Space Lower Bounds for Itemset Frequency Sketches*

Edo Liberty[†] Michael Mitzenmacher[‡] Justin Thaler[§]

Abstract

Given a database, computing the fraction of rows that contain a query itemset or determining whether this fraction is above some threshold are fundamental operations in data mining. A uniform sample of rows is a good sketch of the database in the sense that all sufficiently frequent itemsets and their approximate frequencies are recoverable from the sample, and the sketch size is independent of the number of rows in the original database. For many seemingly similar problems there are better sketching algorithms than uniform sampling. In this paper we show that for itemset frequency sketching this is not the case. That is, we prove that there exist classes of databases for which uniform sampling is nearly space optimal.

[*] The full version of this paper is attached as an appendix

[†] Yahoo! Labs

[‡] Harvard University, School of Engineering and Applied Sciences. Supported in part by NSF grants CCF-1320231, CNS-1228598, IIS-0964473, and CCF-0915922. Part of this work was done while visiting Microsoft Research, New England.

[§] Yahoo! Labs. Parts of this work were performed while the author was a Research Fellow at the Simons Institute for the Theory of Computing, UC Berkeley. Supported by a Research Fellowship from the Simons Institute for the Theory of Computing.

1 Introduction

Identifying frequent itemsets is one of the most basic and well-studied problems in data mining. Formally, we are given a binary database $\mathcal{D} \in \left(\{0,1\}^d\right)^n$ consisting of n rows and d columns, or attributes.[1] An *itemset* $T \subseteq [d]$ is a subset of the attributes, and the *frequency* f_T of T is the fraction of rows of $\mathcal{D}$ that have a 1 in all columns of T.

Computing itemset frequencies is a central primitive that can be used for example for the following problems (and countless others): given a large corpus of text files, compute the number of documents containing a specific search query; given user records, compute the fraction of users who belong to a specific demographic; given event logs, compute sets of events that are observed together; given shopping cart data, identify bundles of items that are frequently bought together.

In many settings, an approximation of f_T, as opposed to an exact result, suffices. Alternatively, in some settings it suffices to recover a single bit indicating whether or not $f_T \geq \epsilon$ for some user defined threshold ϵ; such frequent itemsets may require additional study or processing. It is easy to show that uniformly sampling $\operatorname{poly}(d/\epsilon)$ rows from $\mathcal{D}$ and computing the approximate frequencies on the sample $\mathcal{S}(\mathcal{D})$ provides good approximations to f_T up to additive error ϵ. Our main contribution is to provide lower bounds establishing that uniform sampling is nearly optimal in terms of the space/accuracy tradeoff for many parameter regimes. Note that in general a sketch is not limited to containing a subset of the database rows. Our lower bounds hold for any summary data structure and recovery algorithm that constitute a valid sketch.

1.1 Motivation

1.1.1 The Case Against Computing Frequent Itemsets Exactly

If the task is only to identify frequent itemsets ($f_T \geq \epsilon$ for some ϵ), it is natural to ask whether we can compute all ϵ-frequent itemsets and store only those. Assuming that only a small fraction of itemsets are ϵ-frequent, this will result in significant space saving relative to naive solutions. The extensive literature on exact mining of frequent itemsets dates back to work of Agrawal et al. [AIS93], whose motivation stemmed from the field of market basket analysis. As the search space for frequent itemsets is exponentially large (i.e., size 2^d), substantial effort was devoted to developing algorithms that rapidly prune the search space and minimize the number of scans through the database. While the algorithms developed in this literature offer substantial concrete speedups over naive approaches to frequent itemset mining, they may still take time $2^{\Omega(d)}$, simply because there may be these many frequent itemsets. For example, if there is a frequent itemset of cardinality $d/10$, each of its $2^{d/10}$ subsets is also frequent. Motivated by this simple observation, there is now an extensive literature on condensed or non-redundant representations of exact frequent itemsets. Reporting only *maximal* and *closed* frequent itemsets often works well in practice, but it still requires exponential size in the worst case (see the survey [CG07]).

Irrespective of space complexity, the above methods face computational challenges. Yang [Yan04] determined that counting all frequent itemsets is #P-complete, and a bottleneck for enumeration is that the number of frequent itemsets can be exponentially large. Hamilton et al. [HCW06] provide further hardness results based on parametrized complexity. Here we observe that finding even a single frequent itemset of approximately maximal size is NP-hard. (The authors of [LLSW05] noticed this connection as well but did not mention approximation-hardness.)

Consider the bipartite graph containing n nodes (rows) on one side and d nodes (attributes) on the other. An edge exists between the two sides if and only if the row contains the attribute with value 1. Assume there exists a frequent itemset of cardinality ϵn and frequency ϵ. This itemset induces a balanced complete

[1] Throughout, we use the terms *attributes* and *items* interchangeably. While in many applications, attributes may be non-binary, any attribute with m possible values can be decomposed into $2\lceil \log m \rceil$ binary attributes, using two binary attributes to mark whether the value is 0 or 1 in the ith bit. We therefore focus on the binary case.

bipartite subgraph with ϵn nodes on both sides. Likewise, any balanced complete bipartite subgraph with ϵn nodes on each side implies the existence of an itemset of cardinality ϵn and frequency ϵ. Finding the maximal balanced complete bipartite subgraph is NP-hard, and even approximating it requires super polynomial time assuming that SAT does not admit subexponential time algorithms [FK04]. It follows that finding an itemset of approximately maximal frequency requires superpolynomial time under the same assumption.

1.1.2 The Case for Itemset Sketches

Determining the smallest possible size of itemset sketches is of interest in several data analysis settings.

Interactive Knowledge Discovery. Knowledge discovery in databases is often an interactive process: an analyst poses a sequence of queries to the dataset, with later queries depending on the answers to earlier ones [MT96]. For large databases, it may be inefficient or even infeasible to reread the entire dataset every time a query is posed. Instead, a user can keep around an itemset sketch only; this sketch will be much smaller than the original database, while still providing fast and accurate answers to itemset frequency queries.

Efficient Data Release. Itemset oracles capture a central problem in *data release*. In this setting, a data curator (such as a government agency like the US Census Bureau) wants to make a dataset publicly available. Due to their utility and ease interpretation, the data format of choice in these settings is typically *marginal contingency tables* (marginal tables for short). Here, for any itemset $T \subseteq d$ with $|T| = k$, the marginal table corresponding to T has 2^k entries, one for each possible setting of the attributes in T; each entry counts how many rows in the database are consistent with the corresponding setting of the k attributes. Notice that marginal tables are essentially just a list of itemset frequencies for $\mathcal{D}$.[2]

However, marginal tables can be extremely large (as any k-attribute marginal table has 2^k entries), and each released table may be downloaded by an enormous number of users. Rather than releasing marginal tables in their entirety, the data curator can instead choose to release an itemset summary. This summary can be much smaller than any single k-attribute marginal table, while still permitting any user to obtain fast and accurate estimates for the frequency of any k-attribute marginal query.

