

On Inverse Frequent Set Mining

Taneli Mielikäinen
HIIT Basic Research Unit
Department of Computer Science
University of Helsinki, Finland
Taneli.Mielikainen@cs.Helsinki.fi

Abstract

Frequent set mining is a well-known technique to summarize binary data. However, it is an open problem how difficult it is to invert the frequent set mining, i.e., how difficult it is to find a binary data set that is compatible with frequent set mining results, the frequent sets. This inverse data mining problem is related to the questions of how well privacy is preserved in the frequent sets and how well the frequent sets characterize the original data set. In this paper we analyze the computational complexity of the problem of finding a binary data set compatible with a given collection of frequent sets and show that in many cases the problem is computationally very difficult.

1 Introduction

One of the most important data mining techniques to analyze data is finding interesting patterns from a given data set [20]. The most prominent example of finding interesting patterns from data is finding frequently occurring sets from binary data. This task is known as *frequent set mining*. The goal in the task is to find all sets that are contained in at least a given fraction of sets in a given sequence of sets, i.e., to find the *frequent sets*. The number of sets (in the given sequence of set) that contain a set X is called the *support* of X . (For more detailed definitions, see Section 2.)

The frequent sets and their supports can be considered to be a reasonable summary of the data set. Furthermore, there exist several (output)efficient methods to obtain the frequent sets from large data sets [2, 15].

As the frequent set mining is one of the most prominent tasks in data mining, it is natural to ask whether there exists a data set compatible with a given collection of sets (and their supports), i.e., whether there exists a data set that agree with the supports of the given set collection and that the supports of the other sets would be less than the smallest

support in the given set collection. We call this task the *inverse frequent set mining*.

The existence of a data set compatible with the given set collection of is usually highly desirable also as the frequent sets are typically assumed to be a (good) summary of a data set. Deciding whether there exists a compatible data set for a collection of frequent sets can be considered as a very harsh quality control and an efficient method answering that question could have also some practical impact to data summarization by frequent sets as several instances are not willing to share their data but may on the other hand deliver some summaries of the data.

The inverse frequent set mining could be a serious threat for privacy. There are some privacy problems in the frequent set mining [9, 23]. However, more severe dangers for the privacy lurk in the mining results: It is quite reasonable to assume that the mining of data, such as the frequent set mining, could be conducted by a trusted party. For example, in the most confidential situations (when money is not of central importance) the data mining can be done by the instance that has collected the data (or as a secure multi-party computation, see e.g. [1]). On the other hand, the mining results might be available also for possibly hostile instances. Thus our primary concern in defending the privacy in data mining is to ensure that the mining results do not reveal any secrets that are not intended to be concealed [10]. In this paper we require especially that the malicious parties should not be able to find (a good approximation of) the original data set from the mining results (in reasonable time). (It can be argued that this might be too weak notion of privacy in most cases.)

In this paper we study how well the frequent sets preserve privacy from the viewpoint of computational complexity. In particular, we analyze the computational complexity of the inverse frequent set mining problem. We show that deciding whether there is a data set compatible with the given frequent sets is NP -hard and computing the number of data sets compatible with the given frequent sets is $\#P$ -hard even in the case when the original data

set d consists of six sets, i.e., d is of form $d_1d_2d_3d_4d_5d_6$. Also, we show that some of their special cases are solvable in polynomial time but already very simple minimality requirements make some trivial special cases again NP -hard.

The rest of this paper is organized as follows. In Section 2 the problems of frequent set mining and inverse frequent set mining are defined. In Section 3 we study projections of data sets and relate them to collections of frequent sets. In Section 4 we analyze the complexity of inverse data mining based on the results about the projections. Section 5 concludes the paper.

