$ and $ubNDIRep'_{k+1} \sqsubseteq ubNDIRep'_k$.*

Proof
Given the discussion above, it is immediate that $ubNDIRep_k$ and $ubNDIRep'_k$ are both representations.

For $ubNDIRep'_k \sqsubseteq NDIRep_k$, suppose that an itemset I is not in $NDIRep_k$. Then, either I is infrequent, or $LB_k(I) = UB_k(I) \geq t$. In the first case, I is not in $ubNDIRep'_k$, because $ubNDIRep'_k$ only contains frequent sets. In the second case, I is not in the first component of $ubNDIRep'_k$ because $ubNDI_k \subseteq NDI_k$, and not in the second component because $LB_k(I) \geq t$. Therefore, if I is not in $NDIRep_k$, then neither it is in $ubNDIRep'_k$.

For the third part it suffices to notice that for all sets $S \subseteq T$, $S \cup cover(S) \subseteq T \cup cover(T)$ □

Thus, $ubNDIRep'_k$ is always a subset of $NDIRep_k$. Since $ubNDIRep_k$ can contain infrequent sets, it is not true that $ubNDIRep_k$ is always a subset of $NDIRep_k$. For the other inclusions, that is, for $NDIRep$ in $ubNDIRep_k$ and $NDIRep$ in $ubNDIRep'_k$, consider the following counterexamples.

$$\mathcal{D}_1 = \begin{array}{|c|c|}\hline \texttt{TID} & \texttt{Items} \\ \hline 1 & a \\ \hline \end{array} \qquad \begin{array}{l} freq(\{\}) = 1 \\ freq(a) = 1 \end{array}$$

In this database, with $t = 1$,

$$ubNDIRep_1 = \{(\phi, 1)\} \ ,$$

and

$$NDIRep = ubNDIRep' = \{(\phi, 1), (a, 1)\} \ .$$

The second example is the following:

$$\mathcal{D}_2 = \begin{array}{|c|c|}\hline \texttt{TID} & \texttt{Items} \\ \hline 1 & a, b, c \\ 2 & a, b, c \\ 3 & a, b, c \\ 4 & a \\ 5 & b \\ 6 & c \\ 7 & \\ \hline \end{array} \qquad \begin{array}{l} freq(\{\}) = 1 \\ freq(a) = freq(b) = freq(c) = \frac{4}{7} \\ freq(ab) = freq(ac) = freq(bc) = \frac{3}{7} \\ freq(abc) = \frac{3}{7} \end{array}$$

In $\mathcal{D}_2$, $LB_2(abc) = \frac{2}{7}$, and $UB_2(abc) = \frac{3}{7}$. Thus, with $t = \frac{1}{7}$,

$$ubNDIRep'_2 = \left(\left\{(\phi, 1), \left(a, \frac{4}{7}\right), \left(b, \frac{4}{7}\right), \left(c, \frac{4}{7}\right), \left(ab, \frac{2}{7}\right), \left(ac, \frac{2}{7}\right), \left(bc, \frac{2}{7}\right)\right\}, \{\}\right)$$

and

$$NDIRep = \left\{(\phi, 1), \left(a, \frac{4}{7}\right), \left(b, \frac{4}{7}\right), \left(c, \frac{4}{7}\right), \left(ab, \frac{2}{7}\right), \left(ac, \frac{2}{7}\right), \left(bc, \frac{2}{7}\right), \left(abc, \frac{3}{7}\right)\right\}$$

Figure 6.3 gives the relations between the different representations introduced in this section. Notice that because $lbNDIRep_k$ only uses lower bounds, $lbNDIRep_{2l} = lbNDIRep_{2l+1}$ for all $l \geq 0$. A similar remark applies for $ubNDIRep_k$; since it only uses upper bounds, $ubNDIRep_{2l-1} = ubNDIRep_{2l}$ for all $l \geq 1$.

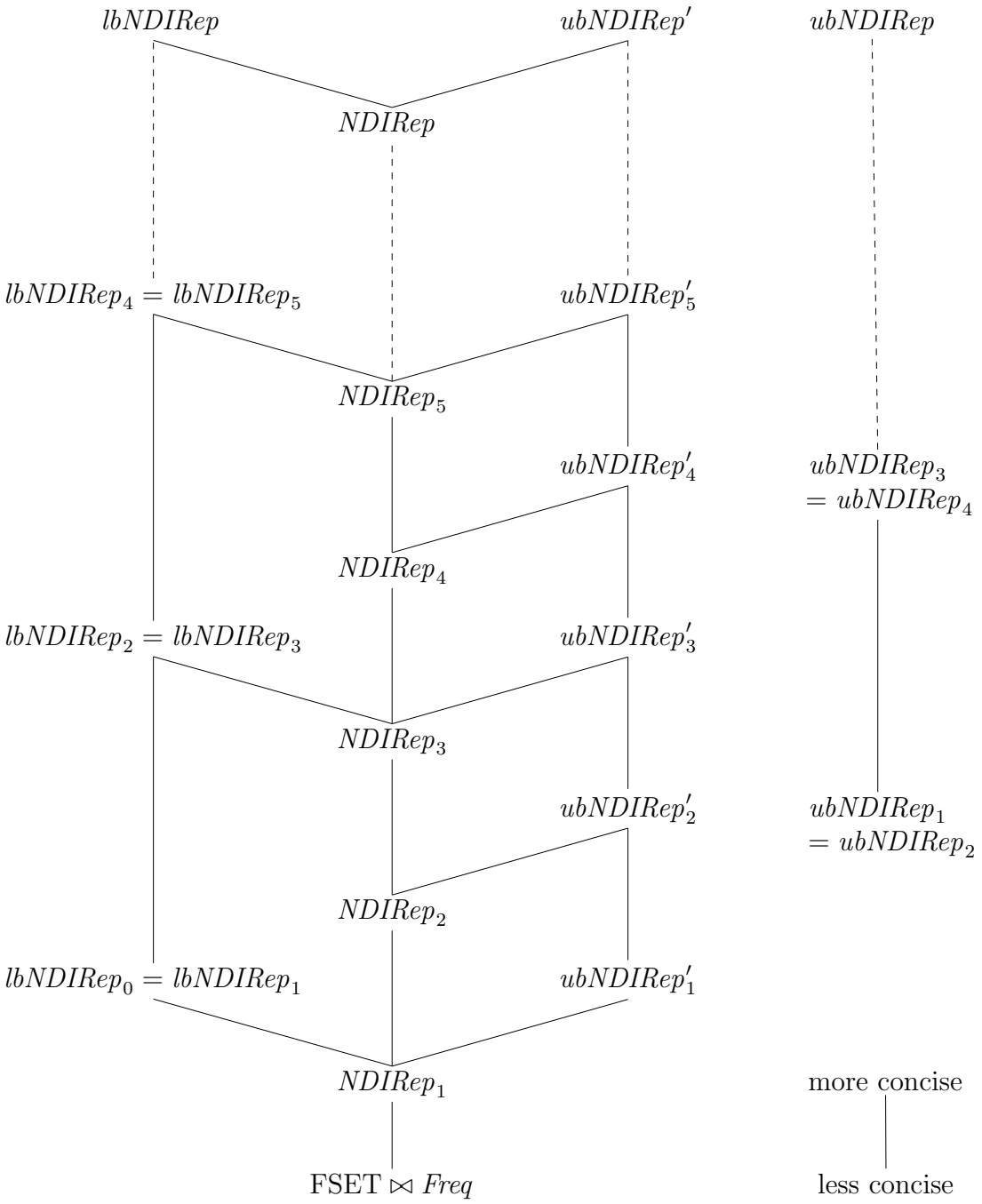


Figure 6.3: Relation between different representations. The lines denote, from top to bottom, the relation "more concise".

6.4 Unifying Framework

In this section we will show how the different representations (except the closed sets representation) can be seen as the manifestation of the same strategy. This strategy is as follows: a set I is stored with its frequency in a representation if its frequency equal $LB_k(I)$ nor $UB_k(I)$, for a fixed k. This information is not enough, so some of the sets in the cover need to be stored as well. Depending on the type of the representation, the frequent, the infrequent, the sets with $freq(I) = UB_k(I)$, or the sets with $freq(I) = UB_k(I)$ are stored. A key tool will be the notion of a k-free set as direct extension of free sets, and the analysis of the covers of these k-free sets.

6.4.1 k-Free Sets

We next define k-freeness of itemsets. We stress that this definition is relative to a transaction database, since it involves the bounds $LB_k(I)$ and $UB_k(I)$.

Definition 18

- *A set I is said to be* k-free*, if $freq(I) \neq LB_k(I)$ and $freq(I) \neq UB_k(I)$.*
- *A set I is said to be ∞-free, if $freq(I) \neq LB(I)$, and $freq(I) \neq UB(I)$.*
- *The set of all k-free (∞-free) sets is denoted by $Free_k$ ($Free_\infty$).* □

As the next lemma states, these definitions cover freeness, disjunction-freeness, and generalized disjunction-freeness.

Lemma 25 *Let I be an itemset.*

- *I is free if and only if I is 1-free*
- *I is disjunction free if and only if I is 2-free.*
- *I is generalized disjunction-free if and only if I is ∞-free.*

Proof

- The rules of depth 1 are exactly the rules $freq(I) \leq freq(I - \{a\}, \mathcal{D})$. Therefore, I is 1-free if and only if the frequency of I does not equal the frequency of one of its subsets, and hence I is free.

- The rules of depth 2 are the rules $freq(I) \leq freq(I-a) + freq(I-b) - freq(I-ab)$. According to Lemma 20, $freq(I-a) + freq(I-b) - freq(I-ab) = freq(I)$ if and only if the $I-ab \rightarrow (a \vee b)$ holds. If $a = b$, $freq(I-a) + freq(I-b) - freq(I-ab)$ reduces to $freq(I-a)$. Hence, there is a rule $I-a \rightarrow a$ that holds if and only if $freq(I) = UB_1(I)$, and there is a rule $I-ab \rightarrow (a \vee b)$, $a \neq b$ that holds if and only if $freq(I) = LB_2(I)$. Therefore, I is disjunction-free if and only if it is 2-free.

- The third statement can be proven in a similar way as the previous one, using Lemma 23. □

Let now $FreqFree_k$ be the set $Free_k \cap \text{FSET}$. As we argued before for the sets *FreqFree* and *FreqDFree*, $FreqFree_k \bowtie Freq$ is not a representation. Indeed, if a set I is not in the representation, there is no way to know whether I was left out the representation because I is infrequent, or because $freq(I) = LB_k(I)$, or because $freq(I) = UB_k(I)$. To resolve this problem, parts of the cover $cover(FreqFree_k)$ have to be stored. If we can restore the cover exactly, the other sets can be determined as well. This can be seen as follows: if a set I is not in $cover(FreqFree_k)$, and not in $FreqFree_k$, then it has a subset J in the cover. If this set J is infrequent, then so is I. If $freq(J) = LB_k(J)$, then $freq(I) = LB_k(I)$, and, if $freq(J) = UB_k(J)$, then $freq(I) = UB_k(I)$. Hence, if we can restore the complete cover, then we can restore all information.