Mitigating Runtime Bottlenecks. While the use of itemset sketches cannot circumvent the hardness results discussed in Section 1.1.1, in many settings the empirical runtime bottleneck is the number of scans through the database, rather than the total runtime of the algorithm. The use of itemset sketches eliminates the need for the user to repeatedly scan or even keep a copy of the original database – the user can instead run a computationally intensive algorithm on the *sketch* to solve (natural approximation variants) of the hard decision or search problems. Indeed, there has been considerable work in the data mining community devoted to bounding the magnitude of errors that build up as a result of using approximate itemset frequency information when performing more complicated data mining tasks, such as rule identification [MT96].

1.2 Other Prior Work

The idea of producing condensed representations of approximate frequent itemsets is not new. Most relevant to our work, an influential paper by Mannila and Toivonen defined the notion of an *ϵ-adequate representation* of any class of queries [MT96]. Our Itemset-Frequency-Estimator sketching task essentially asks for an ϵ-adequate representation for the class of all itemset frequency queries. Mannila and Toivonen analyzed the magnitude of errors that build up when using ϵ-adequate representations to perform more complicated data mining tasks, such as rule identification. Subsequent work by Boulicaut et al. [BBR03] presented algorithms yielding ϵ-adequate representations for the class of all itemset queries, while Pei et al. [PDZH04] gave algorithms for approximating the frequency of all *frequent* itemsets to error ϵ. Unlike the trivial algorithms that we describe in Section 2, the algorithms presented in [MT96, BBR03, PDZH04] take exponential time in the worst case, and do not come with worst-case guarantees on the size of the output.

[2] More precisely, itemset frequency queries are equivalent to *monotone conjunction* queries on a database [BCD+07a, HRS12, KRSU10a], while marginal tables are equivalent to general (non-monotone) conjunction queries.

Streaming algorithms for both exact and approximate variants of frequent itemset mining have also been extensively studied — see the survey [CKN08]. However, to the best of our knowledge, there has been no work establishing lower bounds on the space complexity of streaming algorithms for identifying approximate frequent itemsets that are better than the lower bounds that hold for the much simpler *approximate frequent items* problem (a.k.a. the heavy hitters problem). Note that the lower bounds that we establish in this work apply even to summaries computed by non-streaming algorithms.

Itemset sketches have also received intense attention in the context of *differentially private data release*. In the terminology of this body of literature, an itemset sketch is equivalent to a database summary that accurately answers all (monotone) conjunction queries [TUV12, CTUW14, BCD+07b, KRSU10b, GHRU13]. The optimal size of *differentially private* itemset sketches is now understood up to polylogarithmic factors, though the fastest known algorithms for achieving the information-theoretic optimum run in exponential time [BUV14]. Our work is orthogonal to this body of literature, as we do not consider privacy constraints.

1.3 Notation and Problem Statements

Throughout, $\mathcal{D} \in \left(\{0,1\}^d\right)^n$ will denote a binary database consisting of n rows and d columns, or attributes. We denote the set $\{1, \ldots, d\}$ by $[d]$. An *itemset* $T \subseteq [d]$ is a subset of the attributes; abusing notation, we also use T to refer to the *indicator vector* in $\{0,1\}^d$ whose ith entry is 1 if and only if $i \in T$. We refer to any itemset T with $|T| = k$ as a k*-itemset*. The ith row of $\mathcal{D}$ will be denoted by $\mathcal{D}(i)$, and the jth entry of the ith row will be denoted by $\mathcal{D}(i,j)$. We say that a row *contains* an itemset T if the row has a 1 in all columns in T. The *frequency* $f_T(\mathcal{D})$ of T is the fraction of rows of $\mathcal{D}$ that contain T. Alternatively, $f_T(\mathcal{D}) = \frac{1}{n}\sum_{i=1}^n \mathbb{I}_{\{T \subset \mathcal{D}(i)\}}$. We use the simplified notation f_T instead of $f_T(\mathcal{D})$ when the meaning is clear.

We consider *four* different sketching problems that each capture a natural notion of approximate itemset frequency analysis. The first two problems (Definitions 1 and 2) require sketches from which it is possible (with probability $1-\delta$) to simultaneously recover accurate frequency estimates for *all* k-itemsets. The latter two problems (Definitions 3 and 4) are analogous, but with a weaker requirement: they only require sketches from which it is possible to obtain an accurate estimate for any (single) k-itemset with probability $1-\delta$ (but it may be very unlikely that one can recover accurate estimates for all k-itemsets from the sketch simultaneously). We refer to these latter two variants as *single-query* sketching problems.

Definition 1 (Itemset-Frequency-Indicator sketches)**.** *An Itemset-Frequency-Indicator sketch is a tuple* $(\mathcal{S}, \mathcal{Q})$*. The first term* $\mathcal{S}$ *is a randomized sketching algorithm. It takes as input a database* $\mathcal{D} \in \left(\{0,1\}^d\right)^n$*, a precision* ϵ*, an itemset size* k*, and a failure probability* δ*. It outputs a* summary $\mathcal{S}(\mathcal{D}, k, \epsilon, \delta) \in \{0,1\}^s$ *where* s *is the size of the sketch in bits. The second term is a deterministic query procedure* $\mathcal{Q} : \{0,1\}^s \times \{0,1\}^d \to \{0,1\}$*. It takes as input a summary* $\mathcal{S}$ *and a* k*-itemset* T *and outputs a single bit indicating whether* T *is frequent in* $\mathcal{D}$ *or not. More precisely, for a triple of input parameters* (k, ϵ, δ)*, the following two conditions must hold with probability* $1-\delta$ *over the randomness of the sketching algorithm* $\mathcal{S}$*, for every database* $\mathcal{D}$*:*

$$\forall \text{ } k\text{-itemsets } T \text{ s.t. } f_T > \epsilon, \quad \mathcal{Q}(\mathcal{S}(\mathcal{D}, k, \epsilon, \delta), T) = 1, \text{ and} \tag{1}$$

$$\forall \text{ } k\text{-itemsets } T \text{ s.t. } f_T < \epsilon/2, \quad \mathcal{Q}(\mathcal{S}(\mathcal{D}, k, \epsilon, \delta), T) = 0. \tag{2}$$

Note that if $\epsilon/2 \le f_T \le \epsilon$ *then either bit value can be returned.*

Definition 2 (Itemset-Frequency-Estimator sketches)**.** *An Itemset-Frequency-Estimator sketch is a tuple* $(\mathcal{S}, \mathcal{Q})$*. Here* $\mathcal{S}$ *is defined as above but* $\mathcal{Q} : \{0,1\}^s \times \{0,1\}^d \to [0,1]$ *outputs an approximate frequency. To be precise, the pair* $(\mathcal{S}, \mathcal{Q})$ *is a valid Itemset-Frequency-Estimator sketch for a triple of input parameters* (k, ϵ, δ) *if for every database* $\mathcal{D}$*:*

$$\Pr[\forall \text{ } k\text{-itemsets } T, \quad |\mathcal{Q}(\mathcal{S}(\mathcal{D}, k, \epsilon, \delta), T) - f_T| \le \epsilon] \ge 1 - \delta. \tag{3}$$