2 Frequent Set Mining Problem and Its Inverse

The *frequent set mining problem* can be formulated as follows: given a finite set R , a sequence $d = d_1 \dots d_n$ of subsets of R , and a threshold value $\sigma \in [0, 1]$, find all sets that are contained in at least fraction σ of the sets d_i , $1 \leq i \leq n$, i.e., to find the collection

$$\mathcal{F}(\sigma, d) = \{X \subseteq R : \text{supp}(X, d) \geq \sigma n\}$$

of *frequent sets* where

$$\text{supp}(X, d) = |\{i : X \subseteq d_i, 1 \leq i \leq n\}|,$$

called the *support* of X in d . The collection $\{X \subseteq R : X \notin \mathcal{F}(\sigma, d)\} = 2^R \setminus \mathcal{F}(\sigma, d)$ is called the collection of *infrequent sets*.

A popular alternative for the support of a set X is its normalized version, the *frequency* of the set X , which is denoted by

$$fr(X, d) = \frac{|\{i : X \subseteq d_i, 1 \leq i \leq n\}|}{n} = \frac{\text{supp}(X, d)}{n}$$

Although the frequencies can be considered sometimes more digestible, a main difference between support and frequency is that the frequencies can be computed from the supports as, by definition, $\text{supp}(\emptyset, d) = n$, but the supports cannot be determined uniquely from the frequencies.

In this paper our primary concern is the *inverse frequent set mining problem* which can be formulated as follows: given a finite collection \mathcal{F} of finite sets $X \in \mathcal{F}$ and supports $\text{supp}(X, \mathcal{F})$ for them, find a data set d compatible with the set collection and the supports, i.e., find a sequence $d = d_1 \dots d_n$ of subsets of $\bigcup_{X \in \mathcal{F}} X$ such that $\text{supp}(X, \mathcal{F}) = \text{supp}(X, d)$ for all $X \in \mathcal{F}$. (In Section 3 we shall relax this formulation a bit since the compatible data set can be exponentially larger than the collection of its frequent sets.)

3 Frequent Sets and Projections

The projection of the data set d onto a set X is a sequence

$$pr(X, d) = (d_1 \cap X) \dots (d_n \cap X).$$

The collection of projections $pr(X, d)$ onto the sets $X \in \mathcal{F}$ is denoted by

$$pr(\mathcal{F}, d) = \{pr(X, d) : X \in \mathcal{F}\}.$$

Two projections $pr(X, d)$ and $pr(X, d')$ are considered to be equal if there is a permutation $\pi : [n] \rightarrow [n]$ such that $pr(X, d_i) = X \cap d_i = X \cap d'_{\pi(i)} = pr(X, d'_{\pi(i)})$ for all $1 \leq i \leq n$. The projections shall be used to give more approachable view to the inverse frequent set mining problem.

Another collection of sets of which we are interested in (in addition to the frequent sets) is the collection $\mathcal{M}(\sigma, d)$ of *maximal frequent sets*. It consists of those frequent sets that have no frequent supersets, i.e.,

$$\mathcal{M}(\sigma, d) = \{X \in \mathcal{F}(\sigma, d) : X \subset Y \Rightarrow Y \notin \mathcal{F}(\sigma, d)\}.$$

They can be mined even more efficiently than the frequent sets [3, 4, 5, 13, 14].

However, the reason why we are interested in maximal frequent sets is that the collection $\mathcal{F}(\sigma, d)$ with the supports of $X \in \mathcal{F}(\sigma, d)$ contains the same information as the projections $pr(\mathcal{M}(\sigma, d), d)$:

Theorem 1 *The frequent sets $\mathcal{F}(\sigma, d)$ and their supports can be computed from the projections $pr(\mathcal{M}(\sigma, d), d)$ and the projections $pr(\mathcal{M}(\sigma, d), d)$ can be computed from the frequent sets $\mathcal{F}(\sigma, d)$.*

Proof. For each $X \in \mathcal{F}(\sigma, d)$ and each $Y \supseteq X$ we have $\text{supp}(X, d) = |\{i : X \subseteq d_i\}| = |\{i : X \subseteq d_i \cap Y\}| = \text{supp}(X, pr(Y, d))$. By definition, each set $X \in \mathcal{F}(\sigma, d)$ is contained in some set $Y \in \mathcal{M}(\sigma, d)$. Thus the first claim holds.