The sets in $cover(FreqFree_k)$ can be divided in different groups, depending on whether they are frequent or not, have frequency equal to the lower bound or not, and have frequency equal to the upper bound or not. In order to make the discussion easier, we introduce a 3-letter notation to denote the different groups in the cover. The first letter denotes whether the sets in the group are frequent: f is frequent, i is infrequent. The second letter is l if the sets I in the group have $freq(I) = LB_k(I)$, otherwise it is $\overline{l}$. The third letter is u for groups with $freq(I) = UB_k(I)$, and $\overline{u}$ otherwise. The rule depth k is indicated as a subscript to the notation. For example, $f\overline{l}u_k$ denotes the group

$$f\overline{l}u_k =_{def} cover(FreqFree_k) \cap \text{FSET} \cap \{I \mid freq(I) \neq LB_k(I)\} \cap \{I \mid freq(I) = UB_k(I)\} ,$$

and $il\overline{u}_k$ denotes the group

$$il\overline{u}_k =_{def} cover(FreqFree_k) \cap Infrequent \cap \{I \mid freq(I) = LB_k(I)\} \cap \{I \mid freq(I) \neq UB_k(I)\} .$$

We split some of the groups even further, based on whether or not the bounds allow to conclude that a set is certainly frequent or certainly infrequent. For example, in the group $f\overline{l}u$, we distinguish between sets I such that the lower bound allows to derive that I is frequent, and the other sets. That is, $cf\overline{l}u$ (c of certain), is the set

$$cf\overline{l}u_k =_{def} cover(FreqFree_k) \cap \text{FSET} \begin{array}{l} \cap \{I \mid freq(I) \neq LB_k(I)\} \\ \cap \{I \mid freq(I) = UB_k(I)\} \\ \cap \{I \mid LB_k(I) \geq t\} \ . \end{array}$$

The other sets are in $uf\overline{l}u$ (u of uncertain). Thus, $uf\overline{l}u$ is the set

$$uf\overline{l}u_k =_{def} cover(FreqFree_k) \cap \text{FSET} \begin{array}{l} \cap \{I \mid freq(I) \neq LB_k(I)\} \\ \cap \{I \mid freq(I) = UB_k(I)\} \\ \cap \{I \mid LB_k(I) < t\} \ . \end{array}$$

Some of the groups only contain certain or uncertain sets, such as $fl\overline{u}$. Since $fl\overline{u}$ only contains frequent sets I with $freq(I) = LB_k(I)$, automatically the condition $LB_k(I) \geq t$ is fulfilled. The different groups are depicted in Figure 6.4. Notice that there are no groups with code $f\overline{lu}$, because sets that are frequent, and have a frequency that equals neither the lower, nor the upper bound must be in $FreqFree_k$, and hence cannot be in $cover(FreqFree_k)$. To make notations more concise, we will sometimes leave out some of the letters. For example, fl denotes the union $flu \cup fl\overline{u}$, and $i\overline{l}$ denotes $i\overline{l}u \cup ci\overline{lu} \cup ui\overline{lu}$.

Instead of storing the complete cover in a representation, we can restrict ourselves to some of the groups. It is, for example, not necessary to store the sets in the groups flu and ilu, because the sets I in these groups have $freq(I) = LB_k(I) = UB_k(I)$, and thus, are derivable. Furthermore, it is not necessary to store sets in $i\overline{l}u$, $cil\overline{u}$, and $ci\overline{lu}$, because these sets have $UB_k(I) < t$. Therefore, when we have to decide, based on the representation, whether sets in these groups are frequent, we can directly decide, based on $UB_k(I)$, that these sets must be infrequent.

We can reduce the number of groups even more. For some subsets $\mathcal{G}$ of the remaining groups

$$\{fl\overline{u}, cf\overline{l}u, uf\overline{l}u, uil\overline{u}, ui\overline{lu}\} \ ,$$

the structure

$$FreqFree_k \bowtie Freq \times \Pi_{g \in \mathcal{G}} g_k$$

will be a representation, and for some groups not. For example, the group $\mathcal{G} = \{fl\overline{u}, cf\overline{l}u\}$ gives the representation

$$(FreqFree_k \bowtie Freq, fl\overline{u}_k, cf\overline{l}u) \ .$$

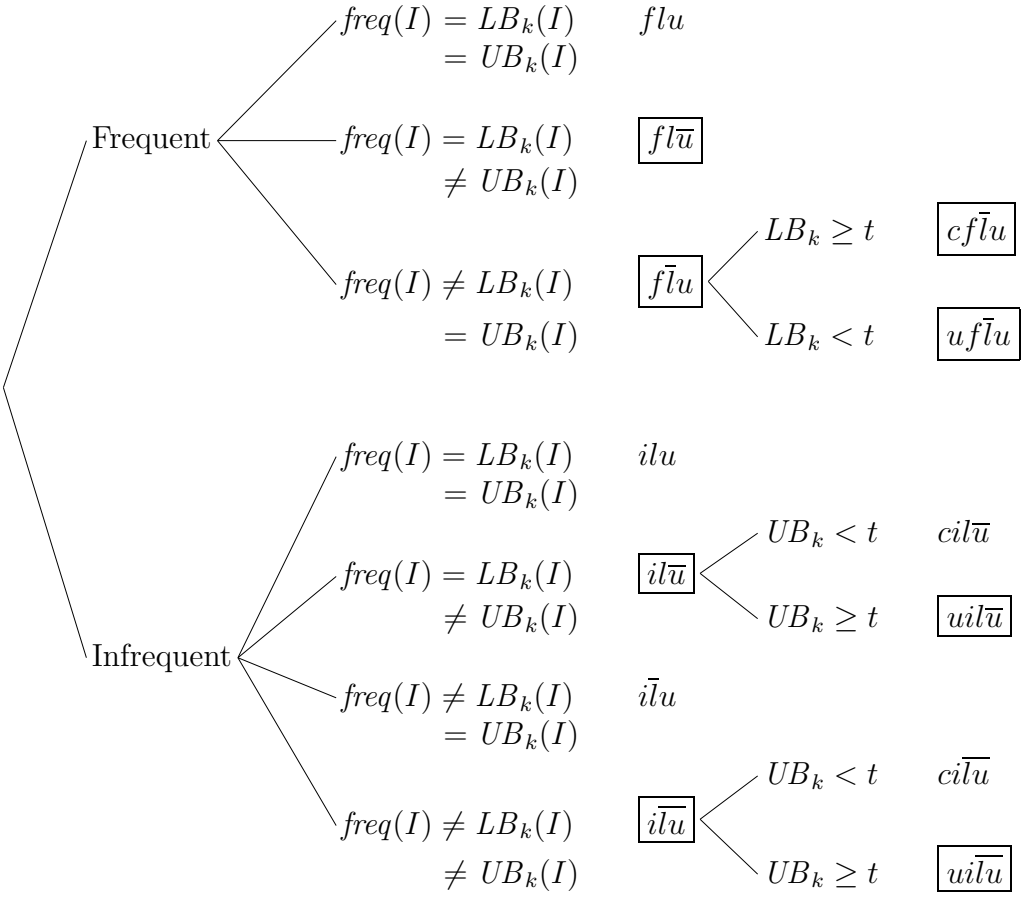


Figure 6.4: Types of itemsets in the cover

We denote the structure associated with $\mathcal{G}$ and rules up to depth k with $\mathcal{S}_k(\mathcal{G})$. Notice that there is no need to store the frequencies of the sets in $fl\overline{u}_k$; for all sets I in $fl\overline{u}_k$, $freq(I) = LB_k(I)$. Hence, we can derive the frequency. Similar observations apply for the other groups. We will now identify the minimal subsets $\mathcal{G}$ such that the associated structure is a representation.

The only minimal groups $\mathcal{G}$ such that the associated structures are representations are:

$$\begin{aligned}
\mathcal{G}_1 &= \{fl\overline{u}, uf\bar{l}u\} \ , \\
\mathcal{G}_2 &= \{cf\bar{l}u, uf\bar{l}u\} \ , \\
\mathcal{G}_3 &= \{fl\overline{u}, uil\overline{u}, ui\overline{lu}\} \ , \text{ and} \\
\mathcal{G}_4 &= \{cf\bar{l}u, uil\overline{u}, ui\overline{lu}\} \ .
\end{aligned}$$

Hence, we get the following theorem:

Theorem 30 *Let*

$$\mathcal{G} \subseteq \{fl\overline{u}, cf\bar{l}u, uf\bar{l}u, uil\overline{u}, ui\overline{lu}\} \ .$$

$\mathcal{S}_k(\mathcal{G})$ is a representation if and only if either $\mathcal{G}_1 \subseteq \mathcal{G}$, or $\mathcal{G}_2 \subseteq \mathcal{G}$, or $\mathcal{G}_3 \subseteq \mathcal{G}$, or $\mathcal{G}_4 \subseteq \mathcal{G}$.

Proof

The proof of this theorem will be in two stages. First, we show that $\mathcal{S}_k(\mathcal{G}_1)$, $\mathcal{S}_k(\mathcal{G}_2)$, $\mathcal{S}_k(\mathcal{G}_3)$, and $\mathcal{S}_k(\mathcal{G}_4)$ are representations. Second, we give counterexamples that show that for the following sets $\mathcal{G}$, $\mathcal{S}_k(\mathcal{G})$ is not a representation:

$$\{fl\overline{u}, cf\bar{l}u, uil\overline{u}\} \ , \ \{fl\overline{u}, cf\bar{l}u, ui\overline{lu}\} \ , \text{ and } \{uf\bar{l}u, uil\overline{u}, ui\overline{lu}\} \ .$$

Since every subset of

$$\{fl\overline{u}, cf\bar{l}u, uf\bar{l}u, uil\overline{u}, ui\overline{lu}\}$$

is either a superset of $\mathcal{G}_1$, $\mathcal{G}_2$, $\mathcal{G}_3$, or $\mathcal{G}_4$ or is a subset of the three sets above, these two parts suffice. Indeed, the superset of a representation is again a representation, and the subset of a non-representation is not a representation as well.

$\mathcal{S}_k(\mathcal{G}_1)$, $\mathcal{S}_k(\mathcal{G}_2)$, $\mathcal{S}_k(\mathcal{G}_3)$, and $\mathcal{S}_k(\mathcal{G}_4)$ are representations

We can always recognize sets that are in the groups flu, ilu, $cil\overline{u}$, $i\bar{l}u$, and $ci\overline{lu}$. We will therefore call these groups the *certain groups*. Thus, for $i = 1, \ldots, 4$, we need to show that we can distinguish between the groups in

$$\{fl\overline{u}, cf\bar{l}u, uf\bar{l}u, uil\overline{u}, ui\overline{lu}\} - \mathcal{G}_i \ .$$

$\mathcal{G}_1 = \{fl\overline{u}, uf\bar{l}u\}$: The sets that are left out are the ones in the groups $cf\bar{l}u$, $uil\overline{u}$, and $ui\overline{lu}$. Thus, if we have a set I that is in $cover(FreqFree_k)$, but not in $\mathcal{S}(\mathcal{G}_1)$, and not in one of the certain groups, then we know that I must be in one of $cf\bar{l}u$, $uil\overline{u}$, and $ui\overline{lu}$. It is straightforward that such a set I is in $cf\bar{l}u$ if and only if $LB_k(I) \geq t$. Therefore, we can decide whether I is frequent or not, and if I is frequent, then it is in $cf\bar{l}u$, and thus has $freq(I) = UB_k(I)$.
$\mathcal{G}_2 = \{cf\bar{l}u, uf\bar{l}u\}$: A set I in the cover of $FreqFree_k$, but not in $\mathcal{S}(\mathcal{G}_2)$, and not in the certain groups is in $fl\overline{u}$ if and only if $LB_k(I) \geq t$, otherwise I is in $uil\overline{u}$ or $ui\overline{lu}$, and hence infrequent.
$\mathcal{G}_3 = \{fl\overline{u}, uil\overline{u}, ui\overline{lu}\}$: A set I in the cover of $FreqFree_k$, but not in $\mathcal{S}(\mathcal{G}_3)$, and not in the certain groups is in $cf\bar{l}u$ if and only if $LB_k(I) \geq t$, otherwise I is in $uf\bar{l}u$.
$\mathcal{G}_4 = \{cf\bar{l}u, uil\overline{u}, ui\overline{lu}\}$: We can distinguish between sets in $fl\overline{u}$ and $uf\bar{l}u$ as

follows: if $LB_k(I) < t$, then I cannot be in $fl\overline{u}$, if $LB_k(I) \geq t$, then I cannot be in $uf\bar{l}u$.

Counterexamples

Suppose that $\mathcal{G}$ is a subset of $\{fl\overline{u}, cf\bar{l}u, uf\bar{l}u, uil\overline{u}, ui\overline{lu}\}$ for which we want to show that $\mathcal{S}_k(\mathcal{G})$ is not a representation. We can proof this by giving an example database $\mathcal{D}$ and threshold t such there are two itemsets $a_1 \ldots a_n$ and $b_1 \ldots b_n$ with the following properties: (a) the substitutions $a_i \rightarrow b_i$, $i = 1, \ldots, n$ leave the structure $\mathcal{S}_k(\mathcal{G})$ unchanged, and (b) either $a_1 \ldots a_n$ is frequent and $b_1 \ldots b_n$ is infrequent or they are both frequent, but with $freq(a_1 \ldots a_n, \mathcal{D}) \neq freq(b_1 \ldots b_n, \mathcal{D})$. Such an example shows that the structure $\mathcal{S}_k(\mathcal{G})$ is not a representation, since it does not allow for distinguishing between $a_1 \ldots a_n$ and $b_1 \ldots b_n$.