Definition 3 (Single-Query Itemset-Frequency-Indicator sketches)**.** *A Single-Query Itemset-Frequency-Indicator sketch is identical to an Itemset-Frequency-Indicator sketch, except that Equations* (1) *and* (2) *are replaced with the requirement that for every database $\mathcal{D}$ and any (single) k-itemset T:*

$$\textit{If } f_T > \epsilon, \textit{ then } \mathcal{Q}(\mathcal{S}(\mathcal{D},k,\epsilon,\delta),T) = 1 \textit{ with probability at least } 1-\delta, \textit{ and}$$

$$\textit{If } f_T < \epsilon/2, \textit{ then } \mathcal{Q}(\mathcal{S}(\mathcal{D},k,\epsilon,\delta),T) = 0 \textit{ with probability at least } 1-\delta.$$

Definition 4 (Single-Query Itemset-Frequency-Estimator sketches)**.** *A Single-Query Itemset-Frequency-Estimator sketch is identical to an Itemset-Frequency-Estimator sketch, except that Equation* (3) *is replaced with the requirement that for every database $\mathcal{D}$ and any (single) k-itemset T:* $\Pr[|\mathcal{Q}(\mathcal{S}(\mathcal{D},k,\epsilon,\delta),T) - f_T| \leq \epsilon] \geq 1-\delta$.

Definition 5. *The space complexity of a sketch, denoted by $|\mathcal{S}(n,d,k,\epsilon,\delta)|$, is the maximum sketch size generated by $\mathcal{S}$ for any database with n rows and d columns. That is, $|\mathcal{S}(n,d,k,\epsilon,\delta)| = \max_{\mathcal{D}\in\left(\{0,1\}^d\right)^n} |\mathcal{S}(\mathcal{D},k,\epsilon,\delta)|$.*

For brevity, we typically omit the parameters (n,d,k,ϵ,δ) when the meaning is clear, and simply write $|\mathcal{S}|$ to denote the space complexity of a sketch.

2 Naïve upper bounds

In the following we describe three trivial sketching algorithms.

Definition 6 (RELEASE–DB)**.** *This algorithm simply releases the database verbatim. In other words, the function $\mathcal{S}$ is the identity and $\mathcal{Q}$ is a standard database query.*

The space complexity of RELEASE–DB is clearly $|\mathcal{S}| = O(nd)$ and it produces exact estimates for both Itemset-Frequency-Estimator and Itemset-Frequency-Indicator sketches and their single-query analogs.

Definition 7 (RELEASE–ANSWERS)**.** *This algorithm computes and stores the results to all possible queries.*

Since there are $\binom{d}{k}$ possible k-itemset queries, the space complexity of RELEASE–ANSWERS is $|\mathcal{S}| = O(\binom{d}{k})$ for Itemset-Frequency-Indicator sketches and their single-query analogs, and $|\mathcal{S}| = O\left(\binom{d}{k}\log(1/\epsilon)\right)$ for Itemset-Frequency-Estimator sketches and their single-query analogs. The extra $\log(1/\epsilon)$ factor is needed to represent frequencies as floating point numbers up to precision ϵ.

Definition 8 (SUBSAMPLE)**.** *This algorithm samples rows uniformly at random with replacement from the database. The samples constitute the sketch $\mathcal{S}(\mathcal{D},k,\epsilon,\delta)$. The recovery algorithm $\mathcal{Q}(\mathcal{S}(\mathcal{D}),T)$ returns the frequency of T in the sampled rows via a standard database query.*

SUBSAMPLE produces a valid Itemset-Frequency-Indicator sketch of space complexity of $|\mathcal{S}| = O\left(\epsilon^{-1} d \log\left(\binom{d}{k}/\delta\right)\right)$, an Itemset-Frequency-Estimator sketch of space complexity $|\mathcal{S}| = O\left(\epsilon^{-2} d \log\left(\binom{d}{k}/\delta\right)\right)$, a Single-Query Itemset-Frequency-Indicator sketch of space complexity $|\mathcal{S}| = O\left(\epsilon^{-1}\cdot\log(1/\delta)\cdot d\right)$, and a Single-Query Itemset-Frequency-Estimator sketch of space complexity $|\mathcal{S}| = O\left(\epsilon^{-2}\cdot\log(1/\delta)\cdot d\right)$. To see this, note that the number of required bits is d (to describe one database row) times a sufficient number of sampled rows, which is $O\left(\epsilon^{-1}\log\left(\binom{d}{k}/\delta\right)\right)$ for Itemset-Frequency-Indicator sketches, $O\left(\epsilon^{-2}\log\left(\binom{d}{k}/\delta\right)\right)$ for Itemset-Frequency-Estimator sketches, $O\left(\epsilon^{-1}\cdot\log\left(1/\delta\right)\right)$ for Single-Query Itemset-Frequency-Indicator sketches, and $O\left(\epsilon^{-2}\log\left(1/\delta\right)\right)$ for Single-Query Itemset-Frequency-Estimator sketches. This follows from a standard application of Chernoff bounds, followed by a union bound of over all $\binom{d}{k}$ possible k-itemsets in the case of Itemset-Frequency-Indicator and Itemset-Frequency-Estimator sketches.

For any setting of the parameters (n,d,k,ϵ,δ), the minimal space usage among the above three trivial algorithms constitutes our naïve upper bound for all four sketching problems that we consider (in the full version, we formalize these upper bounds in a theorem whose statement we omit from this abstract).

3 Lower Bounds

In this section, we turn to proving lower bounds on the size of Itemset-Frequency-Indicator and Itemset-Frequency-Estimator sketches. Notice that the algorithms RELEASE–ANSWERS and SUBSAMPLE produce sketches whose size is independent of n; hence, it is impossible to prove lower bounds that grow with n. Consequently, we state our lower bounds in terms of the parameters $(d, k, 1/\epsilon)$, with all of our lower bounds holding as long as n is sufficiently large relative to these three parameters. This parameter regime — with n a sufficiently large polynomial in d, k, and $1/\epsilon$ — is consistent with typical usage scenarios, where the number of rows in a database far exceeds the number of attributes. In our theorem statements, we make explicit precisely how large a polynomial n must be in terms of d, k, and $1/\epsilon$ for the lower bound to hold.

Each of our lower bounds also requires d, k, and $1/\epsilon$ to satisfy certain mild technical relationships with each other — for example, Theorems 13 and 14 requires that $1/\epsilon < \binom{d/2}{k-1}$. In several cases, the assumed technical relationship between the parameters is necessary for the claimed lower bound to hold. For instance, the $\Omega(d/\epsilon)$ lower bound of Theorems 13 and 14 is false for $1/\epsilon \gg \binom{d/2}{k-1}$, as the algorithm RELEASE–ANSWERS would output a sketch of size $o(d/\epsilon)$ in this parameter regime.

3.1 Overview of the Lower Bounds

We now provide a high-level overview of the lower bounds we prove, and place our results in context. Throughout this section, we assume that the failure probability $\delta < 1$ of the sketching algorithm is a constant.