Each projection $pr(X, d)$ onto a maximal set $X \in \mathcal{M}(\sigma, d)$ can be computed from the set collection $\mathcal{F} = \{Y \in \mathcal{F}(\sigma, d) : Y \subseteq X\}$ and the supports $\text{supp}(X, \mathcal{F}) = \text{supp}(X, d)$ as follows:

1. Find the set $\mathcal{M} = \{Y \in \mathcal{F} : Y \subset Z \Rightarrow Z \notin \mathcal{F}\}$. Halt if \mathcal{M} is empty.
2. For each $Y \in \mathcal{M}$, add $\text{supp}(Y, \mathcal{F})$ copies of Y to $pr(X, d)$ and decrease the support $\text{supp}(Z, \mathcal{F})$ of each subset of Y by $\text{supp}(Y, \mathcal{F})$.
3. Remove the sets $Y \in \mathcal{F}$ with $\text{supp}(Y, \mathcal{F}) = 0$. Go to step 1.

Moreover, the computations can be done in polynomial time in the cardinality of \mathcal{F} and in $|R|$. \square

Note that Theorem 1 also implies that if $\mathcal{F}(\sigma, d) = 2^R$ then the data set d can be reconstructed from the supports of the sets $X \in \mathcal{F}(\sigma, d)$ in polynomial time as $\mathcal{M}(\sigma, d) = \{R\}$. Also, sometimes the collection of frequent sets $\mathcal{F}(\sigma, d)$ determine (implicitly) the supports of the infrequent sets $2^R \setminus \mathcal{F}(\sigma, d)$. If the bounds are computable in reasonable time then we can determine the supports for the whole set collection 2^R , and thus the compatible data set can be found, too.

Let us denote the projections determined by the frequent set collection \mathcal{F} by $pr(\mathcal{M}, \mathcal{F})$. The number of different sets in $pr(\mathcal{M}, \mathcal{F})$ can be considerably smaller than the number of sets in \mathcal{F} . Thus the projections $pr(\mathcal{M}(\sigma, d), d)$ (represented as a list of different sets with number of times each of them occur) can be interpreted as a *condensed representation* of the frequent sets $\mathcal{F}(\sigma, d)$ [8, 21, 22].

As the projections are (seemingly) closer to the actual data set than the corresponding frequent sets, they can be used to make the inverse frequent set mining problem more comprehensible by an equivalent formulation of the problem: given projections $pr(\mathcal{M}, \mathcal{F})$ find a data set d such that $pr(\mathcal{M}, \mathcal{F}) = pr(\mathcal{M}, d)$. Let us call this the *data set reconstruction problem*.

However, there are sets of projections that cannot be realized as frequent set collections. Thus we should be able to ensure (in time polynomial in the combined size of the projections) that the set of projections can be realized as a frequent set collection. Fortunately, there are simple conditions necessary and sufficient to guarantee that there is frequent set collection for a given set of projections:

Theorem 2 *The projections $pr(X_1, \mathcal{F}_1), \dots, pr(X_m, \mathcal{F}_m)$ have the compatible collection \mathcal{F} of frequent sets, i.e., a collection \mathcal{F} of sets such that $supp(Y, pr(X_i, \mathcal{F}_i)) = supp(Y, \mathcal{F})$ for all $Y \subseteq X_i, 1 \leq i \leq m$, if and only if $pr(X_i \cap X_j, \mathcal{F}_i) = pr(X_i \cap X_j, \mathcal{F}_j)$ for all $1 \leq i, j \leq m$.*

Proof. If there is a frequent set collection \mathcal{F} such that $supp(Y, pr(X_i, \mathcal{F}_i)) = supp(Y, \mathcal{F})$ for all $Y \subseteq X_i, 1 \leq i \leq m$, then $pr(X_i \cap X_j, \mathcal{F}_i) = pr(X_i \cap X_j, \mathcal{F}_j)$ for all $1 \leq i, j \leq m$, because otherwise $pr(X_i \cap X_j, \mathcal{F}_i)$ and $pr(X_i \cap X_j, \mathcal{F}_j)$ would determine different supports for some $Y \subseteq X_i \cap X_j, 1 \leq i, j \leq m$.