$\mathcal{S}_k(\{fl\overline{u}, cf\bar{l}u, uil\overline{u}\})$ is not a representation

Consider the following database:

$\mathcal{D}_1 =$

TID	Items
1	a,b,c
2	a,b,c
3	a,b,c
4	a,b,c
5	a,b,c
6	a,c
7	b
8	

$$\begin{array}{llll} freq(\{\}) = 1 & & freq(ab) = \frac{5}{8} \\ freq(a) = \frac{3}{4} & & freq(ac) = \frac{3}{4} \\ freq(b) = \frac{3}{4} & & freq(bc) = \frac{5}{8} \\ freq(c) = \frac{3}{4} & & freq(abc) = \frac{5}{8} \end{array}$$

The lower and upper bounds of depth 2 on the frequency of ab, ac, and bc give the interval $\left[\frac{1}{2}, \frac{3}{4}\right]$. Let now $t = \frac{3}{4}$. Hence we have:

$$FreqFree_2 = \{\{\}, a, b, c\} \ , \qquad cover(FreqFree_2) = \{ab, ac, bc\} \ .$$

This gives the following groups in the cover:

$$ui\overline{lu} = \{ab, bc\} \ , \qquad uf\bar{l}u = \{ac\} \ .$$

Because neither ab, nor ac is stored in $\mathcal{S}_2(\{fl\overline{u}, cf\bar{l}u, uil\overline{u}\})$, we cannot distinguish between the frequent set ac and the infrequent set ab.

$\mathcal{S}_k(\{fl\overline{u}, cf\bar{l}u, ui\overline{lu}\})$ is not a representation

Consider the following database:

$\mathcal{D}_2 =$

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

$$\begin{array}{llll} freq(\{\}) = 1 & & freq(ab) = \frac{1}{2} \\ freq(a) = \frac{3}{4} & & freq(ac) = \frac{3}{4} \\ freq(b) = \frac{3}{4} & & freq(bc) = \frac{1}{2} \\ freq(c) = \frac{3}{4} & & freq(abc) = \frac{1}{2} \end{array}$$

The lower and upper bounds of depth 2 on the frequency of ab, ac, and bc give the interval $\left[\frac{1}{2}, \frac{3}{4}\right]$. Let now $t = \frac{3}{4}$. Hence we have:

$$\mathit{FreqFree}_2 \;=\; \{\{\}, a, b, c\} \;, \qquad \mathit{cover}(\mathit{FreqFree}_2) \;=\; \{ab, ac, bc\} \;.$$

This gives the following groups in the cover:

$$\mathit{uil}\overline{u} \;=\; \{ab, bc\} \;, \qquad \mathit{uf}\bar{l}u \;=\; \{ac\} \;.$$

Because neither ab, nor ac is stored in $\mathcal{S}_2(\{\mathit{fl}\overline{u}, \mathit{cf}\bar{l}u, \mathit{ui}\overline{\mathit{lu}}\})$, we cannot distinguish between the frequent set ac and the infrequent set ab.

$\{\mathit{uf}\bar{l}u, \mathit{uil}\overline{u}, \mathit{ui}\overline{\mathit{lu}}\}$ is not a representation
Consider the following database:

$\mathcal{D}_3 =$

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

$$\begin{array}{llll} \mathit{freq}(\{\}) = 1 & & \mathit{freq}(ab) = \frac{1}{2} \\ \mathit{freq}(a) = \frac{3}{4} & & \mathit{freq}(ac) = \frac{3}{4} \\ \mathit{freq}(b) = \frac{3}{4} & & \mathit{freq}(bc) = \frac{1}{2} \\ \mathit{freq}(c) = \frac{3}{4} & & \mathit{freq}(abc) = \frac{1}{2} \end{array}$$

$\mathcal{D}_3$ is invariant under permutations of a and c. The lower and upper bounds of depth 2 on the frequency of ab, ac, and bc give the interval $\left[\frac{1}{2}, \frac{3}{4}\right]$. Let now $t = \frac{1}{2}$. Hence we have:

$$\mathit{FreqFree}_2 \;=\; \{\{\}, a, b, c\} \;, \qquad \mathit{cover}(\mathit{FreqFree}_2) \;=\; \{ab, ac, bc\} \;.$$

This gives the following groups in the cover:

$$\mathit{fl}\overline{u} \;=\; \{ab, bc\} \;, \qquad \mathit{cf}\bar{l}u \;=\; \{ac\} \;.$$

Because neither ab, nor ac is stored in $\mathcal{S}_2(\mathit{uf}\bar{l}u, \mathit{uil}\overline{u}, \mathit{ui}\overline{\mathit{lu}})$, we cannot distinguish between the two frequent sets ab and ac. Because these two sets have different frequency this proves that $\mathcal{S}_2(\mathit{uf}\bar{l}u, \mathit{uil}\overline{u}, \mathit{ui}\overline{\mathit{lu}})$ is not a representation. □

Table 6.1 describes the different representations in terms of the groups. For the generalized disjunction-free generators representation there is a slight difficulty. The authors of [58, 57] did not realize that $X \rightarrow \bigvee Y$ gives rise to a lower bound if $|Y|$ is odd, and an upper bound if $|Y|$ is even. Therefore, when pruning the cover, the lower bound is not used as a evaluation criterium. The non-free sets however, are pruned from the representation. Therefore, in fact, for the sets in the cover, only rules of depth 1 are evaluated. Thus,

$$\begin{array}{lll} f\overline{u}_{\infty,1} \;=_{def}\; \mathit{cover}(\mathit{FreqFree}_\infty) & \cap \text{FSET} \\ & \cap \{I \mid \mathit{UB}_1(I) \neq \mathit{freq}(I)\} \;. \end{array}$$

Representation	Base	With frequency	Without frequency
FreeRep	$FreqFree_1$		$\overline{u}_1$
FreeRep$'$	$FreqFree_1$		$f\bar{l}u_1$
DFreeRep	$FreqFree_2$	complete cover	
DFreeGenRep	$FreqFree_2$	$fl\overline{u}_2$	$i\overline{u}_2$
GDFreeRep	$FreqFree_\infty$	complete cover	
GDFreeGenRep	$FreqFree_\infty$	$f\overline{u}_{\infty,1}$	$i\overline{u}_{\infty,1}$
NDIRep	$FreqFree_\infty$	$f\bar{l}u_\infty, fl\overline{u}_\infty$	
$lbNDIRep_{2k}$	$FreqFree_{2k}$	$f\bar{l}u_{2k}$	
$= lbNDIRep_{2k+1}$			
$ubNDIRep_{2k-1}$	$FreqFree_{2k-1}$	$fl\overline{u}_{2k-1}$	$ui\overline{u}_{2k-1}$
$= ubNDIRep_{2k}$			
$ubNDIRep'_{2k-1}$	$FreqFree_{2k-1}$	$fl\overline{u}_{2k-1}$	$uf\bar{l}u_{2k-1}$

Table 6.1: Representations in function of the groups in the cover

For the first six representations, the correctness of the Table 6.1 can easily be seen using the definitions of these representations. For the last six representations, this is more difficult, because the definitions of these last representations are not based on $FreqFree_k$ and $cover(FreqFree_k)$. For example, $ubNDIRep_k$ is based on $ubNDI_k \cap \text{FSET}$ and $cover(ubNDI_k \cap \text{FSET})$. The next lemma however, expresses the sets on which these representations are based in function of $FreqFree_k$ and the different groups.

Lemma 26

- $NDI \cap \text{FSET} = FreqFree_\infty \cup f\bar{l}u_\infty \cup fl\overline{u}_\infty$
- $lbNDI_{2k} \cap \text{FSET} = FreqFree_{2k} \cup f\bar{l}u_{2k}$
- $ubNDI_{2k-1} \cap \text{FSET} = FreqFree_{2k-1} \cup fl\overline{u}_{2k-1}$
- $\left(\begin{array}{l} cover(ubNDI_{2k-1} \cap \text{FSET}) \\ \cap\ NDI_{2k-1} \end{array} \right) = \left(\begin{array}{l} cover(FreqFree_{2k-1}) \\ -fl\overline{u}_{2k-1} - flu_{2k-1} - ilu_{2k-1} \end{array} \right)$
- $\left(\begin{array}{l} cover(ubNDI_{2k-1} \cap \text{FSET}) \\ \cap\ Infrequent \\ \cap\ \{I \mid UB_{2k-1}(I) \geq t\} \end{array} \right) = ui\overline{u}_{2k-1}$
- $\left(\begin{array}{l} cover(ubNDI_{2k-1} \cap \text{FSET}) \\ \cap\ \text{FSET} \\ \cap\ \{I \mid LB_{2k-1}(I) < t\} \end{array} \right) = uf\bar{l}u_{2k-1}$

Proof
Let I not be in $FreqFree_\infty \cup f\bar{l}u_\infty \cup fl\overline{u}_\infty$. Then either I is infrequent, or I is a derivable itemset, since every superset of sets in $f\bar{l}u_\infty$ and $fl\overline{u}_\infty$ must be derivable. Hence, I is in $FreqFree_\infty \cup f\bar{l}u_\infty \cup fl\overline{u}_\infty$ if and only if I is non-derivable and frequent.

The inclusion $FreqFree_{2k} \cup f\bar{l}u_{2k} \subseteq lbNDI_{2k} \cap \text{FSET}$ is straightforward. For the other inclusion, let I be a set in $lbNDI_{2k} \cap \text{FSET} - FreqFree_{2k}$. Since I is in $lbNDI_{2k} \cap \text{FSET}$, I is frequent, and $freq(I) \neq LB_{2k}(I)$. Because I is not in $FreqFree_{2k}$, either I is infrequent, or $freq(I) = LB_{2k}(I)$, or $freq(I) = UB_{2k}(I)$. Hence, we can conclude that I is frequent, has $freq(I) \neq LB_{2k}(I)$, and has $freq(I) = UB_{2k}(I)$. We show next that I has to be in $cover(FreqFree_{2k})$. Let J be a strict subset of I. J is frequent because of monotonicity. Furthermore, on the one hand, $freq(J) \neq LB_{2k}(J)$, because otherwise $freq(I) = LB_{2k}(I)$. On the other hand, suppose $freq(J) = UB_{2k}(J)$. Since $UB_{2k-1}(J) = UB_{2k}(J)$ (rules of even depth are lower bounds), this implies that $UB_{2k-1}(J) = freq(J)$, and hence $LB_{2k}(I) = freq(I)$ (Theorem 26). This is a contradiction, and thus $freq(J) \neq UB_{2k}(J)$. Because J is frequent, and has $freq(J) \neq LB_{2k}(J)$ and $freq(J) \neq UB_{2k}(J)$, $J \in FreqFree_{2k}$. Since J was an arbitrary strict subset of I, I must be in $cover(FreqFree_{2k})$. Hence, I is in $cover(FreqFree_{2k})$, I is frequent, $freq(I) \neq LB_{2k}(I)$, and $freq(I) = UB_{2k}(I)$. Therefore, I is in $f\bar{l}u$.

The proof of the equality $ubNDI_{2k-1} \cap \text{FSET} = FreqFree_{2k-1} \cup fl\overline{u}_{2k-1}$ is similar.

We show that

$$cover(FreqFree_{2k-1} \cup fl\overline{u}_{2k-1}) \cap NDI_{2k-1} = cover(FreqFree_{2k-1}) - fl\overline{u}_{2k-1} - flu_{2k-1} - ilu_{2k-1} \ .$$

Let I be in $cover(FreqFree_{2k-1} \cup fl\overline{u}_{2k-1})$. Then, all subsets of I are in $FreqFree_{2k-1} \cup fl\overline{u}_{2k-1}$. Suppose I has a strict subset J in $fl\overline{u}_{2k-1}$. Then $freq(J) = LB_{2k-1}(J) = LB_{2k-2}(J)$, and thus $freq(I) = LB_{2k-1}(I)$ and also $freq(I) = UB_{2k-1}(I)$. Hence this implies that I is derivable. Therefore, if I is in $cover(FreqFree_{2k-1} \cup fl\overline{u}_{2k-1}) \cap NDI_{2k-1}$, then all subsets of I are in $FreqFree_{2k-1}$, and thus, I is in

$$cover(FreqFree_{2k-1}) - fl\overline{u}_{2k-1} - flu_{2k-1} - ilu_{2k-1} \ .$$

For the other inclusion, suppose I is in $cover(FreqFree_{2k-1}) - fl\overline{u}_{2k-1} - flu_{2k-1} - ilu_{2k-1}$. Since I is not in $flu_{2k-1} \cup ilu_{2k-1}$, I is in NDI_{2k-1}. Because I is in $cover(FreqFree_{2k-1}) - fl\overline{u}_{2k-1}$, I is in $cover(FreqFree_{2k-1} \cup fl\overline{u}_{2k-1})$.