First lower bounds for Itemset-Frequency-Indicator sketches and their single-query analogs. We begin with a relatively simple bound for Itemset-Frequency-Indicator sketches.

Theorem 9 (Informal version of Theorem 13)**.** *Assume $k \geq 2$ and $1/\epsilon < \binom{d/2}{k-1}$. If n is sufficiently large relative to d, k, and $1/\epsilon$, then any sketch $\mathcal{S}$ for the Itemset-Frequency-Indicator problem must satisfy $|\mathcal{S}(n, d, k, \epsilon, \delta)| = \Omega(d/\epsilon)$.*

In fact, the same $\Omega(d/\epsilon)$ lower bound applies even to the easier Single-Query Itemset-Frequency-Indicator sketching problem.

Theorem 10 (Informal version of Theorem 14)**.** *Assume $k \geq 2$ and $1/\epsilon < \binom{d/2}{k-1}$. If n is sufficiently large relative to d, k, and $1/\epsilon$, then any sketch $\mathcal{S}$ for the Single-Query Itemset-Frequency-Indicator problem must satisfy $|\mathcal{S}(n, d, k, \epsilon, \delta)| = \Omega(d/\epsilon)$.*

Resolving the complexity of Single-Query Itemset-Frequency-Indicator sketches. Theorem 10 is tight whenever it applies (i.e., when $1/\epsilon < 1/\binom{d/2}{k-1}$), as it matches the $O(d/\epsilon)$ upper bound obtained by the algorithm SUBSAMPLE for the Single-Query Itemset-Frequency-Indicator sketching problem. And when $1/\epsilon \geq 1/\binom{d/2}{k-1}$ and $k = O(1)$, the algorithm RELEASE–ANSWERS achieves an asymptotically optimal summary size of $\binom{d}{k}$ for the Single-Query Itemset-Frequency-Indicator sketching problem. Therefore, our naive upper bounds and Theorem 10 together precisely resolve the complexity of Single-Query Itemset-Frequency-Indicator sketches for all values of d and ϵ, when $k = O(1)$.

An improved lower bound for Itemset-Frequency-Indicator sketches. As we discuss in Section 3.2.1, the $\Omega(d/\epsilon)$ lower bound of Theorem 9 is tight for the Itemset-Frequency-Indicator problem when $1/\epsilon$ is large relative to d — specifically, when $k = O(1)$ and $1/\epsilon = \Theta\left(\binom{d/2}{k-1}\right)$, or when $n = 1/\epsilon$. This fact is arguably surprising, as it shows that in these parameter regimes, the Itemset-Frequency-Indicator sketching problem is *equivalent* in complexity to its single-query analog.

However, when $1/\epsilon \ll \binom{d/2}{k-1}$, Theorem 9 is not tight for Itemset-Frequency-Indicator sketches, because it has suboptimal dependence on d. Our main result establishes a lower bound with optimal dependence on d. For clarity, in this informal overview, we omit the technical relationships that the parameters d, k, and $1/\epsilon$ must satisfy for the following theorems to hold.

Theorem 11 (Informal version of Theorem 17)**.** *For any $k \geq 2$, if n is sufficiently large relative to d, k, and $1/\epsilon$, then any sketch $\mathcal{S}$ for the Itemset-Frequency-Indicator problem must satisfy $|\mathcal{S}(n, d, k, \epsilon, \delta)| = \Omega(d \log(d)/\epsilon^{1-1/k})$.*

Implications of the improved lower bound. Notice that for $k = O(1)$, Theorem 11 matches the $O(\epsilon^{-1} d \log \binom{d}{k})$ upper bound for the Itemset-Frequency-Indicator sketching problem obtained by the algorithm SUBSAMPLE up to a factor $\epsilon^{1/k}$. Moreover, the $O(d/\epsilon)$ upper bound for the Single-Query Itemset-Frequency-Indicator sketching problem achieved by SUBSAMPLE, and the $\Theta(d \log(d)/\epsilon^{1-1/k})$ lower bound of Theorem 11 for the Itemset-Frequency-Indicator sketching problem, together establish the following unsurprising yet non-trivial fact: the Itemset-Frequency-Indicator sketching problem is strictly harder than its single-query analog in a wide range of parameter regimes (specifically, when $1/\binom{d/2}{k-1} \ll \epsilon \ll 1/\log(d)^{1/k}$). The appendix of the full version contains further discussion of the significance of Theorem 11.

An improved lower bound for Itemset-Frequency-Estimator sketches. Finally, we establish a lower bound for the Itemset-Frequency-Estimator sketching problem. This lower bound has the same optimal dependence on the number of attributes, d, as Theorem 11, and a stronger dependence on ϵ.

Theorem 12 (Informal version of Thm. 22)**.** *Let $k = 2$. If n is sufficiently large relative to d, k, and $1/\epsilon$, then any sketch $\mathcal{S}$ for the Itemset-Frequency-Indicator problem must satisfy $|\mathcal{S}(n, d, 2, \epsilon, \delta)| = \Omega(d \log(d)/\epsilon)$.*

3.2 Lower Bounds for Itemset-Frequency-Indicator Sketches

3.2.1 First Lower Bounds

We begin with two relatively simple bounds (Theorems 13 and 14). The former applies to Itemset-Frequency-Indicator sketches, and the latter applies even to their single-query analogs. The proof considers databases in which even a single appearance of an itemset already makes it frequent. We show that, unsurprisingly, essentially no compression is possible in this setting. (We assume that $1/\epsilon$ is an integer throughout).

Theorem 13. *Let $k \geq 2$. Suppose that $1/\epsilon \leq \binom{d/2}{k-1}$, and $\delta < 1$ is constant. Then for $n \geq 1/\epsilon$, the space complexity of any valid Itemset-Frequency-Indicator sketch is $|\mathcal{S}(n, k, d, \epsilon, \delta)| = \Omega(d/\epsilon)$.*

Proof. Our proof uses an encoding argument. Consider the following family of databases. There will be $1/\epsilon$ possible settings for each row; as $n \geq 1/\epsilon$, some rows may be duplicated. For expository purposes, we begin by describing the setting with $n = 1/\epsilon$, in which case there are no duplicated rows. The first $d/2$ columns in each row contain a unique set of exactly $k-1$ attributes. The last $d/2$ attributes in each row are unconstrained. The only minor technicality is that to ensure that each row can receive a unique set of $k-1$ items from the first $d/2$ attributes, we require $1/\epsilon \leq \binom{d/2}{k-1}$.

Given a valid Itemset-Frequency-Indicator or Itemset-Frequency-Estimator sketch for this database, one can recover all of the values $\mathcal{D}(i, j)$ where $j \geq d/2$ as follows. For any $j \geq d/2$, let $T_{i,j}$ be a set of k attributes, where the first $k-1$ attributes in $T_{i,j}$ correspond to the $k-1$ attributes in the first $d/2$ columns in the ith row, and the final attribute in $T_{i,j}$ is j. Notice that $T_{i,j} \in \mathcal{D}$ if and only if $\mathcal{D}(i, j) = 1$. Moreover, since $n = 1/\epsilon$ we have that $f_T \geq \epsilon$ if and only if $\mathcal{D}(i, j) = 1$. Given a valid Itemset-Frequency-Indicator or Itemset-Frequency-Estimator sketch for this database, one can iterate over all itemsets $T_{i,j}$ to recover all the values $\mathcal{D}(i, j)$ where $j \geq d/2$. Since these are an unconstrained set of $d/(2\epsilon)$ bits, the space complexity of storing them (with $1 - \Omega(1)$ failure probability) is $\Omega(d/\epsilon)$ by standard information theory.