If $pr(X_i \cap X_j, \mathcal{F}_i) = pr(X_i \cap X_j, \mathcal{F}_j)$ for all $1 \leq i, j \leq m$, then $supp(Y, pr(X_i \cap X_j, \mathcal{F}_i)) = supp(Y, pr(X_i \cap X_j, \mathcal{F}_j))$ for all $Y \subseteq X_i \cap X_j, 1 \leq i, j \leq m$. Thus the projections agree with the supports on all subsets of $X_i, 1 \leq i \leq m$. \square

The number sets in the data set d can be exponential in the number of frequent sets (and also in the combined size

of the frequent sets) as the set collection can consist of just one set R with support exponential in $|R|$.

This fact does not have to be a problem since most of our results are hardness results and it is reasonable to assume that if one is trying to reconstruct a data set then the size of the data set to be reconstructed is not considered to be infeasibly large.

Let us finally note that we consider the projections as sequences of sets instead of sets with number of occurrences for each set also because the minimum number of different sets in a compatible data set d can be exponentially larger than the number of frequent sets $\mathcal{F}(\sigma, d)$ (and thus exponentially larger than the number of different sets in the projections $pr(\mathcal{M}(\sigma, d), d)$). For example, let $\mathcal{F}(\sigma, d) = \{X \subseteq R : |X| \leq 2\}$ with supports $supp(X, d) = \binom{|R|-|X|}{\lfloor |R|/2 \rfloor - |X|}$. Then the sets of size two are disjunctive (for definition and properties of disjunctive sets, see e.g. [6, 19]). Due to the properties of disjunctive sets, the frequent sets $\mathcal{F}(\sigma, d)$ with $\sigma = \binom{|R|-2}{\lfloor |R|/2 \rfloor - 2} / \binom{|R|}{\lfloor |R|/2 \rfloor}$ determine the data set uniquely: the data set d is the sequence consisting of the size $\binom{|R|}{\lfloor |R|/2 \rfloor}$ subsets of R .

4 Inverse Frequent Set Mining

Our first hardness result on the inverse frequent set mining shows that the problem is quite difficult in general:

Theorem 3 *The problem of deciding whether there is a data set d that is compatible with the projections $pr(\mathcal{M}, \mathcal{F})$ is NP-complete even for a data set consisting of only six sets.*

Proof. The problem is clearly in NP since we can verify in time polynomial in $|\bigcup_{X \in \mathcal{M}} X|$ (and thus in $|\mathcal{M}|$, too) whether a certain data set d is compatible with the projections $pr(\mathcal{M}, \mathcal{F})$ simply by computing the projections $pr(\mathcal{M}, d)$.

We show the NP-hardness of the problem by reduction from the *graph 3-colorability problem*. The graph 3-colorability problem is, given a graph $G = (V, E)$, to decide whether there is a good 3-coloring $c : V \rightarrow \{r, g, b\}$, i.e., a 3-coloring c such that $c(i) \neq c(j)$ for all $\{i, j\} \in E$ [12].

Let the set R of attributes be $\{r_i, g_i, b_i : i \in V\}$. We construct the projections as follows: For each edge $\{i, j\} \in E$ we define a projection $pr(\{r_i, g_i, b_i, r_j, g_j, b_j\}, \mathcal{F})$ to be the sequence

$$\{r_i, g_j\} \{r_i, b_j\} \{g_i, r_j\} \{g_i, b_j\} \{b_i, r_j\} \{b_i, g_j\}.$$

If the graph is not 3-colorable then there is no data set d compatible with the projections: For every 3-coloring of

the graph, there is an edge $\{i, j\}$ with same colors associated to both vertices incident to the edge but none of the pairs $\{r_i, r_j\}$, $\{g_i, g_j\}$ and $\{b_i, b_j\}$ appear in the projection $pr(\{r_i, g_i, b_i, r_j, g_j, b_j\}, \mathcal{F})$. Thus there is not even a partial solution of one set compatible with the projections.

