Efficient Algorithms for Privately Releasing Marginals via Convex Relaxations

Cynthia Dwork 1 · Aleksandar Nikolov 2 · Kunal Talwar 3

Received: 1 August 2014 / Revised: 22 January 2015 / Accepted: 22 January 2015 / Published online: 24 April 2015
© Springer Science+Business Media New York 2015

Abstract Differential privacy is a definition giving a strong privacy guarantee even in the presence of auxiliary information. In this work, we pursue the application of geometric techniques for achieving differential privacy, a highly promising line of work initiated by Hardt and Talwar (Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC’10, pp 705–714. ACM Press, New York, 2010). We apply these techniques to the problem of marginal release. Here, a database refers to a collection of the data of n individuals, each characterized by d binary attributes. A k-way marginal query is specified by a subset S of k attributes, together with a |S|-dimensional binary vector β specifying their values. The true answer to this query is a count of the number of people in the database whose attribute vector restricted to S agrees with β. Information theoretically, the error complexity of marginal queries—how “wrong” do the answers have to be in order to preserve differential privacy—is well understood: the per-query additive error is known to be at least Ω(\min{\sqrt{n}, d^{k/2}}) and at most \tilde{O}(\sqrt{nd^{k/2}/4}). However, no polynomial time algorithm with error complexity

Editors-in-Charge: Siu-Wing Cheng and Olivier Devillers

Preliminary version in Proceedings of the 30th Annual Symposium on Computational Geometry, 2014.

Cynthia Dwork
dwork@microsoft.com

Aleksandar Nikolovalenik@microsoft.com; anikolov@cs.rutgers.edu

Kunal Talwar
kunal@kunaltalwar.org

1 Microsoft Research, Mountain View, CA, USA
2 Microsoft Research, Redmond, WA, USA
3 Google, Mountain View, CA, USA
as low as the information-theoretic upper bound is known for small $n$. We present a polynomial time algorithm that matches the best known information-theoretic bounds when $k = 2$; more generally, by reducing to the case $k = 2$, for any distribution on marginal queries, our algorithm achieves average error at most $\tilde{O}(\sqrt{nd^{k/2}/4})$, an improvement over previous work when $k$ is small and when error $o(n)$ is desirable. Using private boosting, we are also able to give nearly matching worst-case error bounds. Our algorithms are based on the geometric techniques of Nikolov et al. (Proceedings of the 45th Annual ACM Symposium on Theory of Computing, STOC’13, pp 351–360. ACM Press, New York, 2013), wherein a vector of “sufficiently noisy” answers is projected onto a particular convex body. We reduce the projection step, which is expensive, to a simple geometric question: given (a succinct representation of) a convex body $K$, find a containing convex body $L$ that one can efficiently optimize over, while keeping the Gaussian width of $L$ small. This reduction is achieved by a careful use of the Frank–Wolfe algorithm.

Keywords Differential privacy · Convex geometry · Combinatorial optimization

1 Introduction

A basic task in data analysis is the release of a specified set of statistics of the data. In this work, we address the question of privacy-preserving release of the set of low-dimensional marginals of a dataset. These are a ubiquitous and important subclass of queries, constituting contingency tables in statistics and OLAP cubes in databases. Official agencies such as the Census Bureau, the Internal Revenue Service, and the Bureau of Labor Statistics all release certain sets of low-dimensional marginals for the data they collect.

In this work, the database will be a collection of the data of $n$ individuals, each characterized by $d$ binary attributes. A $k$-way marginal query is specified by a subset $S$ of $k$ attributes, and a $\left| S \right|$-dimensional binary vector $\beta$ specifying their values. The result for this query is a count of the number of people in the database whose attribute vector restricted to $S$ agrees with $\beta$. In this work, we will be interested in releasing all $k$-way marginals of a database in $\left(\{-1, 1\}^d\right)^n$ for some small integer $k$.