The last two equalities follow directly from the third equality and the fact that both $Infrequent \cap \{I \mid UB_{2k-1}(I) \geq t\}$ and $\text{FSET} \cap \{I \mid LB_{2k-1}(I) < t\}$

are subsets of NDI_{2k-1}. □

Corollary 7 *For each entry* $(\mathcal{R}, B, \{p_1, \ldots, p_k\}, \{q_1, \ldots, q_l\})$ *in Table 6.1, it holds that for every database* $\mathcal{D}$ *and frequency threshold* t*,*

$$\mathcal{R} \equiv (B \bowtie Freq, p_1 \bowtie Freq, \ldots, p_k \bowtie Freq, q_1, \ldots, q_l) \ .$$

Notice that in the summary table, $NDIRep_k$ and $ubNDIRep'_{2k}$ are missing. The reason for this is that for these representations, it is not true that they consist of $FreqFree_k$ or $FreqFree_{2k}$ together with a part of the cover. The reason for this is that supersets J of a set I with $freq(I) = LB_{2k}(I)$, have $freq(J) = LB_{2k}(J)$, and $freq(J) = UB_{2k+1}(J)$, but not necessarily $UB_{2k}(J) = freq(J)$. Consider for example the following database:

$\mathcal{D} =$

TID	Items
1	a
2	b
3	a,b
4	a,c
5	b,c
6	a,b,c
7	a,d
8	b,d
9	a,b,d
10	a,c,d
11	b,c,d
12	a,b,c,d

$$\begin{array}{llll} freq(\{\}) = 1 & & freq(bc) = \frac{1}{3} \\ freq(a) = \frac{2}{3} & & freq(bd) = \frac{1}{3} \\ freq(b) = \frac{2}{3} & & freq(cd) = \frac{1}{4} \\ freq(c) = \frac{1}{2} & & freq(abc) = \frac{1}{6} \\ freq(d) = \frac{1}{2} & & freq(abd) = \frac{1}{6} \\ freq(ab) = \frac{1}{3} & & freq(acd) = \frac{1}{6} \\ freq(ac) = \frac{1}{3} & & freq(bcd) = \frac{1}{6} \\ freq(ad) = \frac{1}{3} & & freq(abcd) = \frac{1}{12} \end{array}$$

The lower and upper bounds up to depth 2 in $\mathcal{D}$ are:

$$\begin{array}{llll} freq(\{\}) \in [0, \infty] & & freq(bc) \in \left[\frac{1}{6}, \frac{1}{2}\right] \\ freq(a) \in [0, 1] & & freq(bd) \in \left[\frac{1}{6}, \frac{1}{2}\right] \\ freq(b) \in [0, 1] & & freq(cd) \in \left[0, \frac{1}{2}\right] \\ freq(c) \in [0, 1] & & freq(abc) \in \left[\frac{1}{6}, \frac{1}{3}\right] \\ freq(d) \in [0, 1] & & freq(abd) \in \left[\frac{1}{6}, \frac{1}{3}\right] \\ freq(ab) \in \left[\frac{1}{3}, \frac{2}{3}\right] & & freq(acd) \in \left[\frac{1}{6}, \frac{1}{4}\right] \\ freq(ac) \in \left[\frac{1}{6}, \frac{1}{2}\right] & & freq(bcd) \in \left[\frac{1}{6}, \frac{1}{4}\right] \\ freq(ad) \in \left[\frac{1}{6}, \frac{1}{2}\right] & & freq(abcd) \in \left[\frac{1}{12}, \frac{1}{6}\right] \end{array}$$

Hence, all sets are in $ubNDI_2$ and in NDI_2. However,

$$FreqFree_2 = \{\{\}, a, b, c, d, ac, ad, bc, bd, cd, acd, abd\} \ .$$

The sets ab, abc, abd, and $abcd$ are not 2-free. The cover of $FreqFree_2$ is $\{ab\}$. This example shows that $ubNDI_2$ and NDI_2 are not necessarily subsets of $FreqFree_2 \cup cover(FreqFree_2)$.

6.4.2 Closures of Representations

Another approach to improve representations that is orthogonal to the approach describe above is based on closures. It is based on the observation that for a set of itemsets $\mathcal{S}$, $cl(\mathcal{S}) \bowtie Freq$ is always smaller than and contains the same information as $\mathcal{S} \bowtie Freq$. Hence, given a representation $\mathcal{S}_k(\{g_1, \ldots, g_n\})$, we construct the representation $cl(\mathcal{S}_k(\{g_1, \ldots, g_n\}))$ as follows:

$$\begin{aligned} &cl(\mathcal{S}_k(\{g_1, \ldots, g_n\})) \ =_{def} \\ &\qquad (cl(FreqFree_k) \bowtie Freq, g_1 - cl(FreqFree_k), \ldots, g_n - cl(FreqFree_k)) \ . \end{aligned}$$

We can easily reconstruct the original representation $\mathcal{S}_k(\{g_1, \ldots, g_n\})$ based on $cl(\mathcal{S}_k(\{g_1, \ldots, g_n\}))$.

For a representation $\mathcal{R}$, let $sets(\mathcal{R})$ denote the set of itemsets I such that either $(I, freq(I))$ or I is in one of the components of $\mathcal{R}$, and let $fsets(\mathcal{R})$ denote the set of itemsets I such that $(I, freq(I))$ is in one of the components of $\mathcal{R}$. From the definition of the closure of a representation, we can easily derive the next theorem.

Theorem 31 *Let $\mathcal{S}_k(\{g_1, \ldots, g_n\})$ be a representation.*

$$|sets(\mathcal{S}_k(\{g_1, \ldots, g_n\}))| \geq |sets(cl(\mathcal{S}_k(\{g_1, \ldots, g_n\})))| \ ,$$

and

$$|fsets(\mathcal{S}_k(\{g_1, \ldots, g_n\}))| \geq |fsets(cl(\mathcal{S}_k(\{g_1, \ldots, g_n\})))| \ .$$

Example 17 *Let $\mathcal{D}$ be the following database:*

$\mathcal{D} =$

TID	Items
1	a,b,c,d
2	a,b,c
3	a
4	b,c,d
5	b,c,d
6	b
7	c,d
8	d
9	

$$
\begin{array}{llll}
freq(\{\}) = 1 & & freq(bc) = \frac{4}{9} \\
freq(a) = \frac{1}{3} & & freq(bd) = \frac{1}{3} \\
freq(b) = \frac{5}{9} & & freq(cd) = \frac{4}{9} \\
freq(c) = \frac{5}{9} & & freq(abc) = \frac{2}{9} \\
freq(d) = \frac{5}{9} & & freq(abd) = \frac{1}{9} \\
freq(ab) = \frac{2}{9} & & freq(acd) = \frac{1}{9} \\
freq(ac) = \frac{2}{9} & & freq(bcd) = \frac{1}{3} \\
freq(ad) = \frac{1}{9} & & freq(abcd) = \frac{1}{9}
\end{array}
$$

The lower and upper bounds up to depth 2 in $\mathcal{D}$ are:

$$
\begin{array}{llll}
freq(\{\}) \in [0,\infty] & & freq(bc) \in \left[\frac{1}{9},\frac{5}{9}\right] \\
freq(a) \in [0,1] & & freq(bd) \in \left[\frac{1}{9},\frac{5}{9}\right] \\
freq(b) \in [0,1] & & freq(cd) \in \left[\frac{1}{9},\frac{5}{9}\right] \\
freq(c) \in [0,1] & & freq(abc) \in \left[\frac{1}{9},\frac{2}{9}\right] \\
freq(d) \in [0,1] & & freq(abd) \in \left[0,\frac{1}{9}\right] \\
freq(ab) \in \left[0,\frac{1}{3}\right] & & freq(acd) \in \left[\frac{1}{9},\frac{1}{9}\right] \\
freq(ac) \in \left[0,\frac{1}{3}\right] & & freq(bcd) \in \left[\frac{1}{3},\frac{1}{3}\right] \\
freq(ad) \in \left[0,\frac{1}{3}\right] & & freq(abcd) \in \left[\frac{1}{9},\frac{1}{9}\right]
\end{array}
$$

Let $t = \frac{2}{9}$. The representation $\mathcal{S}_2(cf\bar{l}u, uf\bar{l}u)$ is

$$\left(\{\ \{\}, a, b, c, d, ab, ac, ad, bc, bd, cd\ \} \bowtie Freq, \{\}, \{abc\}\right) \ .$$

The closure of this representation, $cl(\mathcal{S}_2(cf\bar{l}u, uf\bar{l}u))$ is

$$\left(\{\ \{\}, a, b, c, d, ad, bd, cd, abc, bcd\ \} \bowtie Freq, \{\}, \{\}\right) \ .$$

□

6.4.3 Relations Between the Representations

From Table 6.1, we can derive relations between the different representations. In Figure 6.5, these relations are depicted.

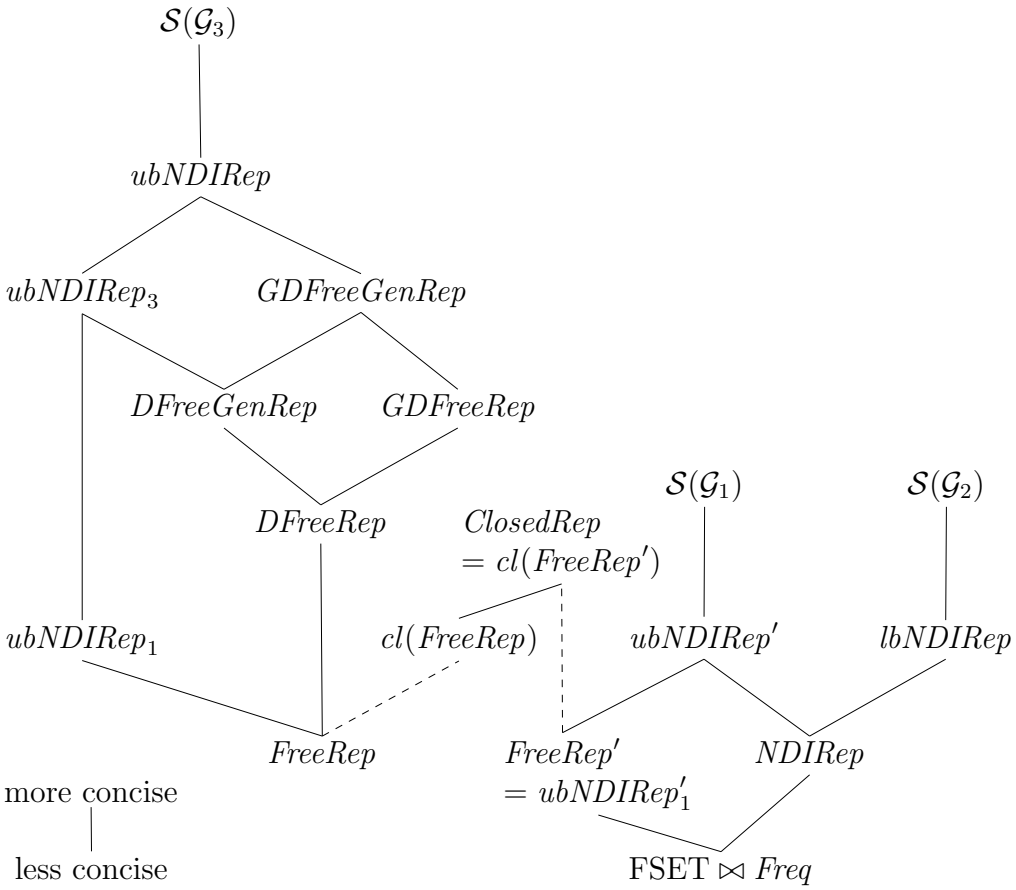


Figure 6.5: Relations between the different representations. Solid lines denote the relation "more concise", dashed lines denote the relation "smaller than". The more to the top, the more concise the representations become.

Most of the inclusions in Figure 6.5 are straightforward if we look at Table 6.1. For $FreeRep' = ubNDIRep'_1$ and $ubNDIRep_1 \sqsubseteq FreeRep$, we point out that $cf\bar{l}u_1$ and $fl\overline{u}_1$ are both empty, since the only lower bound of depth 1 is the trivial bound 0 (We assume implicitly that $t \neq 0$.) For the equality $ClosedRep = cl(FreeRep')$ we use Lemma 18 to proof that $ClosedRep$ is in $cl(FreeRep)$ and $cl(FreeRep')$. The inclusion of $cl(FreeRep')$ in $ClosedRep$ follows from the fact that $FreeRep'$ only contains frequent sets.