If the graph is 3-colorable then there is a data set d that is compatible with the projections: the six sets in the data set d are the six permutations of a 3-coloring c such that $c(i) \neq c(j)$ for all $\{i, j\} \in E$. \square

It would be desirable to be able to estimate how many compatible data sets there exist. For example, if the number of compatible data sets is huge then

- the summary might not be a serious privacy threat since without any other assumptions the risk of finding the correct compatible data set by mistake would be small, and
- the summary does not describe the data set very accurately.

The proof of Theorem 3 can also be adapted to give the hardness result for that problem:

Theorem 4 *The problem of computing the number of data sets d compatible with the projections $pr(\mathcal{M}, \mathcal{F})$ is $\#P$ -complete.*

Proof. The problem is in $\#P$ since its decision version is in NP .

Using the reduction described in the proof of Theorem 3, we can count the good 3-colorings: the number of the good 3-colorings is $1/6! = 1/720$ times the number of data sets compatible with the projections corresponding to the given graph. As counting the good 3-colorings is $\#P$ -hard, also counting the compatible data sets is $\#P$ -hard [12]. \square

Although the data set reconstruction problem is NP -complete in general, there are some special cases that can be solved in polynomial time. One of the most simplest cases is when there are only two projections (with arbitrary number of attributes):

Theorem 5 *It can be decided in polynomial time whether there is a data set d compatible with given projections $pr(X_1, \mathcal{F}_1)$ and $pr(X_2, \mathcal{F}_2)$ and the number of compatible data sets can be computed in polynomial time.*

Proof. By definition, the projection $pr(X_1, \mathcal{F}_1)$ is compatible with a data set d if and only if $pr(X_1, \mathcal{F}_1) = pr(X_1, d)$ and the projection $pr(X_2, \mathcal{F}_2)$ is compatible with a data set d if and only if $pr(X_2, \mathcal{F}_2) = pr(X_2, d)$. A data set d is compatible with both projections if and only if $pr(X_1 \cap X_2, \mathcal{F}_1) = pr(X_1 \cap X_2, d) = pr(X_1 \cap X_2, \mathcal{F}_2)$, $pr(X_1 \setminus X_2, \mathcal{F}_1) = pr(X_1 \setminus X_2, d)$

and $pr(X_2 \setminus X_1, \mathcal{F}_1) = pr(X_2 \setminus X_1, d)$. The data set d compatible with the projections $pr(X_1, d)$ and $pr(X_2, d)$ can be found by simply sorting the projections $pr(X_1, \mathcal{F}_1)$ and $pr(X_2, \mathcal{F}_2)$ by $pr(X_1 \cap X_2, \mathcal{F}_1)$ and $pr(X_1 \cap X_2, \mathcal{F}_2)$. This can be implemented in time $\mathcal{O}(|X_1 \cap X_2|n)$ from the projections $pr(X_1 \cap X_2, \mathcal{F}_1)$ and $pr(X_1 \cap X_2, \mathcal{F}_2)$ [18].

The number of data sets compatible with d can be computed in closed form from the counts of different sets in the projections $pr(X_1 \cap X_2, d)$, $pr(X_1, d)$ and $pr(X_2, d)$ [17]. \square

The practical relevance of the above positive result depends on how much the domains X_1 and X_2 overlap: if $|X_1 \cap X_2|$ is very small but $|X_1|$ and $|X_2|$ are large then there is great danger that there are several compatible data sets. Fortunately, in the case of two projections we are able to compute the number of compatible data sets and thus evaluate the usefulness of the found data set. The data set reconstruction problem turns out to be decidable in polynomial time also when the input consists of three projections.