In many of the settings mentioned above, the data in question contain individuals’ private information, and there are ethical, legal, or business reasons to prevent the disclosure of individual information. Differential privacy [14] is a recent definition that gives a strong privacy guarantee even in the presence of auxiliary information. It is the only known privacy definition to do so, and has quickly become the standard definition of privacy in such settings. It has been the subject of extensive research in the last decade (see the upcoming book [13]), and will be the definition of privacy in this work. Specifically, we will be working with a variant known as $(\varepsilon, \delta)$-differential privacy or approximate differential privacy. We are thus interested in differentially private mechanisms for releasing (estimates of) low-dimensional marginals. Our mechanisms will release noisy answers to these queries, and we would like to design computationally efficient mechanisms that add as little noise as possible. In particular, we are interested
in achieving error per query $\alpha n$ for $\alpha < 1$ and possibly subconstant when the database size $n$ is not too large.

This problem of differentially private release of marginals has attracted a lot of interest. The Gaussian noise mechanism [10,12,14] works in a very general setting and adds noise only $\tilde{O}(d^{k/2})$ to each of the $O(d^k)$ marginals,$^1$ independent of $n$. This implies that the error per query is $\alpha n$ as long as $n = \tilde{\Omega}(d^{k/2} \alpha^{-1})$. Barak et al. [2] showed that these noisy answers can be made consistent with a real database without sacrificing accuracy. In general, this bound is tight: Kasiviswanathan et al. [27] show that no differentially private mechanism (even for approximate DP) can add error $o(\min(\sqrt{n}, d^{k/2}))$ for constant $k$.

Starting with the work of Blum et al. [3], a long line of work [16,17,22,23,26,31] has shown that private mechanisms with error significantly smaller than that of the Gaussian mechanism exist for small $n$. Specifically, an error bound of about $O(\sqrt{n}d^{1/4})$ per query is achievable [22,23,26]; this error bound nearly matches the lower bound and implies error per query at most $\alpha n$ for $n$ as small as $\tilde{\Omega}(d^{1/2} \alpha^{-2})$. The latter lower bound on $n$ is tight (within factors logarithmic in $d^k$) when $\alpha$ is a small enough constant [4]. However, the known algorithms giving these results have running time that is at least exponential in $d$, which may be restrictive in settings where $d$ is large. Ullman and Vadhan [33] show that, assuming the existence of one-way functions, any private mechanism that generates synthetic data must have running time $d^{\omega(1)}$ or have error $\Omega(n)$ for some 2-way marginal query. All algorithms cited above, except for private boosting [17], do produce synthetic data.

Recent work has shown that significantly faster mechanisms are possible for marginal queries, using sophisticated learning theory techniques to design approximate but compact representations of databases. In these works, however (see below for a more detailed comparison), either the running time is still $2^{d^{\Omega(1)}}$ for $k = O(1)$ or the error is still much larger than what is achievable inefficiently.

In this work, we show that for any distribution over $k$-way marginals, in time polynomial in $n$ and $d^k$ one can achieve additive error which is within an $\tilde{O}(d^{k/2}/4)$ factor from the lower bound in general.

**Theorem 1** For any distribution $p$ over $k$-way marginal queries, there is an $(\varepsilon, \delta)$-differentially private mechanism $\mathcal{M}$ such that for any database containing $n$ individuals

$$E_{\mathcal{M}}E_{q \sim p}[\text{err}(q)] = O\left(\frac{\sqrt{n}d^{[k/2]/4}(\log 1/\delta)^{1/4}}{\varepsilon^{1/2}}\right),$$

where $\text{err}(q)$ is the additive error incurred by the mechanism on query $q$. Moreover, the mechanism $\mathcal{M}$ runs in time polynomial in $d^k$ and $n$.

We note that for $k = 2$, this matches the error of the best known (inefficient) mechanism; moreover, it implies that the error of the mechanism is non-trivial as long

$^1$ Throughout this introduction, we will ignore the dependence on privacy parameters such as $\varepsilon$ and $\delta$, and use the $\tilde{O}$ and $\tilde{\Omega}$ notations to hide factors logarithmic in $d^k$. 

© Springer
as \( n = \tilde{\Omega}(\sqrt{d}) \), which is the best possible \([4]\). Prior to our work, efficient mechanisms that match the information-theoretic upper bound on error were known only for \( k = 1 \).