7

Related Work

In this chapter we discuss related work in probabilistic logics, approximate inclusion-exclusion, statistical data protection, data mining algorithm optimization, and concise representations.

7.1 Probabilistic Logics

In artificial intelligence literature, probabilistic logic [41] and reasoning about uncertainty and belief [71] is studied intensively. The link with this thesis 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. Let

$$\mathcal{C} = \{freq(I_1) \in [l_1, u_1], \ldots, freq(I_n) \in [l_n, u_n]\}$$

be a FREQSAT-problem. Every item a in one of the sets I_i's can be associated with a proposition P_a, and an itemset I with the conjunction of the propositions associated with the items it contains. Hence,

$$I \equiv \bigwedge_{a \in I} P_a \ .$$

A transaction T can then be considered as a truth assignment; in the assignment represented by T, proposition P_a is true if and only if transaction T contains item a. In this view, a transaction database $\mathcal{D}$ becomes a probability distribution. The probability of a certain truth assignment equals the fraction of transactions in the database representing this truth assignment. The probability that in the actual world I is true, is in this respect equal to the frequency of I.

Nilsson introduced in [68] the following *probabilistic logic problem*: given a finite set of m logical sentences $S_1, \ldots, S_m$ defined on a set $X = \{x_1, \ldots, x_n\}$

of n boolean variables with the usual boolean operators $\wedge, \vee$, and $\neg$, together with probabilities $p_1, \ldots, p_m$, does there exists a probability distribution on the possible truth assignments of X, such that the probability of S_i being true, is *exactly* p_i for all $1 \leq i \leq m$. *Georgakopoulos et al.* proved [34] that this problem, they suggest the name *probabilistic satisfiability problem* (PSAT), is NP-complete. Notice that this result implies that FREQSAT for sets of frequency constraints of the form

$$\mathcal{C} \;=\; \{freq(I_1) = f_1, \ldots, freq(I_n) = f_n\}$$

is in **NP**. This proof can easily be extended to include intervals. It is this extended proof that is in 2. For the **NP**-completeness there is not directly a straightforward reduction. In the logic introduced by *Nilsson*, arbitrary propositional logic sentences are allowed. The frequency constraints however, can only model conjunctions; that is, no negations nor disjunctions. However, as shown in Chapter 2, the proof in [34] can easily be extended to our case.

Another interesting problem, also stated by *Nilsson* in [68], is that of *probabilistic entailment*. Again a set of logical sentences $S_1, \ldots, S_m$, together with probabilities $p_1, \ldots, p_m$ is given, and one extra logical sentence S_{m+1}, the target. It is asked to find best possible upper and lower bounds on the probability that S_{m+1} is true, given $S_1, \ldots, S_m$ are satisfied with respective probabilities $p_1, \ldots, p_m$. The interval defined by these lower and upper bounds forms the so-called *tight entailment* of S_{m+1}.

For a comprehensive overview of probabilistic logic, probabilistic entailment and various extensions, we refer to [46, 45]. *Nilsson's* probabilistic logic and entailment are extended in various ways, including assigning intervals to logical expressions instead of exact probability values and also considering conditional probabilities [30, 60, 61]. In [61], *Lukasiewicz* studies the complexity of (tight) entailment of conditional probabilities in a systematic and structured way. This study of *Lukasiewicz* was a motivation to study the complexity of FREQENT and T-FREQENT. Because the frequency constraints cannot express conditional constraints, the results we obtain in Chapter 2 are stronger than the completeness results in [61].

In [26], *Fagin et al.* study the following extension. 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 semantics are introduced by an underlying probability space. The weight of a formula corresponds to the probability that it is true. The main contribution of [26] is the description of a sound and complete axiomatization for this probabilistic logic. All types

of frequency constraints can be expressed in this probabilistic logic. The frequent set expression $freq(K) \geq p_K$ can be translated as $w(\bigwedge_{i \in K} P_i) \geq p_K$. Since *Fagin et al.* showed that deciding satisfiability for this logic is **NP**-complete, it follows that FREQSAT is in **NP**. In Chapter 2 however, we have chosen explicitly for adapting the proofs in [34], since they are more instructive, and we needed the representation of the FREQSAT-problem as a linear programming instance in the rest of the thesis.

Also in [30], axioms for a probabilistic logic are introduced. However, the authors are unable to prove completeness of the axioms. For a restricted sub-language (Type-A problems), they prove that their set of axioms is complete. However, this sub-language is not sufficiently powerful to express frequency constraints. In [30], the usefulness of an axiomatization is motivated by the fact that it provides human-readable proofs. Also, when inference is stopped before termination, still a partial inference of the frequencies is provided.

A great number of inference rules have been proposed in artificial intelligence studies. As pointed out in [49], rule based deduction has its biggest advantage over global optimization (e.g. linear programming) when working in a restricted setting with specialized knowledge bases. When studying deduction rules, in most artificial intelligence work, locally complete rules are studied; when global completeness is required, linear programming techniques are more appropriate. For example, in [60], *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.

In [49], *Jaeger* develops a method for automatic derivation of probabilistic inference rules for conditional probabilities comparable to the method proposed in Chapter 5. 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. *Jaeger* however does not use elimination methods as we do in Chapter 5, but instead analyzes a list of the parameterized vertices of the polytope $V(\mathcal{C})$ consisting of the instantiations that satisfy the input constraints.

7.2 Combinatorics

7.2.1 Approximate Inclusion-Exclusion

Probabilists and statisticians frequently use the inclusion-exclusion bounds to approximate the probability of a union of finitely many events. The inclusion-

exclusion principle was first discovered by *Jordan* [52] and later on rediscovered by *Bonferroni* [7]. The inclusion-exclusion principle allows to calculate the number of elements in the union of sets $S_1, \ldots, S_n$ given the numbers of elements in all possible intersections $S_{i_1} \cap \ldots \cap S_{i_k}$, $1 \leq i_1, \ldots, i_k \leq n$, $k \leq n$. If for some of these intersections, the number of elements is missing, we can only calculate an approximate bound on the size of the union $S_1 \cup \ldots \cup S_n$. It is exactly this type of problems that is studied in *approximate inclusion-exclusion* [31, 53, 65]. *Melkman and Shimony* study in [65] the case in which only the count of the number of items in $S_1 \cap \ldots \cap S_n$ is missing. As is showed in Chapter 4, in this case, the bounds on the union $S_1 \cup \ldots \cup S_n$ provide us with bounds on the intersection $S_1 \cap \ldots \cap S_n$. Both problems are alike, and hence many of the results of *Melkman and Shimony* also apply to our framework. Actually, the completeness and non-redundancy of the inclusion-exclusion rules $\mathcal{R}_I(J)$ for the frequency of the itemset I are also implicitly proven in [65], even though the proof there is much more involved, and does not provide the same insight as our proof in Chapter 4.

Bonferroni inequalities are a specific family of combinatorial inequalities for approximate inclusion-exclusion when all intersections up to a fixed constant k are known [31]. An interesting application of Bonferroni inequalities to data mining is described in [50, 51]. Based on the frequencies of some itemsets, bounds on the frequency of arbitrary boolean expressions are calculated using these Bonferroni inequalities. The bounds obtained in [50, 51] are however not tight.

7.2.2 Fréchet Bounds

Fréchet bounds [29] are often used in stochastic processes to estimate an upper and/or a lower bound on the queue length in a queuing system with two different but known marginal inter-arrivals times distributions of two types of customers. The simplest form of the bounds is the following.

$$\max(0, P(A) + P(B) - 1) \leq P(AB) \leq \min(P(A), P(B))$$

The lower bound corresponds to the rule $\mathcal{R}_{AB}(\{\})$. The upper bounds are the monotonicity rules $\mathcal{R}_{AB}(A)$ and $\mathcal{R}_{AB}(B)$.

7.2.3 Statistical Data Protection

In statistical databases the privacy of data is studied [28, 24, 23]. In many situations it is common to only provide aggregated data instead of giving

the individual data records. An example of this is census data, in which the individual data records are protected, but at the same time aggregated values are published. Statistical data protection tries to answer the question to which extend the privacy of the data might be compromised by combining different aggregates. In this context the questions studied are very alike our work. If the aggregates are sums, then the main question becomes "based on the sums for different combinations of characteristics, what can we derive for other combinations." Such a combination of characteristics can be seen as an itemset, and the count as its support.

7.3 Data Mining

7.3.1 Counting Inference

MAXMINER [6] In MAXMINER, *Bayardo* uses the following rule to derive a lower bound on the support of an itemset:

$$support(I \cup J) \geq support(I) - \sum_{j \in J} drop(K, j)$$

with $K \subset I$, and $drop(K, j) = support(K) - support(K \cup \{j\})$. The intuition behind this rule is the following: $drop(K, j)$ expresses how many transactions contain K, but do not contain j. Hence, if we add the item j to the itemset K, the support will decrease. How much the support drops from K to $K \cup \{j\}$, is expressed by $drop(K, j)$. $\sum_{j \in J} drop(K, j)$ is used as an estimate of the drop from I to $I \cup J$. Indeed, for every transaction T that does contain I, but does not contain $I \cup J$, there is at least one $j \in J$ such that T does not contain $K \cup \{j\}$. *Bayardo* uses this rule when searching for the frequent itemsets of maximal cardinality. The lower bound is used to jump from I to $I \cup J$ in the search space whenever the lower bound on $I \cup J$ is at least as high as the frequency threshold.

For $K = I$ and $J = \{i_1, i_2\}$, this rule correspond to $\mathcal{R}_{I \cup \{i1,i2\}}(I)$. In general, the rule used in MAXMINER can be derived from the deduction rules in Chapter 4, because the deduction rules $\mathcal{R}_I(J)$ are complete. For example, let $I = abcd$, $J = bcd$, and $K = \{\}$. The MAXMINER rule can be derived from the rules $\mathcal{R}_{abcd}(.)$ as follows:

$$\begin{aligned} &3 \cdot \mathcal{R}_{abcd}(\{\}) + 2 \cdot \mathcal{R}_{abcd}(a) + 2 \cdot \mathcal{R}_{abcd}(b) + 2 \cdot \mathcal{R}_{abcd}(c) + 2 \cdot \mathcal{R}_{abcd}(d) \\ &\quad + \mathcal{R}_{abcd}(ab) + \mathcal{R}_{abcd}(ac) + \mathcal{R}_{abcd}(ad) + \mathcal{R}_{abcd}(bc) + \mathcal{R}_{abcd}(bd) \\ &\quad + \mathcal{R}_{abcd}(cd) + \mathcal{R}_{abcd}(abc) + \mathcal{R}_{abcd}(abd) + \mathcal{R}_{abcd}(acd) + \mathcal{R}_{abcd}(bcd) \end{aligned}$$

gives the MAXMINER rule

$$support(abcd) \geq support(a) + support(b) + support(c) + support(d) - 3support(\{\}) \ .$$

PASCAL In their PASCAL-algorithm, *Bastide et al.* [5] use counting inference to avoid counting the support of all candidates. The rule they are using to avoid counting is based on our rule $\mathcal{R}_I(I - \{i\})$. In fact, the PASCAL-algorithm is the Apriori- algorithm in which the counting of sets derivable with $\mathcal{R}_I(I - \{i\})$ are not counted in the database. In view of the deduction rules presented in Chapter 4, PASCAL can straightforwardly be extended using all rules $\mathcal{R}_I(.)$ for a candidate set I.

7.3.2 Interactive Association Rule Mining

We would also like to point out some analogues with interactive association rule mining. In [36, 37], the authors develop a framework that allows to reuse results of previous data mining queries. For example, parts of the answer to the query asking for the supports of all itemsets containing a certain item A can be reused to answer the query that asks for the support of all itemsets that do not contain B. The deduction rules introduced here can be used orthogonally to this approach. Based on previous results, bounds on the support of new, not yet counted itemsets can be calculated using the supports of other itemsets.

7.3.3 Deduction

Another application of deduction rules is developed in [38]. Based on the observation that highly frequent items tend to blow up the output of a data mining query by an exponential factor, the authors develop a technique to leave out these highly frequent items, and to reintroduce them after the mining phase by using a deduction rule, called the *multiplicative* rule. The multiplicative rule can be stated as follows: let I, J be itemsets, then

$$support(I \cup J, \mathcal{D}) \geq support(I, \mathcal{D}) + support(J, \mathcal{D}) - support(\{\}, \mathcal{D}) \ .$$

This rule can be derived from the rules in our framework. For $J = \{a, b\}$ for example, the multiplicative rule corresponds to $\mathcal{R}_{I \cup \{a,b\}}(I)$.