Theorem 6 *It can be decided in polynomial time whether there is a data set d compatible with given projections $pr(X_1, \mathcal{F}_1)$, $pr(X_2, \mathcal{F}_2)$ and $pr(X_3, \mathcal{F}_3)$.*

Proof. First we construct a data set d' compatible with $pr(X_1, \mathcal{F}_1)$, $pr(X_2, \mathcal{F}_2)$ and $pr(X_3 \setminus (X_1 \setminus X_2), \mathcal{F}_3)$.

Based on the data set d' we construct a bipartite graph $G = (V_1, V_2, E)$ that is used to make d' compatible also with $pr(X_3 \cap X_1, \mathcal{F}_3)$. The vertices i in V_1 and V_2 ($V_1 = V_2$) correspond to sets $pr(X_3 \cap X_1, d'_i)$, $1 \leq i \leq n$. There is an edge $\{i, j\} \in E$ if and only if $pr(X_3 \cap X_1, d'_i) = pr(X_3 \cap X_1, d'_j)$ and either $pr(X_1 \cap X_2, d'_i) = pr(X_1 \cap X_2, d'_j)$ or $pr(X_2 \cap X_3, d'_i) = pr(X_2 \cap X_3, d'_j)$.

There is a data set d compatible with the projections $pr(X_1, \mathcal{F}_1)$, $pr(X_2, \mathcal{F}_2)$ and $pr(X_3, \mathcal{F}_3)$ if and only if there is a perfect bipartite matching in the corresponding bipartite graph G : The bipartite matchings in G correspond to all data sets compatible with projections $pr(X_1 \cap X_2, d')$ and $pr(X_2 \cap X_3, d')$. Perfect matchings in G correspond exactly those data sets that are compatible $pr(X_1, \mathcal{F}_1) = pr(X_1, d')$, $pr(X_2, \mathcal{F}_2) = pr(X_2, d')$ and $pr(X_3, \mathcal{F}_3)$. A perfect matching in a bipartite graph $G = (V, E)$ can be found in time $\mathcal{O}(\sqrt{|V|}|E|)$ [11]. \square

It is not clear how the approach sketched in the above proof could be generalized to the case of more than three projections.

In the simplest case of the data set reconstruction problem all projections $pr(X_1, \mathcal{F}_1), \dots, pr(X_m, \mathcal{F}_m)$ are disjoint, i.e., $X_i \cap X_j = \emptyset$ since in that case any data set with

projections $pr(X_1, \mathcal{F}_1), \dots, pr(X_m, \mathcal{F}_m)$ is a compatible one. Unfortunately this also means that the number of compatible data sets in this case is very large. Thus one should probably require something more than mere compatibility.

One natural restriction, applying the Occam's razor, is to search for the compatible data set with the smallest number of different sets: this kind of data set is (in some sense) the simplest hypothesis based on the frequent set collection which is beneficial for both analyzing the data set and actioning using the data set. It can be shown that finding the data set with the smallest number of different sets is *NP*-hard for already two projections:

Theorem 7 *It is NP-hard to find the data set d compatible with projections $pr(X_1, \mathcal{F})$ and $pr(X_2, \mathcal{F})$ such that $X_1 \cap X_2 = \emptyset$ having the smallest number of different sets.*

Proof. We show the *NP*-hardness by reduction from the 3-partition problem which is, given a set A of $3l$ elements, a bound $B \in \mathbb{N}$, and a size $s(a) \in \mathbb{N}, B/4 < s(a) < B/2$, for each $a \in A$ such that $\sum_{a \in A} s(a) = lB$, decide whether or not A can be partitioned into l disjoint sets A_1, A_2, \dots, A_m such that, for $1 \leq i \leq l, \sum_{a \in A_i} s(a) = B$ [12]. As the 3-partition problem is strongly *NP*-complete, we can assume that the sizes $s(a), a \in A$, are bounded above polynomially by l .