Further, we show that this average error bound can be converted to a worst-case bound using the boosting framework of Dwork et al. \([17]\).

**Theorem 2** Let \( 2^{-n} \leq \delta \leq n^{-2} \). For any \( k \) and \( n \leq d^k \leq 2^k n \), there is an \((\varepsilon, \delta)\)-differentially private mechanism \( M \) such that for any database containing \( n \) individuals, with probability at least \( 2/3 \), we have the following error bound for all \( k \)-way marginal queries \( q \):

\[
\text{err}(q) \leq O\left(\sqrt{n} \cdot d^{\lceil k/2 \rceil/4} \cdot (k \log d + \log 1/\delta)^{1/2} (k \log d)^{1/2}/\varepsilon\right).
\]

Moreover, the mechanism \( M \) runs in time polynomial in \( d^k \) and \( n \).

For \( k = 2 \) our worst-case error bound is in fact an improvement on the error bound achieved with the (inefficient) synopsis generator in \([17]\); the reason behind our improvement is that we are able to compute more concise synopses. Our algorithms avoid the negative results of Ullman and Vadhan \([33]\) because they do not produce synthetic data, i.e. a database in \((\{-1, 1\}^d)^n\); see below for a more detailed discussion.

### 1.1 Techniques

We pursue the approach of applying geometric techniques to differential privacy, which goes back to the work of Hardt and Talwar \([24]\). More specifically, our starting point is a simple mechanism of Nikolov et al. \([29]\) with near optimal (up to a factor of \( O(d^{1/4}) \)) average additive error for \( k \)-way marginals. This mechanism requires least squares projection onto \( nK \), where \( n \) is the number of people in the database and \( K \) is the symmetric convex hull of the columns of the query matrix. (Equivalently, \( nK \) is the symmetric convex hull of all possible vectors of query answers on databases of at most \( n \) people.) The Frank–Wolfe algorithm \([18]\) shows that, given a subroutine that can efficiently optimize linear functions over \( K \), one can compute an approximately optimal projection. This reduces the projection step to optimizing a linear function over \( K \). However, optimizing a linear function over \( K \) derived from the \( k \)-way marginal queries generalizes \( \text{MAX}k\cdot\text{XOR} \), and is thus \( \text{NP} \)-hard. Thus one cannot hope to optimize over \( K \) in time \( \text{poly}(d) \).

To get around this obstacle, we first give a general reduction. We observe that to accurately answer a set of queries, it suffices to project to a polytope \( L \) containing \( K \) such that \( L \) approximates \( K \) well enough. A natural approach is to find an \( L \) satisfying \( K \subseteq L \subseteq C \cdot K \) for some small \( C \) (we abuse terminology and call the minimum such \( C \) the Banach–Mazur distance between \( K \) and \( L \)). Instead we show that a much weaker notion of approximation suffices. The mechanism that projects to \( L \supseteq K \) will have error that is close to optimal as long as the expected length of the projection of \( L \) onto a random Gaussian (known as the Gaussian width of \( L \)) is not much larger than the corresponding quantity for \( K \). Thus it suffices that \( L \) approximates \( K \) only in an average sense, instead of the approximation in every direction that the Banach–Mazur
distance condition imposes. If in addition we can efficiently optimize linear functions over $L$, then, using the Frank–Wolfe algorithm, we can also efficiently project on $L$. This reduction is of independent interest and may allow one to design accurate and efficient mechanisms for other query sets.

After giving general reductions, we return to the problem of releasing marginals. For $k = 2$, the existence of a relaxation $L$ with the required properties follows from Grothendieck’s inequality [20] in functional analysis. Informally, it shows that the maximum value of the quadratic form $\sum_{ij} g_{ij} x_i x_j$ over $x \in \{-1, 1\}^d$ is within a constant factor of the maximum value of $\sum_{ij} g_{ij} \langle u_i, u_j \rangle$ over unit vectors $u_i$. The former is closely related to linear functions over the polytope $K$ derived from 2-way marginals; the latter is a semidefinite program and its feasible region (appropriately projected) gives us the convex body $L$. For $k \geq 3$, we show that the problem can be reduced to the $k = 2$ case problem where $d$ is replaced by $2d^{\lceil k/2 \rceil}$.