For n a multiple of $1/\epsilon$, we construct a database with $1/\epsilon$ rows as above, and duplicate each row $n\epsilon$ times; in this case we have $f_T \geq \epsilon$ if and only if $\mathcal{D}(i, j) = n\epsilon$. More generally, when $n \geq 1/\epsilon$, duplicating each row at least $\lfloor n\epsilon \rfloor$ times, we have $f_T \geq \epsilon$ if and only if $\mathcal{D}(i, j) \geq \lfloor n\epsilon \rfloor$. □

We remark that an entirely similar proof holds whenever $1/\epsilon \leq \binom{\alpha d}{k-1}$ for any constant $\alpha < 1$; we chose $\alpha = 1/2$ for convenience. One simply uses the last $(1-\alpha)d$ bits in each row as the unconstrained bits of $\mathcal{D}$ within the proof of Theorem 13.

As mentioned in Section 3.1, Theorem 13 is tight for Itemset-Frequency-Indicator sketches when ϵ is small relative to the other input parameters n or d. In particular, when $n = 1/\epsilon$, RELEASE–DB provides a trivial matching sketch that is $O(n \cdot d) = O(d/\epsilon)$ bits in size. In addition, when $k = O(1)$ and $1/\epsilon \geq \binom{d/2}{k-1}$, RELEASE–ANSWERS provides a matching sketch that is $O(\binom{d}{k}) = O(d/\epsilon)$ bits in size.

In fact, the argument of Theorem 13 extends even to the single-query case.

Theorem 14. *Let $k \geq 2$. Suppose that $1/\epsilon \leq \binom{d/2}{k-1}$, and the failure probability is $\delta < 1/3$. Then for $n \geq 1/\epsilon$, the space complexity of any valid Itemset-Frequency-Indicator sketch is $|\mathcal{S}(n, k, d, \epsilon, \delta)| = \Omega(d/\epsilon)$.*

Proof. Recall that in the setting of one-way randomized communication complexity, there are two parties, Alice and Bob. Alice has an input $x \in \mathcal{X}$, Bob has an input $y \in \mathcal{Y}$, and Alice and Bob both have access to a public random string r. Their goal is to compute $f(x, y)$ for some agreed upon function $f\colon \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$. Alice sends a single message $m(x, r)$ to Bob. Based on this message, Bob outputs a bit, which is required to equal $f(x, y)$ with probability at least $2/3$.

We consider the well-known INDEX function. In this setting, Alice's input x is an N-dimensional binary vector, Bob's input y is an index in $[N]$, and $f(x, y) = x_y$, the y'th bit of x. It is well-known that the one-way randomized communication protocols for INDEX require cost $\Omega(N)$ [Abl96]. We show how to use any Single-Query Itemset-Frequency-Indicator sketching algorithm $\mathcal{S}$ to obtain a one-way communication protocol for INDEX on vectors of length $N = (d/2) \cdot 1/\epsilon$, with cost proportional to $|\mathcal{S}(n, d, \epsilon, k, \delta)|$.

Specifically, let $(n, d, k, \epsilon, \delta)$ be as in the statement of the theorem. Consider any Boolean vector $x \in \{0, 1\}^N$, where $N = (d/2) \cdot 1/\epsilon$. We associate each index $y \in [N]$ with a unique k-itemset $T_y \subseteq [d]$ of the following form: the first $k-1$ attributes in T_y are each in $[d/2]$, and the final attribute in T_y is in $\{d/2+1, \ldots, d\}$. The proof of Theorem 13 established the following fact: given any vector $x \in \{0, 1\}^N$, there exists a database $\mathcal{D}_x$ with d columns and n rows satisfying the following two properties for all $y \in [N]$:

$$x_y = 1 \Longrightarrow f_{T_y}(\mathcal{D}_x) \geq \epsilon, \text{ and } x_y = 0 \Longrightarrow f_{T_y}(\mathcal{D}_x) = 0 < \epsilon/2. \quad (4)$$

Hence, we obtain a one-way randomized protocol for the INDEX function as follows: Alice sends to Bob $\mathcal{S}(\mathcal{D}_x, k, \epsilon, \delta)$ at a total cost of $|\mathcal{S}(n, d, \epsilon, k, \delta)|$ bits, and Bob outputs $\mathcal{Q}(\mathcal{S}(\mathcal{D}_x, k, \epsilon, \delta), T_y)$. It follows immediately from Equation (4) and Definition 3 that Bob's output equals x_y with probability $1 - \delta$. We conclude that $|\mathcal{S}(n, d, \epsilon, k, \delta)| = \Omega(N) = \Omega(d/\epsilon)$, completing the proof. □

3.2.2 The Core Argument: Encoding Patterns

We now turn to stating and proving Theorem 17, which establishes that SUBSAMPLE outputs a summary of nearly optimal size for the Itemset-Frequency-Indicator sketching problem in a wide range of parameter regimes. As before, we use an encoding argument. The idea is to show the existence of a large collection of *patterns* that can be encoded into a database $\mathcal{D}$, and that can be accurately recovered from any sketch of $\mathcal{D}$. Here a pattern will be a collection of k-itemsets. The minimal sketch size in bits must then be at least the logarithm of the number encodable patters. We begin with defining the notion of an encodable pattern.

Definition 15 (Encodable Pattern)**.** *A pattern $R = \{T_1, \ldots, T_t\}$ such that $|T_i| = k$ and $T_i \subseteq [d]$ is* encodable *if there exists a database with d attributes $\mathcal{D} = \textsc{GenDB}(R)$ such that*

$$\forall\, T \subseteq [d], |T| = k, \qquad T \in R \Longrightarrow f_T(\mathcal{D}) \geq \epsilon \quad \textit{and} \quad T \notin R \Longrightarrow f_T(\mathcal{D}) \leq \epsilon/2\ .$$

For the encoding argument to go through, we must show that there are many encodable patterns.

Lemma 16. *If $t = d/(6k\epsilon^{1-1/k})$ and $k/(6\sqrt{d}) \leq \epsilon^{1-1/k} \leq 1/(18\log(10d))$, then the collection $\mathbf{R}$ of encodable patterns has size $|\mathbf{R}| = \Omega(\binom{\binom{d}{k}}{t})$. Moreover, this statement remains valid if we require that the database $\mathcal{D} = \text{GenDB}(R)$ in Definition 15 contains at most $30\epsilon^{-2}\log\binom{d}{k}$ rows.*

Lemma 16 is the most technically challenging theorem of this section. Its proof is given in Sections 3.2.3 and 3.2.4. Before we prove Lemma 16, we present our main result assuming its correctness.