The instance (A, B, s) of 3-partition is coded into two projections as follows. W.l.o.g., let $A = [3l]$. Then $X_1 = [\lceil \log l \rceil]$ and $X_2 = \lceil \log 3l \rceil + \lceil \log l \rceil$. Projection $pr(X_1, \mathcal{F})$ consists of $s(a)$ copies of each set $X(a) \subseteq X_1$ corresponding to binary code of the natural number a for each $a \in A = [3l]$. Projection $pr(X_2, \mathcal{F})$ consists of B copies of each set $X(b) \subseteq X_2$ corresponding to binary code of the natural number b for each $b \in [l]$.

Clearly, there is a 3-partition for (A, B, s) if and only if there is a compatible data set with $3l$ different sets in the both projections $pr(X_1, \mathcal{F})$ and $pr(X_2, \mathcal{F})$. \square

5 Conclusions

In this paper we analyzed the computational complexity of inverse frequent set mining, i.e., the problem of finding a data set compatible with a collection of frequent sets. We showed that in many cases the problem are computationally difficult (i.e., *NP*-hard and $\#P$ -hard).

The results are mostly negative for the users of precomputed collections of frequent sets: the user cannot count the number of different data sets compatible with the given collection of frequent sets and not even to decide if there are any compatible data sets. From the viewpoint of privacy preservation the results are somewhat positive as they say that delivering frequent sets might not cause serious threat to privacy (if the privacy is understood as not revealing the

original data set) as the inverse frequent set mining is computationally difficult.

There are still several important open problems related to inverse frequent set mining and to inverse data mining in general:

- Can the inverse frequent set mining problem be solved in polynomial for any fixed number of maximal sets in the set collection? If the problem can be solved in polynomial time, can the number of compatible data sets be computed in that case, too?
- What is the average case complexity of inverse frequent mining? Also, what is the average case? (Some suggestions on the average case are given in [24].)
- Defining a good search space for the inverse frequent set mining problem seems to be quite challenging. Are there reasonably efficient search strategies to find a compatible data set in practice?
- If the frequent sets are mined then the data miner can often have also the data set. (If there is a positive answer to the previous question then also the user of the frequent sets might be able to obtain a compatible data set.) Can the number of compatible data sets be approximated in practice (e.g., using Markov chains [16]) when one compatible data set is given?
- Can the search for compatible data set be guided by lower and upper bounds for the infrequent sets (computed e.g. by the NDI rules, see [7])?
- What are the approximability properties of the inverse frequent set mining problem? For example, is it possible to find a data set that is almost compatible with the set collection (and what being almost compatible means)?
- How similar the compatible data sets can be for a fixed collection of frequent sets?
- What kind of data summarization can be achieved without telling practically anything about the actual data set, e.g., are there zero-knowledge proofs that would ensure privacy preservation but that could still be useful in data analysis?
- From which summaries it is difficult to infer a good approximation of the original data set?

Acknowledgments. I wish to thank Floris Geerts for inspiring conversations and valuable suggestions on the manuscript.