The above approach gives us average error bounds, where the average is taken with respect to any distribution on queries. To get a worst-case error bound, we use the Boosting for Queries framework of [17]. This requires that answers returned by the average-error algorithm have a concise representation. We can show that in the case of marginals, these answers can be represented by a relatively small number of unit vectors $u_i$ as above. However, a priori they may be in a high-dimensional space. Using the Johnson–Lindenstrauss Lemma, these vectors can be projected down to a small number of dimensions without adding too much additional error, allowing us to get a concise representation as needed. We also show that in the general case, when the convex body $L$ has extreme points that have a concise representation (e.g. by virtue of having not too many vertices), the properties of the Frank–Wolfe algorithm imply a concise representation for an approximate projection, which leads to a mechanism with small worst-case error. There is a trade-off here: as we (lossily) compress our representation (e.g. by projection into a small number of dimensions), we introduce more error on average, but the smaller synopsis reduces our generalization error and hence leads to smaller additional error from the boosting step. We can thus optimize the error during boosting by picking the right point on this trade-off. Fortunately, for the marginal release problem, this error is of the same order as the average per query error in the relaxed projection mechanism, leading to the claimed bounds.

**Discussion** We note that our work reduces the problem of privately releasing $k$-way marginals with small error to finding a suitably tight relaxation $L$ of the $k$-XOR polytope $K_k$, such that we can efficiently optimize over $L$. If we want $L$ to be within a small Banach–Mazur distance from $K_k$, then this problem seems hard, and it is known that starting from the natural linear programming relaxation $K_k^{LP}$, nearly linear number of levels of the Lasserre hierarchy are needed to get an $L$ which is close in Banach–Mazur distance to $K$. These results [5, 19] in fact show that under a particular distribution over directions, the gap between $K$ and a $o(n)$-level lift of the LP relaxation $K_k^{LP}$ remains large on average. However, these distributions, while natural for the $k$-XOR problem, are far from standard Gaussian and there is little reason to believe that a better approximation is not possible under the Gaussian distribution. In particular, the distributions in [5, 19] produce sparse vectors with high probability, while a sample from a Gaussian typically has coordinates bounded away from 0. One line of attack is
to find a low degree sum-of-squares proof (see e.g. [30]) of the small Gaussian width of $K$: this would imply that a small number of levels of the Lasserre hierarchy yield a good $L$. We leave open this question of designing a better $L$ for the $k$-XOR polytope.

1.2 Related Work

The most closely related works to ours are those of Thaler et al. [32] and Chandrasekaran et al. [6], and we now discuss these in more detail. Improving on a long line of work [7,22,25], the authors in [32] show that in time $d^{O(\sqrt{k \log(1/\alpha)})}$ one can construct a private synopsis of a dataset such that any $k$-way marginal query can be answered from it, with error $\alpha \cdot n$, as long as $n$ is at least $d^{O(\sqrt{k \log(1/\alpha)})}$. For a constant $\alpha$, the algorithm in [32] has the advantage of being online and, when only a few of the $d^k$ queries are asked, has running time much smaller than $d^k$. However, they add error that is much more than necessary (e.g. the best inefficient mechanisms get error $\alpha n$ as long as $n$ is $\Omega(\sqrt{kd} \alpha^{-2})$).

The recent work of Chandrasekaran et al. [6] presents a different point in the trade-off: they show that one can get error $0.01n$ for $n$ at least $d^{0.51}$, with running time $\min\{\exp(d^{(1-\Omega(1/\sqrt{k}))}), \exp(d/\log^{0.99}d)\}$ for any sequence of $k$-way marginals. Thus they improve on the $2^d$ running time even when $k$ is large. However, the running time of this mechanism is still $\exp(d^c)$ for some constant $c$ depending on $k$.