Theorem 17. *Let $k \geq 2$ and $\delta < 1$ be constants, and suppose that $k/(6\sqrt{d}) \leq \epsilon^{1-1/k} \leq 1/(18\log(10d))$. Then for sufficiently large n, any valid Itemset-Frequency-Indicator sketch $\mathcal{S}$ must satisfy $|\mathcal{S}(n, d, k, \epsilon, \delta)| = \Omega\left((d\log d)/\epsilon^{1-1/k}\right)$. Specifically, the lower bound holds as long as $n \geq 30\epsilon^{-2}\log\binom{d}{k}$.*

Proof. Let R be an encodable pattern and let $\mathcal{D} = \text{GenDB}(R)$ be the encoding of R into a database. Let $(\mathcal{S}, \mathcal{Q})$ be a valid sketch of $\mathcal{D}$. Given $(\mathcal{S}, \mathcal{Q})$ one could recover the pattern R exactly with probability $1-\delta$, simply by exhaustively checking, for all k-itemsets T, whether $\mathcal{Q}(\mathcal{S}(\mathcal{D}), T) = 1$. Assume there exists a sketching algorithm $\mathcal{S}$ exhibiting space complexity $|\mathcal{S}|$. Since the sketch identifies a specific encodable pattern R with positive probability $1 - \delta = \Omega(1)$, we must have that $|\mathcal{S}| \geq \Omega(\log|\mathbf{R}|)$ where $\mathbf{R}$ is the set of all encodable patterns. Lemma 16 then provides $|\mathcal{S}| = \Omega(\log|\mathbf{R}|) = \Omega\left(\log\binom{\binom{d}{k}}{t}\right) = \Omega\left(t\log\binom{d}{k}\right) = \Omega((d\log d)/\epsilon^{1-1/k})$. □

We remark that for any constant $c < 1$, our analysis can be modified to replace the condition that $k/(6\sqrt{d}) \leq \epsilon^{1-1/k}$ in Theorem 17 with the weaker condition $k/(6d^c) \leq \epsilon^{1-1/k}$. We chose $c = 1/2$ in the statement of the theorem for simplicity and concreteness.

3.2.3 Encoding Patterns Into Databases

In order to investigate the set of encodable patterns (Definition 15) we first explore an encoding procedure GenDB in Algorithm 1.

Algorithm 1 GenDB(R, ϵ, d)

```
$\mathcal{D} \leftarrow$ database with $d$ columns and no rows
for $i \in [12k\log(d)/\epsilon]$ do
    $\mathcal{D}' \leftarrow$ GenSmallDB$(R, \epsilon, d)$
    Append $\mathcal{D}'$ to $\mathcal{D}$
Return $\mathcal{D}$
function GenSmallDB($R$, $\epsilon$, $d$)
    $\mathcal{D}' \leftarrow$ all zeros database with $d$ columns and $1/\epsilon$ rows
    for $T_i \in R$ do
        Choose $r$ uniformly at random from $[1/\epsilon]$
        for $j \in T_i$ do
            $\mathcal{D}'(r, j) \leftarrow 1$ //set the $j$th entry of the $r$th row of $D'$ to 1.
    Return $\mathcal{D}'$.
```

Lemma 18. *Given as input a pattern $R = \{T_1, \ldots, T_t\}$, suppose there exists a randomized algorithm that constructs a database $\mathcal{D}'$ such that if $T \in R$ then $f_T(\mathcal{D}') \geq \epsilon$ and $\mathbb{E}[f_T] \leq \epsilon/4$ otherwise (here, the expectation is taken only over the random bits used by the algorithm in the construction of $\mathcal{D}'$). Then R is encodable by a database $\mathcal{D} \in \left(\{0,1\}^d\right)^n$, where $n = 12k\log(d)/\epsilon^2$.*

Proof. Let $\mathcal{D} = [\mathcal{D}'_1; \mathcal{D}'_2; \ldots; \mathcal{D}'_m]$ be a database containing m i.i.d. constructions of $\mathcal{D}'$; that is, the randomized algorithm for constructing the database given the pattern R is run m times independently. We have that $f_T(\mathcal{D}) = \frac{1}{m}\sum_{i=1}^{m} f_T(\mathcal{D}'_i)$. For $T \notin R$ we have that $f_T(\mathcal{D}'_i)$ are i.i.d. random variables in the range $[0, 1]$ with expectation at most $\epsilon/4$. A standard multiplicative Chernoff bound [MU05, Theorem 4.4] implies that $\Pr[f_T(\mathcal{D}) \geq \epsilon/2] \leq e^{-m\epsilon/12}$. Setting $m > 12k\ln(d)/\epsilon$ we get that $\Pr[f_T(\mathcal{D}) \geq \epsilon/2] < 1/\binom{d}{k}$. By invoking the union bound we get that $\Pr[\forall T \notin R,\ f_T(\mathcal{D}) \leq \epsilon/2] > 0$. This shows that there exists a database for which R is encodable. □

We now turn our attention to the function GENSMALLDB in Algorithm 1. GENSMALLDB creates a small database $\mathcal{D}'$ with exactly $1/\epsilon$ rows. For every itemset $T \in R$, GENSMALLDB chooses a database row r uniformly at random out of $[1/\epsilon]$ and sets all attributes in T in that row to 1. Note that some rows may include multiple itemsets from R, and in fact some rows might not contain any. We prove that GENSMALLDB produces databases with the properties given in the assumptions of Lemma 18, under certain assumptions about the structure of pattern R.

Definition 19 (Balanced Pattern). *A pattern $R = \{T_1, \ldots, T_t\}$ such that $T_i \subseteq [d]$ and $|T_i| = k$ is* balanced *if the following two conditions are satisfied.*

1. *$\forall\ j \in [d]$, there are at most $2kt/d$ values of i for which $j \in T_i$.*
2. *$\forall\ \{j_1, j_2\} \subseteq [d]$ with $j_1 \neq j_2$, there are at most 3 itemsets $T_i \in R$ for which $\{j_1, j_2\} \subset T_i$.*

We now prove that DBGEN successfully encodes any balanced pattern with high probability.

Lemma 20. *Let $k \geq 2$ be a constant. A pattern $R = \{T_1, \ldots, T_t\}$ such that $T_i \in [d]$ and $|T_i| = k$ that is balanced is encodable for some $t \leq d/(6k\epsilon^{1-1/k})$, for ϵ smaller than a small constant depending on k.*

Proof. We consider constant $k \geq 3$ (the proof for the case $k = 2$ is simpler, and is provided in the full version). We show that the randomized algorithm GENSMALLDB satisfies the conditions of Lemma 18 when run on input (R, ϵ, d). Let $\mathcal{D}'$ denote the (random) database generated by GENSMALLDB. First, observe that for any $T_i \in R$ $f_{T_i}(\mathcal{D}') \geq \epsilon$ with probability 1, since GENSMALLDB ensures that each T_i is contained in at least one of the $1/\epsilon$ rows of $\mathcal{D}'$.