References

- [1] R. Agrawal, A. Evfimievski, and R. Srikant. Information sharing across private databases. In *Proceedings of the Twenty-Second ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 284–295. ACM, 2003.
- [2] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo. Fast discovery of association rules. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, chapter 12, pages 307–328. AAAI/MIT Press, 1996.
- [3] R. J. Bayardo Jr. Efficiently mining long patterns from databases. In A. T. Laura M. Haas, editor, *SIGMOD 1998, Proceedings ACM SIGMOD International Conference on Management of Data*, pages 85–93. ACM, 1998.
- [4] E. Boros, V. Gurvich, L. Khachiyan, and K. Makino. On the complexity of generating maximal frequent and minimal infrequent sets. In H. Alt and A. Ferreira, editors, *STACS 2002*, volume 2285 of *Lecture Notes in Computer Science*, pages 133–141. Springer-Verlag, 2002.
- [5] D. Burdick, M. Calimlim, and J. Gehrke. MAFIA: A maximal frequent itemset algorithm for transactional databases. In *Proceedings of the 17th International Conference of Data Engineering (ICDE'01)*, pages 443–452, 2001.
- [6] A. Bykowski and C. Rigotti. A condensed representation to find frequent patterns. In *Proceedings of the Twentieth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*. ACM, 2001.
- [7] T. Calders and B. Goethals. Mining all non-derivable frequent itemsets. In T. Elomaa, H. Mannila, and H. Toivonen, editors, *Principles of Data Mining and Knowledge Discovery*, volume 2431 of *Lecture Notes in Artificial Intelligence*, pages 74–865. Springer-Verlag, 2002.
- [8] T. Calders and B. Goethals. Minimal k -free representations of frequent sets. In N. Lavrac, D. Gamberger, L. Todorovski, and H. Blockeel, editors, *Knowledge Discovery in Databases: PKDD 2003*, volume 2838 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 2003.
- [9] A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 217–228. ACM, 2002.
- [10] C. Farkas and S. Jajodia. The inference problem: A survey. *SIGKDD Explorations*, 4(2):6–11, 2002.
- [11] Z. Galil. Efficient algorithms for finding maximum matchings in graphs. *ACM Computing Surveys*, 18(1):23–38, 1986.
- [12] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, 1979.
- [13] K. Gouda and M. J. Zaki. Efficiently mining maximal frequent itemsets. In N. Cercone, T. Y. Lin, and X. Wu, editors, *Proceedings of the 2001 IEEE International Conference on Data Mining*, pages 163–170. IEEE Computer Society, 2001.
- [14] D. Gunopulos, R. Khardon, H. Mannila, S. Saluja, H. Toivonen, and R. S. Sharma. Discovering all most specific sentences. *ACM Transactions on Database Systems*, 28(2):140–174, 2003.
- [15] J. Hipp and U. Güntzler. Is pushing constraints deeply into the mining algorithms really what we want? *SIGKDD Explorations*, 4(1):50–55, 2002.
- [16] M. Jerrum. Mathematical foundations of the Markov Chain Monte Carlo method. In M. Habib, C. McDiarmid, J. Ramirez-Alfonsin, and B. Reed, editors, *Probabilistic Methods for Algorithmic Discrete Mathematics*, volume 16 of *Algorithms and Combinatorics*, pages 116–165. Springer-Verlag, 1998.
- [17] S. Jukna. *Extremal Combinatorics: With Applications in Computer Science*. EATCS Texts in Theoretical Computer Science. Springer-Verlag, 2001.
- [18] D. E. Knuth. *Sorting and Searching*, volume 3 of *The Art of Computer Programming*. Addison-Wesley, second edition, 1998.
- [19] M. Kryszkiewicz. Concise representation of frequent patterns based on disjunction-free generators. In N. Cercone, T. Y. Lin, and X. Wu, editors, *Proceedings of the 2001 IEEE International Conference on Data Mining*, pages 305–312. IEEE Computer Society, 2001.
- [20] H. Mannila. Local and global methods in data mining: Basic techniques and open problems. In P. Widmayer, F. Triguero, R. Morales, M. Hennessy, S. Eidenbenz, and R. Conejo, editors, *Automata, Languages and Programming*, volume 2380 of *Lecture Notes in Computer Science*, pages 57–68. Springer-Verlag, 2002.
- [21] H. Mannila and H. Toivonen. Multiple uses of frequent sets and condensed representations. In E. Simoudis, J. Han, and U. M. Fayyad, editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, pages 189–194. AAAI Press, 1996.
- [22] T. Mielikäinen. Finding all occurring sets of interest. In J.-F. Boulicaut and S. Džeroski, editors, *2nd International Workshop on Knowledge Discovery in Inductive Databases*, pages 97–106, 2003.
- [23] S. R. M. Oliveira and O. R. Zaiane. Privacy preserving frequent itemset mining. In C. Clifton and V. Estivill-Castro, editors, *IEEE Workshop on Privacy, Security, and Data Mining*, volume 14 of *Conferences in Research and Practice in Information Technology*, 2002.
- [24] G. Ramesh, W. A. Maniatty, and M. J. Zaki. Feasible itemset distributions in data mining: Theory and application. In *Proceedings of the Twenty-Second ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 284–295. ACM, 2003.