7.3.4 Completeness

There have been attempts to prove completeness results for pruning in frequent itemset mining. One such attempt is described shortly in [59]. In the presence of constraints on the allowable itemsets, the authors introduce the notion of *ccc-optimality*[1]. ccc-optimality can intuitively be understood as "in every loop, the algorithm only generates and tests allowable itemsets that can still be frequent. For this the algorithm does not use redundant operations." Our notion of tight entailment however, is more general, since we do not restrict ourselves to a particular algorithm.

7.4 Concise Representations

In the literature, there exist already a number of concise representations for frequent itemsets. In Chapter 6, we discussed different proposals in the literature in depth. The main goal of concise representations is to store the collection of frequent itemsets and their frequencies in an as concise as possible way. In this perspective, more efficient deduction methods allow for making more concise representations, since the more we can derive, the less we need to store. Since in Chapter 6 the different concise representations have been studied in much detail we refer to Chapter 6 for related work about this topic.

[1] ccc-optimality stands for Constraint Checking and Counting-optimality

8

Summary and Further Work

Summary Frequent itemset mining is one of the most important problems in data mining. Based on the importance in frequent set mining of pruning criteria such as the monotonicity principle, we studied what information can be derived if we have information about the frequencies of some itemsets. To this end, frequency constraints, such as $freq(I) \in [l, u]$, were introduced as a mean to model information about frequencies.

In Chapter 2, we concentrated on the formal introduction of frequency constraints and on satisfiability and implication problems. The goal of this chapter was to establish the general structure and properties of the problems studied in the thesis. The satisfiability problem of a set of frequency constraints was formalized as the FREQSAT-problem, and can be seen as an algorithm-independent abstraction and generalization of the pruning problem. The implication problems were formalized as FREQENT for implication and T-FREQENT for tight implication. The general properties of frequency constraints were explored. Connection between the satisfiability and implication problems for frequencies and solving linear programming problems and were given. This connection resulted in a the proof that FREQSAT is **NP**-complete and a graphical interpretation of the FREQSAT-problem.

Due to the high complexity of the FREQSAT-problem, special cases are studied. A first restriction, described in Chapter 3, was to focus on lower bounds and upper bounds in isolation. In the lower bound case, only expressions of the form $freq(I) \geq l$ were allowed. Since systems of such expressions are always satisfiable, the notion of completeness was introduces. Completeness of a set of frequency constraints expresses that the information implicitly contained in the set is also explicitly contained. Hence, all constraints in the set must be tightly entailed. For the lower bounds situation, a complete axiomatization was given, and it was proven that completeness can be decided in polynomial time. In the upper bound case only constraints of the form $freq(I) \leq u$ were studied. Deciding completeness for sets of upper bound

constraints was shown to be logarithmic space.

Another special case, studied in Chapter 4, was based on the information we have in the Apriori-algorithm. In this algorithm, for each candidate set, we know the frequency of all its subsets exactly. Hence, tight implication for a set I, based on the set of constraints $\{freq(J) = f_J \mid J \subset I\}$ was studied. Sound and complete deduction rules for this type of tight implication were given. Based on the deduction rules, the notion of derivable itemset was introduced. An itemset is called derivable in a database $\mathcal{D}$, if, based on the frequencies of its subsets, the frequency of the set can be derived perfectly. As some limited experiments in Chapter 4 showed, this situation is not very uncommon. This observation was further strengthened by a theorem stating that any set of size larger than $^2\log(\mathcal{D})+1$ must be derivable in the database $\mathcal{D}$. Based on the properties of derivable itemsets, such as monotonicity, an algorithm for finding all frequent non-derivable itemsets is proposed.

In Chapter 6, an application of the deduction rules in Chapter 4 is given. Based on the deduction rules, different concise representations are constructed. Also an overview of the concise representations in the literature is given, and the connections with the representations based on the deduction rules are given by constructing a unifying framework.

In Chapter 7, related work was discussed. Especially links with probabilistic logics, Bonferroni inequalities, and optimization of frequent itemset mining algorithms is explored.

Further work For our further work, there are different directions we can pursue.

- In Chapter 4, we studied a special case that was closely connected to the Apriori-algorithm, a breadth-first algorithm. Another interesting problem is to see what can be derived from the information we get in depth-first algorithms such as the FPGrowth-algorithm [43], and to extend the notion of a derivable itemset to this situation.

- Besides frequent itemset mining, in other problems monotonicity of frequency is very important as well. A future research direction is to explore to what extent the deduction rules can be extended to for example approximate dependencies [54], roll-up dependencies [79, 21], sequence mining [64, 3], and binary expressions [16]. In [20], a preliminary study about the applicability of monotonicity for data mining queries defined in a simple data mining query language is presented.

This work can be seen as a first step in the direction of extending the deduction rules to a broader framework.

- The concise representation for frequent itemsets based on the deduction rules presented in Chapter 6 are all subset-closed. A possible extension is to include representations "with holes." The study of such representations requires the development of deduction rules for the frequency of a set I if both frequencies of sub- and supersets are known.

- Up to now we used mainly combinatorial techniques for deriving lower and upper bounds on the frequency of itemsets. Instead of these combinatorial bounds, statistical estimates for the frequency could be used. For example, in [74], the use of probabilistic models based on maximum entropy is studied: based on a set of given frequencies, an estimate for the frequency of a target itemset is calculated, using the principle of maximum entropy. We are also interested in the other direction: given a database, and a strategy for inference (e.g. logical implication, or maximum entropy), find a collection of sets such that this collection allows for accurate inference of the other frequencies. Since these collections are highly dependent on the inference strategy used, they can be very different from one strategy to another.

Bibliography

[1] 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, Washington, D.C., 1993.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proc. VLDB Int. Conf. Very Large Data Bases*, pages 487–499, Santiago, Chile, 1994.

[3] R. Agrawal and R. Srikant. Mining sequential patterns. In *Proc. IEEE ICDE Int. Conf. on Data Engineering*, pages 3–14, Taipei, Taiwan, 1995.

[4] K. Ali, S. Manganaris, and R. Srikant. Partial classification using association rules. In *Proc. KDD Int. Conf. Knowledge Discovery in Databases*, pages 115–118, 1997.

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

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

[7] C.E. Bonferroni. Teoria statistica della classi e calcolo della probabilitá. *Publicazioni del R. Instituto Superiore di Scienze Economiche e Commerciali di Firenze*, 8:1–62, 1936.

[8] J.-F. Boulicaut and A. Bykowski. Frequent closures as a concise representation for binary data mining. In *Proc. PaKDD Pacific-Asia Conf. on Knowledge Discovery and Data Mining*, pages 62–73, 2000.

[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, M. Klemettinen, and H. Mannila. Modeling KDD processes within the inductive database framework. In *Proc. DaWaK Int. Conf. Data Warehousing and Knowledge Discovery*, pages 293–302, 1999.

[11] S. Brin, R. Motwani, J.D. Ullman, and S. Tsur. Dynamic itemset counting and implication rules for market basket data. In *Proc. ACM SIGMOD Int. Conf. Management of Data*, pages 255–264, Tucson, AZ, 1997.

[12] A. Bykowski and C. Rigotti. A condensed representation to find frequent patterns. In *Proc. PODS Int. Conf. Principles of Database Systems*, 2001.

[13] A. Bykowski, J.K. Seppänen, and J. Hollmén. Model-independent bounding of the supports of boolean formulae in binary data. In *In Proceedings ECML-PKDD Workshop Knowledge Discovery in Inductive Databases (KDID)*, pages 20–31, 2002.

[14] T. Calders. Deducing bounds on the frequency of itemsets. In *EDBT Workshop DTDM Database Techniques in Data Mining*, 2002.

[15] 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.

[16] T. Calders and J. Paredaens. Mining frequent binary expressions. In *Proc. DaWaK Int. Conf. Data Warehousing and Knowledge Discovery*, pages 399–408, Greenwich, 2000.

[17] T. Calders and J. Paredaens. A theoretical framework for reasoning about frequent itemsets. Technical Report 2000-06, University of Antwerp, Dept. Math. & Computer Science, 2000.

[18] T. Calders and J. Paredaens. Axiomatization of frequent sets. In *Proc. ICDT Int. Conf. Database Theory*, pages 204–218, London, UK, 2001.

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

[20] T. Calders and J. Wijsen. On monotone data mining langauages. In *Proc. DBPL Workshop on Databases and Programming Languages*, 2001.

[21] T. Calders, J. Wijsen, and R.T. Ng. Searching for dependencies at multiple abstraction levels. *ACM Trans. on Database Systems*, 27(3):229–260, 2002.

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

[23] A. Dobra. Computing sharp integer bounds for entries in contingency tables given a set of fixed marginals. Technical report, Department of Statistics, Carnegie Mellon University, http://www.stat.cmu.edu/∼adobra/bonf-two.pdf, 2001.

[24] A. Dobra and S.E Fienberg. Bounds for cell entries in contingency tables given marginal totals and decomposable graphs. *Proceedings of the National Academy of Sciences*, 97(22):11885–11892, 2000.

[25] G. Dong and J. Li. Efficient mining of emerging patterns: Discovering trends and differences. In *Proc. KDD Int. Conf. Knowledge Discovery in Databases*, pages 43–52, 1999.

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

[27] R. Fagin and M. Y. Vardi. Armstrong databases for functional and inclusion dependencies. *Information Processing Letters*, 16(1):13–19, 1983.

[28] S. E. Fienberg. Fréchet and bonferroni bounds for multi-way tables of counts with applications to disclosure limitation. In *Statistical Data Protection*, 1998.

[29] M. Fréchet. Sur les tableaux de correlation dont les marges sont donnés. *Ann. Univ. Lyon Sect A, Series 3*, 14:53–77, 1951.

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

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

[32] B. Ganter and R. Wille. *Formal Concept Analysis — Mathematical Foundations*. Springer, 1999.

[33] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. Freeman, New York, 1979.

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

[35] B. Goethals. *Efficient Frequent Pattern Mining.* PhD thesis, Transnational University Limburg, Belgium, December 2002.

[36] B. Goethals and J. Van den Bussche. A priori versus a posteriori filtering of association rules. In *ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, 1999.

[37] B. Goethals and J. Van den Bussche. On supporting interactive association rule mining. In *Proc. DaWaK Int. Conf. Data Warehousing and Knowledge Discovery*, pages 307–316, 2000.

[38] D. Groth and E. Robertson. Discovering frequent itemsets in the presence of highly frequent items. In *In Proceedings Workshop on Rule Based Data Mining, in Conjunction with the 14th International Conference On Applications of Prolog*, 2001.

[39] D. Gunopulos, H. Mannila, and S. Saluja. Discovering all most specific sentences by randomized algorithms. In *Proc. ICDT Int. Conf. Database Theory*, pages 215–229, 1997.

[40] G. Hadley. *Linear Programming.* Addison-Wesley, Reading, Mass., 1962.

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

[42] J. Han and M. Kamber. *Data Mining: Concepts and Techniques.* Morgan Kaufmann, 2000.

[43] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In *Proc. ACM SIGMOD Int. Conf. Management of Data*, pages 1–12, Dallas, TX, 2000.

[44] D. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining.* MIT Press, 2001.

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

[46] 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.

[47] S. Hettich and S. D. Bay. *The UCI KDD Archive. [http://kdd.ics.uci.edu].* Irvine, CA: University of California, Department of Information and Computer Science, 1999.

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

[49] M. Jaeger. Automatic derivation of probabilistic inference rules. *Int. J. of Approxiamte Reasoning*, 28(1):1–22, 2001.

[50] S. Jaroszewicz and D. A. Simivici. Support approximations using bonferroni-type inequalities. In *Proc. PKDD Int. Conf. Principles of Data Mining and Knowledge Discovery*, pages 212–224, 2002.

[51] S. Jaroszewicz, D. A. Simivici, and I. Rosenberg. An inclusion-exclusion result for boolean polynomials and its applications in data mining. In *Proc. of the Discrete Mathematics in Data Mining Workshop, SIAM Datamining Conference*, 2002.

[52] Ch. Jordan. The foundations of the theory of probability. *Mat. Phys. Lapok*, 34:109–136, 1927.

[53] J. Kahn, N. Linial, and A. Samorodnitsky. Inclusion-exclusion: Exact and approximate. *Combinatorica*, 16:465–477, 1996.

[54] J. Kivinen and H. Mannila. Approximate inference of functional dependencies from relations. *Theoretical Computer Science*, 149:129–149, 1995.

[55] D.E. Knuth. *Fundamental Algorithms*. Addison-Wesley, Reading, Massachusetts, 1997.