We now show that for any k-itemset $T \notin R$, $\mathbb{E}[f_T(\mathcal{D}')] \leq \epsilon/4$. By symmetry and the linearity of expectation, we observe that $\mathbb{E}[f_T(\mathcal{D}')] = \Pr[T \in \mathcal{D}'(1)]$. We define a *minimal cover* as a subset $C \subset R$ of the pattern such that $T \subset \cup_{T_i \in C} T_i$, and for no subset C' of C do we have $T \subset \cup_{T_i \in C'} T_i$. Note a minimal cover has size at most k. If every $T_i \in C$ randomly maps to the first row in $\mathcal{D}'$ then T appears in that row as well. We show below that if $|C| = k$, while there are many such minimal covers, each of them only maps to the first row with the small probability $\epsilon^{|C|}$. If $|C| < k$, the probability that such a cover maps to the first row is higher, but this is made up for by the fact that the number of such covers is small. This intuition is made explicit below.

For the case that $|C| = k$, each itemset in C must contribute exactly 1 item to T. Hence the probability of such a cover mapping to the first row is ϵ^k. Since each item in T appears in at most $2kt/d$ of the itemsets in R (by Condition 1 of Definition 19) there are at most $(2kt/d)^k$ such covers. Setting $t \leq d/(6k\epsilon^{1-1/k})$ yields for every $k \geq 3$ that $\Pr[T \in \mathcal{D}'(1)$ due to $|C| = k] \leq \epsilon^k (2kt/d)^k \leq \epsilon/8$.

Consider now the case for a minimal cover C with $|C| < k$. In this case, by the pigeon-hole principle there must be at least one itemset $T_i^* \in C$ such that $|T_i^* \cap T| \geq 2$. By Condition 2 of Definition 19, there could only be at most $3\binom{k}{2}$ itemsets $T_i^* \in R$ for which $|T_i^* \cap T| \geq 2$. The probability that at least one of those itemsets maps to the first row of $\mathcal{D}'$ is at most $\epsilon \cdot 3\binom{k}{2}$. Moreover, as $T \notin R$, there must be at least one more item in $j \in T$ that is mapped to the first row from another of the T_is. By Condition 1 of Definition 19, this item j belongs to at most $2kt/d$ itemsets in R. The probability that j appears in $\mathcal{D}'(1)$ is therefore at most $\epsilon \cdot 2kt/d$. Since both events must happen simultaneously, we conclude that $\Pr[T \in \mathcal{D}'(1)$ due to $|C| < k] \leq 6k^3\epsilon^2 t/d \leq \epsilon/8$. for $t = d/(6k\epsilon^{1-1/k})$ and $\epsilon \leq (1/8k^2)^k$, which is a small constant depending only on $k = O(1)$. A union bound gives that for any $T \notin R$ we have $\Pr[T \in \mathcal{D}'(1)] \leq \epsilon/4$. By Lemma 18, we again have that R is encodable. □

3.2.4 There Are Many Encodable Patterns

In Section 3.2.3 we showed that a *balanced* pattern (according to Definition 19) is encodable. Here we argue that a uniformly chosen random pattern is balanced with constant probability. The (simple) proof is deferred to the full version of the paper due to space constraints.

Lemma 21. *A random pattern R satisfies Conditions 1 and 2 of Definition 19 with probability at least $8/10$ if $3d\log(10d)/k \le t \le d^{3/2}/k^2$.*

We now have all the necessary ingredients to complete the proof of Lemma 16. Recall that in the statement of Lemma 16, we choose $t = d/(6k\epsilon^{1-1/k})$. By the requirement that $k/(6\sqrt{d}) \le \epsilon^{1-1/k} \le 1/(18\log(10d))$, we conclude that $3d\log(10d)/k \le t \le d^{3/2}/k^2$. Hence, Lemma 21 implies that a random pattern R of t k-itemsets is balanced with probability at least $8/10$. To conclude, there are $\binom{d}{k}$ possible k-itemsets and thus $\binom{\binom{d}{k}}{t}$ different patterns with exactly t k-itemsets, and we have shown at least an $8/10$ fraction of them are balanced. Lemma 20 implies that balanced patterns are encodable, and Lemma 16 follows.

4 Lower bounds for Itemset-Frequency-Estimator sketches

In this section, we establish a lower bound for the Itemset-Frequency-Estimator sketching problem. Relative to Theorem 17, this lower bound has the same optimal dependence on the number of attributes, d, and a stronger dependence on ϵ.

Theorem 22. *Let $k = 2$ and $\delta < 1$, $\beta > 0$ be constants. Suppose that $(1/d)^{1-\beta} < \epsilon < 1/(90\log(10d))$. Then for sufficiently large n, any Itemset-Frequency-Estimator sketch $\mathcal{S}$ must exhibit space complexity $|\mathcal{S}(n, d, 2, \epsilon, \delta)| = \Omega(d\log(d)/\epsilon)$. Specifically, the lower bound holds as long as $n \ge 500\epsilon^{-3}\log(d)$.*

Due to space constraints, we defer the proof to the full version, and provide only a high-level overview here.

Proof Overview. Our proof follows the same high-level outline of the encoding argument used to establish Theorem 17, but is more technically involved. First, we define a notion of *weakly encodable* patterns. Weak encodability is a strictly milder (if somewhat more technical) property than encodability (Definition 15). Yet we show that any weakly encodable pattern R can be encoded into a database $\mathcal{D}$ such that R can be exactly recovered, given any Itemset-Frequency-Estimator sketch for $\mathcal{D}$, plus a small amount of additional "advice". Second, we show that the set of weakly encodable patterns is large. Whereas in the proof of Theorem 17 we considered patterns $R = \{T_1, \ldots, T_t\}$ of size $t = \Theta(d/(k\epsilon^{1-1/k}))$ and showed that a large constant fraction of such patterns are encodable, here we consider patterns of a larger size (specifically, of size $t = \Theta(d/\epsilon)$), and show that a large constant fraction are *weakly* encodable.

While our definition of weak encodability is somewhat technical, it carries the following motivation. Consider a pattern $R = \{T_1, \ldots, T_t\}$ such that $|T_i| = 2$ and $T_i \subseteq [d]$. Recall (from Algorithm 1) that the function $\textsc{GenSmallDB}(R, \epsilon, d)$ generates a random database $\mathcal{D}'$ with d columns and $1/\epsilon$ rows by picking a random row of $\mathcal{D}'$ for each $T_i \in R$, and placing both elements of T_i into that row.

For each $j \in [d]$, let $h_j = |\{i : j \in T_i\}|$ denote the number of T_is that contain j. For each 2-itemset $T = \{j_1, j_2\} \notin R$, the expected number of rows of $\mathcal{D}'$ that contain both j_1 and j_2 is a certain function of h_{j_1} and h_{j_2}, and we denote this function by $g_{\bar{R}}(h_{j_1}, h_{j_2})$ Similarly, for each $T_i = \{j_1, j_2\} \in R$, the expected number of rows of $\mathcal{D}'$ that contain both j_1 and j_2 is a different function of h_{j_1} and h_{j_2} alone — we denote this function by $g_R(h_{j_1}, h_{j_2})$. A weakly encodable pattern R is a pattern for which there exists a database $\mathcal{D}$ such that (1) For all 2-itemsets $T = \{j_1, j_2\}$, there is a sufficiently large "gap" between $g_{\bar{R}}(h_{j_1}, h_{j_2})$ and $g_R(h_{j_1}, h_{j_2})$, (2) for all $T = \{j_1, j_2\} \notin R$, $f_{T_i}(\mathcal{D})$ is not much larger than $g_{\bar{R}}(h_{j_1}, h_{j_2})$, and (3) for all $T_i = \{j_1, j_2\} \in R$, $f_{T_i}(\mathcal{D})$ is not much smaller than $g_R(h_{j_1}, h_{j_2})$.

Intuitively, R can be recovered given an Itemset-Frequency-Estimator sketch of $\mathcal{D}$ by simply iterating over every possible 2-itemset $T = \{j_1, j_2\}$, and looking at the estimate of $f_T(\mathcal{D})$ returned by the sketch. If this estimate is significantly larger than $g_{\bar{R}}(T)$, then T must be in R. Otherwise, T cannot be in R.

References

[Abl96] Farid M. Ablayev. Lower bounds for one-way probabilistic communication complexity and their application to space complexity. *Theor. Comput. Sci.*, 157(2):139–159, 1996.

[AIS93] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22(2):207–216, June 1993.

[BBR03] Jean-François Boulicaut, Artur Bykowski, and Christophe Rigotti. Free-sets: A condensed representation of boolean data for the approximation of frequency queries. *Data Min. Knowl. Discov.*, 7(1):5–22, 2003.

[BCD$^+$07a] Boaz Barak, Kamalika Chaudhuri, Cynthia Dwork, Satyen Kale, Frank McSherry, and Kunal Talwar. Privacy, accuracy, and consistency too: A holistic solution to contingency table release. In *Proceedings of the Twenty-sixth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '07, pages 273–282, New York, NY, USA, 2007. ACM.

[BCD$^+$07b] Boaz Barak, Kamalika Chaudhuri, Cynthia Dwork, Satyen Kale, Frank McSherry, and Kunal Talwar. Privacy, accuracy, and consistency too: a holistic solution to contingency table release. In Leonid Libkin, editor, *PODS*, pages 273–282. ACM, 2007.

[BUV14] Mark Bun, Jonathan Ullman, and Salil P. Vadhan. Fingerprinting codes and the price of approximate differential privacy. In *STOC*, 2014.

[CG07] Toon Calders and Bart Goethals. Non-derivable itemset mining. *Data Min. Knowl. Discov.*, 14(1):171–206, February 2007.

[CH10] Graham Cormode and Marios Hadjieleftheriou. Methods for finding frequent items in data streams. *VLDB J.*, 19(1):3–20, 2010.

[CKN08] James Cheng, Yiping Ke, and Wilfred Ng. A survey on algorithms for mining frequent itemsets over data streams. *Knowl. Inf. Syst.*, 16(1):1–27, 2008.

[CTUW14] Karthekeyan Chandrasekaran, Justin Thaler, Jonathan Ullman, and Andrew Wan. Faster private release of marginals on small databases. In Moni Naor, editor, *ITCS*, pages 387–402. ACM, 2014.

[DR98] Devdatt Dubhashi and Desh Ranjan. Balls and bins: A study in negative dependence. *Random Structures and Algorithms*, 13(2):99–124, 1998.

[FK04] Uriel Feige and Shimon Kogan. Hardness of approximation of the balanced complete bipartite subgraph problem. Technical report, 2004.

[GHRU13] Anupam Gupta, Moritz Hardt, Aaron Roth, and Jonathan Ullman. Privately releasing conjunctions and the statistical query barrier. *SIAM J. Comput.*, 42(4):1494–1520, 2013.

[Goe03] B. Goethals. Survey on frequent pattern mining. Manuscript, 2003.

[HCW06] Matthew Hamilton, Rhonda Chaytor, and Todd Wareham. The parameterized complexity of enumerating frequent itemsets. In *Proceedings of the Second International Conference on Parameterized and Exact Computation*, IWPEC'06, pages 227–238, Berlin, Heidelberg, 2006. Springer-Verlag.

[HRS12] Moritz Hardt, Guy N. Rothblum, and Rocco A. Servedio. Private data release via learning thresholds. In Yuval Rabani, editor, *SODA*, pages 168–187. SIAM, 2012.

[KRSU10a] Shiva Prasad Kasiviswanathan, Mark Rudelson, Adam Smith, and Jonathan Ullman. The price of privately releasing contingency tables and the spectra of random matrices with correlated rows. In Schulman [Sch10], pages 775–784.

[KRSU10b] Shiva Prasad Kasiviswanathan, Mark Rudelson, Adam Smith, and Jonathan Ullman. The price of privately releasing contingency tables and the spectra of random matrices with correlated rows. In Schulman [Sch10], pages 775–784.

[LLSW05] Jinyan Li, Haiquan Li, Donny Soh, and Limsoon Wong. A correspondence between maximal complete bipartite subgraphs and closed patterns. In *Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases*, PKDD'05, pages 146–156, Berlin, Heidelberg, 2005. Springer-Verlag.

[LN90] Nathan Linial and Noam Nisan. Approximate inclusion-exclusion. In Harriet Ortiz, editor, *STOC*, pages 260–270. ACM, 1990.

[MT96] Heikki Mannila and Hannu Toivonen. Multiple uses of frequent sets and condensed representations. In *KDD*, 1996.

[MU05] Michael Mitzenmacher and Eli Upfal. *Probability and computing - randomized algorithms and probabilistic analysis*. Cambridge University Press, 2005.

[PDZH04] Jian Pei, Guozhu Dong, Wei Zou, and Jiawei Han. Mining condensed frequent-pattern bases. *Knowl. Inf. Syst.*, 6(5):570–594, 2004.

[Rec11] Benjamin Recht. A simpler approach to matrix completion. *J. Mach. Learn. Res.*, 12:3413–3430, December 2011.

[Sch10] Leonard J. Schulman, editor. *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, 5-8 June 2010*. ACM, 2010.

[Tro12] Joel A Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, 2012.

[TUV12] Justin Thaler, Jonathan Ullman, and Salil P. Vadhan. Faster algorithms for privately releasing marginals. In Artur Czumaj, Kurt Mehlhorn, Andrew M. Pitts, and Roger Wattenhofer, editors, *ICALP (1)*, volume 7391 of *Lecture Notes in Computer Science*, pages 810–821. Springer, 2012.

[Yan04] Guizhen Yang. The complexity of mining maximal frequent itemsets and maximal frequent patterns. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 344–353, New York, NY, USA, 2004. ACM.

[YCPS13] Grigory Yaroslavtsev, Graham Cormode, Cecilia M. Procopiuc, and Divesh Srivastava. Accurate and efficient private release of datacubes and contingency tables. In Christian S. Jensen, Christopher M. Jermaine, and Xiaofang Zhou, editors, *ICDE*, pages 745–756. IEEE Computer Society, 2013.