[56] M. Kryszkiewicz. Concise representation of frequent patterns based on disjunction-free generators. In *Proc. IEEE Int. Conf. on Data Mining*, pages 305–312, 2001.

[57] M. Kryszkiewicz and M. Gajek. Concise representation of frequent patterns based on generalized disjunction-free generators. In *Proc. PaKDD Pacific-Asia Conf. on Knowledge Discovery and Data Mining*, pages 159–171, 2002.

[58] M. Kryszkiewicz and M. Gajek. Why to apply generalized disjunction-free generators representation of frequent patterns? In *Proc. International Syposium on Methodologies for Intelligent Systems*, pages 382–392, 2002.

[59] L. V.S. Laksmanan, R.T. Ng, J. Han, and A. Pang. Optimization of constrained frequent set queries with 2-variable constraints. In *Proc. ACM SIGMOD Int. Conf. Management of Data*, pages 157–168, Philadelphia, Pennsylvania, 1999.

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

[61] T. Lukasiewicz. Probabilistic logic programming with conditional constraints. INFSYS Research Report 1843-00-01, Institut für Informationssysteme, Abteilung Wissenbasierte Systeme, 2000.

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

[63] H. Mannila and H. Toivonen. Levelwise search and borders of theories in knowledge discovery. *DMKD*, 1(3):241–258, 1997.

[64] H. Mannila, H. Toivonen, and A. I. Verkamo. Discovering frequent episodes in sequences. In *Proc. KDD Int. Conf. Knowledge Discovery in Databases*, pages 210–215, 1995.

[65] A. A. Melkman and S. E. Shimony. A note on approximate inclusion-exclusion. *Discrete Applied Mathematics*, 73:23–26, 1997.

[66] S. Morishita and J. Sese. Traversing itemset lattice with statistical metric pruning. In *Proc. PODS Int. Conf. Principles of Database Systems*, pages 226–236, 2000.

[67] K. G. Murty. *Linear Programming.* Wiley, 1983.

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

[69] C.H. Papadimitriou. *Computational Complexity.* Addison-Wesley, 1994.

[70] C.H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity.* Prentence-Hall, Englewood Cliffs, N.J., 1982.

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

[72] 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.

[73] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Efficient mining of association rules using closed itemset lattices. *Journal of Information Systems*, 24(1):25–46, 1999.

[74] D. Pavlov, H. Mannila, and P. Smyth. Probabilistic models for query approximation with large sparse binary datasets. In *Uncertainty in Artificial Intelligence: Proceedings of the Sixteenth Conference (UAI-2000)*, pages 465–472, 2000.

[75] J. Pei, J. Han, and R. Mao. Closet: An efficient algorithm for mining frequent closed itemsets. In *ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, Dallas, TX, 2000.

[76] G. Stumme, R. Taouil, Y. Bastide, N. Pasquier, and L. Lakhal. Fast computation of concept lattices using data mining techniques. In *Knowledge Representation Meets Databases*, pages 129–139, 2000.

[77] P.-N. Tan and V. Kumar. Interestingness measures for association patterns : A perspective. In *KDD'2000 Workshop on Postprocessing in Machine Learning and Data Mining*, 2000.

[78] H. Toivonen. Sampling large databases for association rules. In *Proc. VLDB Int. Conf. Very Large Data Bases*, pages 134–145. Morgan Kaufman, 1996.

[79] J. Wijsen, R.T. Ng, and T. Calders. Discovering roll-up dependencies. In *Proc. KDD Int. Conf. Knowledge Discovery in Databases*, pages 213–222, San Diego, 1999.

[80] M.J. Zaki and C. Hsiao. ChARM: An efficient algorithm for closed association rule mining. In *Proc. SIAM Int. Conf. on Data Mining*, 2002.

[81] Z. Zheng, R. Kohavi, and L. Mason. Real world performance of association rule algorithms. In *Proc. KDD Int. Conf. Knowledge Discovery in Databases*, pages 401–406. ACM Press, 2001.

A

Nederlandse samenvatting

A.1 Voorkennis

De vooruitgang in databases en technologie maakt het mogelijk om grote hoeveelheden data te verzamelen, op te slaan en te bevragen. Bijna alle bedrijven en organisaties beschikken over enorme hoeveelheden data. Echter, niet enkel de hoeveelheid data is belangrijk, maar ook de mogelijkheid om ze te analyseren. Voor een bedrijf is het van vitaal belang om uit de data nuttige informatie en kennis te extraheren. Deze uitdaging is de motivatie voor *data mining*, een relatief jonge wetenschapsdiscipline, gesitueerd op het knooppunt tussen *database onderzoek*, *statistiek* en *machine learning*. In [44] wordt data mining als volgt gedefinieerd:

> Data Mining is de *analyse* van *grote observationele datasets* met als doel het vinden van *onverwachte verbanden* en het *samenvatten* van de data op nieuwe manieren die voor de eigenaar van de data zowel *verstaanbaar als bruikbaar* zijn.

We identificeren de belangrijkste begrippen in deze definitie:

- *Analyse.* In data mining tracht men belangrijke relaties, patronen en trends te identificeren met als doel een beter begrip van de data te krijgen. Hiervoor worden automatische tools ontwikkeld die de analist moeten helpen om een beter inzicht in de data te krijgen, en grote brokken data te verwerken tot begrijpbare kennis.

- *Grote observationele datasets.* De datasets die in data mining toepassingen beschouwd worden, zijn over het algemeen erg groot. Deze eigenschap maakt dat data mining algoritmes zeer efficiënt en schaalbaar moeten zijn om deze grote datasets aan te kunnen.

- *Onverwachte relaties.* In tegenstelling tot traditionele database systemen is er in data mining niet zo iets als een *exacte* query die beantwoord moet worden. In een ideale situatie, zou een gebruiker enkel het type relatie die hij of zij wil vinden ingeven, en het data mining algoritme gaat dan zelf op zoek naar de patronen van dat type die in de database aanwezig zijn.

- *Samenvatten.* De uitvoer van een data mining algoritme geeft typisch algemene kenmerken van de dataset. Deze karakteristieken bieden een ander, meer beknopt gezichtspunt op de data.

- *Verstaanbaar en bruikbaar.* De uitvoer van een data mining algoritme is voor een gebruiker enkel nuttig indien het geïnterpreteerd kan worden. Dit impliceert dat modellen die een grote voorspellende kracht hebben, maar niet door mensen begrepen kunnen worden, niet in aanmerking komen. We benadrukken hier echter dat deze vereiste niet door alle data mining onderzoekers onderschreven wordt.

Het Frequent Itemset Probleem Een van de meest prominente problemen in data mining is ongetwijfeld het *Frequent Itemset*-probleem [1]. De originele context van dit probleem was *market basket analysis* (letterlijk: analyse van winkelmandjes). Beschouw een winkel die producten uit een verzameling I verkoopt. Van elke klant van de winkel wordt bij elk bezoek telkens de set van aangekochte producten geregistreerd en opgeslagen in een database $\mathcal{D}$. Zo een set van producten noemen we een *transactie*. Gebaseerd op deze database, wil een analist te weten komen welke producten vaak samen worden verkocht. Deze setting wordt geformaliseerd door het frequent itemset probleem. Dit probleem is, gegeven een grens s en een database $\mathcal{D}$, vind alle deelverzamelingen van $\mathcal{I}$ die in minstens s van de transacties in $\mathcal{D}$ aanwezig zijn. De deelverzamelingen van $\mathcal{I}$ noemen we itemsets. Het aantal maal dat een itemset I in een database $\mathcal{D}$ voorkomt noemen we de *support van I in $\mathcal{D}$*, en wordt genoteerd als $supp(I, \mathcal{D},)$. De *frequentie van een itemset I in een database $\mathcal{D}$* is de support van I gedeeld door het totaal aantal transacties in $\mathcal{D}$ en wordt genoteerd als $freq(I, \mathcal{D})$. Itemsets die een support hoger dan s hebben worden *(s-)frequent* genoemd.

Het frequent itemset probleem staat in vele data mining algoritmes centraal. Het is een belangrijk deelprobleem bij het zoeken naar onder andere: associatie regels [1], sequentiële patronen [3], classificatie [4] en "emerging" patronen [25]. Sinds de introductie van het frequent itemset probleem in [1] zijn reeds vele verschillende benaderingen en algoritmes voorgesteld, vooral

in de context van associatie regels [1, 2, 43]. Voor overzichten van de verschillende technieken verwijzen we naar [43, 48, 81] en [35, Ch. 2].

Ondanks de eenvoudige definitie is het frequent itemset probleem verre van triviaal. Zo is bijvoorbeeld in [39] aangetoond dat gegeven een grens s voor de support, een database $\mathcal{D}$ en een natuurlijk getal k, het beslissingsprobleem dat vraagt of er een s-frequente itemset van lengte k bestaat in $\mathcal{D}$, **NP**-compleet is.

Monotoniciteit Alle algoritmes voor het vinden van frequente patronen maken gebruik van het volgende monotoniciteits principe [63].

> Laat $I_1 \subseteq I_2$ twee itemsets zijn. In elke transaction database $\mathcal{D}$, is de frequentie of I_2 maximaal zo hoog als de frequentie van I_1.

Deze simpele *deductieregel* is reeds vele malen succesvol toegepast. Het beste voorbeeld hiervan is het bekende Apriori-algoritme [2]. Om de monotoniciteit maximaal uit te buiten, start het Apriori-algoritme met het tellen van de singleton itemsets in één pass over de database. In een tweede pass worden dan enkel de itemsets $\{i_1, i_2\}$ geteld waarvan $\{i_1\}$ en $\{i_2\}$ s-frequent bevonden waren in de vorige pass. De andere itemsets van grootte 2 worden *gepruned*, aangezien ze nooit frequent kunnen zijn, gezien het monotoniciteitsprincipe. In de derde pass over de database worden dan enkel itemsets $\{i_1, i_2, i_3\}$ van grootte 3 geteld waarbij $\{i_1, i_2\}$, $\{i_1, i_3\}$, en $\{i_2, i_3\}$ allen s-frequent waren in de vorige stap. Dit gaat zo voort totdat er geen nieuwe frequente itemsets meer worden gevonden. De zoektocht naar frequente itemsets door het Apriori-algoritme kan als dusdanig beschouwd worden als de afwisseling tussen een *tel-fase* en een *meta-fase*. In de tel-fase worden de frequenties van een aantal vooraf bepaalde *kandidaten* geteld. In de meta-fase worden de resultaten van de tel-fase daarna geanalyseerd. Gebaseerd op het monotoniciteitsprincipe worden dan sommige itemsets *a-priori*, dat wil zeggen, zonder ze te tellen in de database, uitgesloten. Deze bemerkingen zijn ook van toepassing op andere frequent set mining algoritmes zoals DIC [11] en FPGrowth [43]. Sinds de invoering van het Apriori-algoritme zijn al vele verbeteringen voorgesteld. De meeste optimalisaties trachten ofwel de invoer zodanig te reorganiseren dat het tellen van de itemsets eenvoudiger wordt, ofwel het aantal passes over de database te minimaliseren. Er is echter weinig aandacht besteed aan het verbeteren van de pruning.

A.2 Onderwerp van de thesis

Onderzoeksvraag van deze thesis Ondanks het feit dat monotoniciteit van frequentie zeer vaak gebruikt wordt, is er weinig eerder werk dat tracht deze regel uit te breiden. Deze thesis bestudeert deductie regels, zoals het monotoniciteitsprincipe, op een algemene en theoretische manier. Dit houdt onder meer in dat we ons niet concentreren op een bepaald algoritme. De centrale vraag kan nu als volgt geformuleerd worden:

> Gegeven informatie over de frequenties van de itemsets $I_1, \ldots, I_n$, welke informatie kunnen wij afleiden in verband met de frequenties van andere itemsets?

In onze benadering van dit probleem staat de notie van een *frequentie constraint* centraal. Een *frequentie constraint* wordt gedefinieerd als een uitdrukking $\mathit{freq}(I) \in [l, u]$, met I een itemset, en l, u rationale getallen tussen 0 en 1. We zeggen dat een database $\mathcal{D}$ een frequentie constraint $\mathit{freq}(I) \in [l, u]$ *waar maakt* indien $\mathit{freq}(I, \mathcal{D}) \in [l, u]$. De gegeven informatie in de onderzoeksvraag wordt nu gemodelleerd als een eindige verzameling frequentie constraints. Een verzameling frequentie constraints $\mathcal{C}$ *impliceert* de frequentie constraint $\mathit{freq}(I) \in [l, u]$, als elke database die elke constraint in $\mathcal{C}$ waar maakt, eveneens $\mathit{freq}(I) \in [l, u]$ waar maakt. Dus: in elke situatie waarin $\mathcal{C}$ waar is, moet ook $\mathit{freq}(I) \in [l, u]$ waar zijn. Beschouw bijvoorbeeld de volgende verzameling frequentie constraints.

$$\mathcal{C} \;=\; \{\; \mathit{freq}(\{a\}) \in [0.8, 0.9] \;,\; \mathit{freq}(\{b\}) \in [0.6, 0.8] \;\} \;.$$

Omwille van het monotoniciteitsprincipe weten we dat de frequentie van $\{a, b\}$ nooit groter kan zijn dan de frequentie van $\{b\}$. Omdat de frequentie van $\{b\}$ maximaal 0.8 is in $\mathcal{C}$, kunnen we dus besluiten dat $\mathit{freq}(\{a, b\}) \in [0, 0.8]$ geïmpliceerd wordt door $\mathcal{C}$. Een andere belangrijke notie is die van *strikte* implicatie. Strikte implicatie drukt uit dat een interval $[l, u]$ het beste interval is dat we kunnen vinden voor een itemset I, gebaseerd op een verzameling frequentie constraints $\mathcal{C}$. Het beste interval betekent hier dat voor elk kleiner interval $[l', u']$, het niet langer waar is dat $\mathit{freq}(I) \in [l', u']$ geïmpliceerd wordt door $\mathcal{C}$. Beschouw opnieuw de verzameling frequentie constrains $\mathcal{C}$ die hierboven gegeven is. Ondanks het feit dat $\mathit{freq}(\{a, b\}) \in [0, 0.8]$ geïmpliceerd wordt door $\mathcal{C}$, is deze implicatie niet strikt. Aangezien ten minste een fractie 0.8 van de transacties a bevat, en een fractie van ten minste 0.6, b bevat, moet er minstens een overlap van 0.4 zijn tussen de transacties die a bevatten en de transacties die b bevatten. Daarom moet er minstens een fractie van 0.4

van de transacties de itemset $\{a, b\}$ bevatten. Dus, de frequentie van $\{a, b\}$ moet in het interval $[0.4, 0.8]$ vallen. We kunnen nu aantonen dat dit interval strikt is door twee databases $\mathcal{D}_1$ en $\mathcal{D}_2$ te geven, die beide $\mathcal{C}$ waarmaken, en tegelijkertijd $\mathit{freq}(\{a, b\}, \mathcal{D}_1) = 0.4$, en $\mathit{freq}(\{a, b\}, \mathcal{D}_2) = 0.8$ hebben. De volgende databases zijn hier voorbeelden van.

$\mathcal{D}_1 =$

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

$\mathcal{D}_2 =$

TID	Items
1	a, b
2	a, b
3	a, b
4	a, b
5	

Veronderstel nu bijvoorbeeld dat 0.4 geen strikte ondergrens zou zijn voor de frequentie van $\{a, b\}$. In dat geval zou er een getal l, strikt groter dan 0.4 bestaan, zodanig in elke database die $\mathcal{C}$ waarmaakt, de frequentie van $\{a, b\}$ minstens l is. Dit is echter in contradictie met $\mathit{freq}(\{a, b\}, \mathcal{D}_1) = 0.4$. Feitelijk is $\mathcal{D}_1$ een tegenvoorbeeld voor alle l strikt groter dan 0.4. We zullen databases zoals $\mathcal{D}_1$ en $\mathcal{D}_2$, *bewijs-databases* noemen. Deze bewijs-databases spelen een zeer belangrijke rol in de theorie die we ontwikkelen in de thesis.

Een centraal probleem dat we bestuderen in de thesis is het FREQSAT-probleem. FREQSAT is het volgende probleem: gegeven een verzameling frequentie constraints, bestaat er een database die deze alle constraints in deze verzameling gelijktijdig waar maakt? In de thesis tonen we aan dat dit probleem **NP**-compleet is.

Speciale gevallen Omwille van de hoge complexiteit van FREQSAT, is de bruikbaarheid in de praktijk beperkt. Omwille hiervan bestuderen we in de thesis speciale gevallen die een lagere complexiteit hebben, maar nog steeds interessant zijn vanuit een praktisch standpunt. De volgende gevallen worden in de thesis besproken.

- *Ondergrenzen.* Enkel ondergrenzen op de frequentie van de itemsets worden beschouwd; dat wil zeggen, we beschouwen enkel frequentie constraints van de vorm $\mathit{freq}(I) \in [l, 1]$. Een verzameling van zulke constraints wordt een *systeem van frequente sets* genoemd. Zulk een systeem wordt *compleet* genoemd indien alle informatie in het systeem strikt is. Voor zulke systemen van frequente sets is het antwoord op het FREQSAT-probleem steeds positief. We tonen aan dat compleetheid beslist kan worden in polynomiale tijd. We beschrijven eveneens drie axioma's, $\boldsymbol{\mathcal{F}_1}$, $\boldsymbol{\mathcal{F}_2}$, en $\boldsymbol{\mathcal{F}_3}$ voor complete systemen van frequente sets.

- *Bovengrenzen.* We laten enkel constraints van de vorm $freq(I) \in [0, u]$ toe. Opnieuw is FREQSAT steeds positief. Ondanks het feit dat dit geval heel sterk op het vorige lijkt, is het veel simpeler. Bijvoorbeeld: de compleetheid van het systeem kan beslist worden met logaritmische ruimte, en slechts twee eenvoudige axioma's, $\boldsymbol{\mathcal{IF}_1}$ en $\boldsymbol{\mathcal{IF}_2}$ zijn nodig.

- *Exacte frequenties, alle deelverzamelingen.* Dit is ongetwijfeld het meest interessante geval. Er worden enkel intervallen afgeleid voor de frequentie van itemsets waarvan we de frequentie van *alle* subsets *exact* weten. In dat geval kan de deductie van strikte grenzen in polynomiale tijd gebeuren. Dit geval is zeer interessant omdat de veronderstelde informatie exact de informatie is die we hebben in het Apriori-algoritme. Gebaseerd op deductie regels die we voor dit geval vinden, worden *afleidbare itemsets* gedefiniëerd. Een itemset I wordt *afleidbaar in database* $\mathcal{D}$ genoemd indien zijn frequentie uniek bepaald wordt door de frequenties van zijn deelverzamelingen. Er wordt een algoritme ontwikkeld om alle niet-afleidbare itemsets op een efficiënte manier te vinden.

We bestuderen voor elk geval de complexiteit van FREQSAT en een volledige axiomatisatie. We tonen ook voor elk geval hoe de frequentie-grenzen van de itemsets berekend kunnen worden.

Generische techniek We beschrijven een generische techniek die toelaat om in specifieke gevallen een volledige set axioma's te vinden. De methode die hiervoor gebruikt wordt, is gebaseerd op de eliminatietechniek voor lineaire stelsels ongelijkheden van *Fourier* en *Motzkin* [67]. In de thesis tonen we hoe een FREQSAT-probleem kan vertaald worden naar een systeem van lineaire ongelijkheden. In dit systeem elimineren we vervolgens enkele van de variabelen. Het resulterende systeem bevat dan de axiomatisatie. Bijvoorbeeld, veronderstel dat we grenzen willen berekenen op de frequentie van een itemset $\{b\}$, gebaseerd op de informatie dat $freq(\{a\}) = f_a$, and $freq(\{a, b\}) = f_{ab}$. Laat x_a staan voor de fractie van de transacties die gelijk zijn aan $\{a\}$, x_b die gelijk zijn aan b, en x_{ab} die gelijk zijn aan $\{a, b\}$. De (onbekende) frequentie van b wordt genoteerd met f_b. We vertalen deze situatie als het volgende stelsel lineaire ongelijkheden.

$$(x_a + x_{ab} = f_a) \ \wedge \ (x_{ab} = f_{ab}) \ \wedge \ (x_b + x_{ab} = f_b)$$
$$\wedge \ (x_a \geq 0) \ \wedge \ (x_b \geq 0) \ \wedge \ (x_{ab} \geq 0) \wedge \ (x_a + x_b + x_{ab} \leq 1) \ .$$

Vervolgens elimineren we in dit systeem de variabelen x_a, x_b en x_{ab}. Deze eliminatie resulteert in het volgende, equivalente systeem:

$$(0 \leq f_a) \ \wedge \ (f_a \leq 1) \ \wedge \ (0 \leq f_b) \ \wedge \ (f_b \leq 1) \ \wedge \ (0 \leq f_{ab}) \ \wedge \ (f_{ab} \leq 1)$$
$$\wedge \ (f_{ab} \leq f_a) \ \wedge \ (f_{ab} \leq f_b) \ \wedge \ (f_{ab} \geq f_a + f_b - 1) \ .$$

Dus, we kunnen afleiden dat de frequentie van $\{b\}$ in het interval

$$\left[\ \max\{0, f_{ab}\} \ , \ \min\{1, 1 + f_{ab} - f_a\} \ \right]$$

ligt. Dit interval is strikt.

Toepassing: Beknopte representaties Gebaseerd op de deductieregels kunnen we redundanties in the verzameling frequente itemsets identificeren. In het bijzonder het speciale geval met exacte frequenties van alle subsets is hier interessant. We tonen hoe deductie gebruikt kan worden om *beknopte representaties* [62] van de verzameling frequente itemsets te maken. Een beknopte representatie is in feite een deelverzameling van de verzameling frequente itemsets die nog steeds dezelfde frequentie informatie bevat. Dit wil zeggen: gebaseerd op de beknopte representatie moeten we in staat zijn om voor elke itemset te beslissen of hij frequent is of niet. Bovendien moeten we indien een itemset frequent is, ook zijn frequentie uit de representatie kunnen afleiden. Aangezien het doel van de deductie regels die we bestuderen is om frequenties zo exact mogelijk af te leiden, is er een duidelijk verband met beknopte representaties. Andere representaties in de literatuur zijn: *free sets* [9], *closed sets* [72, 8, 75] en *disjunction-free sets* [12]. We tonen hoe deze types beknopte representaties kunnen uitgedrukt worden in termen van de deductie regels die we bestuderen. Op deze manier vormt de benadering in de thesis die gebaseerd is op deductie regels een unificatie voor vele voorstellen in de literatuur.

Gerelateerd werk In artificiële intelligentie zijn probabilistische logica's reeds intensief bestudeerd [41, 71]. De link met deze thesis is dat we de frequentie van een itemset I kunnen beschouwen als de kans dat een willekeurig gekozen transactie uit de transactie database I bevat. Dit wil zeggen, we kunnen de transactie database beschouwen als een onderliggende kansruimte, en de itemsets als de conjunctie van atomen. In Hoofdstuk 7 van de thesis wordt werk in artificiële intelligentie dat aan het onderwerp van deze thesis gerelateerd is, besproken. In het bijzonder de verbanden met de *probabilistische logica* van *Nilsson* [68], de logica om over kansen te redeneren van *Fagin, Hailperin en Megiddo* [26] en het werk van *Lukasiewicz* [60] krijgen speciale aandacht.

Ook connecties met data mining worden besproken. Interessant hier is het MAXMINER algoritme van *Bayardo* [6], en het PASCAL algoritme van *Bastide e.a.* [5] over het zoeken van frequente itemsets. Deze twee algoritmes maken beiden gebruik van deductie van grenzen op de frequentie van itemsets.

Een ander belangrijk stuk gerelateerd werk betreft *beknopte representaties* [62]. Het werk over beknopte representaties in Hoofdstuk 6 wordt vergeleken met andere voorstellen zoals *free sets* [9], *closed sets* [72, 8, 75] en *disjunction-free sets* [12].

Ook connecties tussen de deductie regels in Hoofdstuk 4 en combinatoriek zoals *Bonferonni ongelijkheden* [7, 50, 31] en *statistische data bescherming* [24] komen aan bod.

Structuur van de thesis In Hoofdstuk 2 wordt een formele definitie van de problemen die bestudeerd worden gegeven. De speciale gevallen worden bestudeerd in Hoofdstuk 3 (Onder- en bovengrenzen) en Hoofdstuk 4 (Exacte frequenties). In Hoofdstuk 5 wordt de generische techniek, gebaseerd op het eliminatie algoritme van Fourier en Motzkin geïntroduceerd. Beknopte representaties worden besproken in Hoofdstuk 6 en gerelateerd werk in Hoofdstuk 7. De thesis wordt afgesloten in Hoofdstuk 8 met een samenvatting van de resultaten en interessante onderzoeksrichtingen voor toekomstig werk.