If error $\alpha n$ is desired for small (possibly subconstant) $\alpha$, the lower bound on $n$ and the running time of the above algorithms deteriorate quickly. It is instructive to consider the regime in which the error should be at most $n^{1-\gamma}$ for constant $\gamma$. In order to achieve such small error for $k = O(1)$, the algorithms in [25,32] require databases of size as large as required by the Gaussian noise mechanism (see Lemma 2 below), i.e. $n = \tilde{\Omega}(d^{k/2(1-\gamma)})$. By contrast our work gives error $n^{1-\gamma}$ as long as $n$ is $\tilde{\Omega}(d^{k/2(1-2\gamma)})$, a nearly quadratic improvement. Even for $\alpha$ a small constant, the database size lower bound $n = \tilde{\Omega}(d^{k/2})$ required by our algorithm improves significantly on previous work for small $k$. Alternately, translating to additive error, the additive error in [32] is always at least $\Omega(\min(n2^{-k}, d^{k/2}))$, compared to the $\tilde{O}(\sqrt{n}d^{[k/2]/4})$ bound that we get.

As mentioned above, Kasiviswanathan et al. [27] showed that any differentially private mechanism must incur average case additive error $\Omega(\min(\sqrt{n}, \sqrt{d}^k))$. Moreover, recently Bun et al. showed that to achieve error at most $\alpha_0 n$ for a small enough constant $\alpha_0$, we need $n = \tilde{\Omega}(\sqrt{d})$. These lower bounds come from privacy considerations alone and make no computational assumptions. Ullman and Vadhan [33] show that assuming one-way functions exist, there is an absolute constant $\alpha^* > 0$ such that no polynomial time differentially private algorithm can produce synthetic data that preserve all 2-way marginals up to error $\alpha^* n$, for a database containing $n = \text{poly}(d)$

---

2 In fact the algorithm in [32] is exactly the Gaussian noise mechanism for $\alpha < 2^{-k}$. 

---

© Springer
individuals. Here synthetic data mean that the mechanism computes a new database $D'$ drawn from the original universe $\{-1, 1\}^d$, and to construct the answer to a query $q$ on $D'$, one computes $q(D')$; e.g. the algorithms in [2,3,22,23,26,31] produce such a synopsis (in time exponential in $d$). Our results avoid this lower bound in two ways: the synopses produced by our algorithms are synthetic data from a larger universe (in particular the unit sphere $S^{d-1}$ in $\mathbb{R}^d$), and, moreover, for worst-case error we use boosting for queries, which aggregates different synopses using medians.

2 Preliminaries

2.1 Notation

We denote matrices by upper-case letters, and vectors and scalars by lower-case letters. As standard, we define the $\ell_1$ and $\ell_2$ norms on $\mathbb{R}^m$, respectively, as $\|x\|_1 = \sum_{i=1}^m |x_i|$ and $\|x\|_2 = \sqrt{\sum_{i=1}^m |x_i|^2}$. We use $B_1(n)$ to denote the $\ell_1$ ball of radius $n$, i.e. $B_1(n) = \{x : \|x\|_1 \leq n\}$, and we write $B_1$ for the unit ball $B_1(1)$.

By $x \otimes y$ we denote the tensor product of $x \in \mathbb{R}^{m_1}$ and $y \in \mathbb{R}^{m_2}$, represented as a vector indexed by $[m_1] \times [m_2]$, i.e. $x \otimes y \in \mathbb{R}^{[m_1] \times [m_2]}$, and $(x \otimes y)_{(i,j)} = x_i y_j$.

We use $N(\mu, \sigma^2)$ to denote the Gaussian distribution with mean $\mu$ and variance $\sigma^2$.

We use the notation $\text{poly}(x_1, \ldots, x_k)$ to denote the set of all real polynomials $p(x_1, \ldots, x_k)$ in the variables $x_1, \ldots, x_k$.

2.2 Differential Privacy

A database $D \in \mathcal{U}^n$ of size $n$ is a multiset of $n$ elements from a universe $\mathcal{U}$. Each element of the database represents information about a single individual, and the universe $\mathcal{U}$ is the set of all possible types of individuals. Two databases $D, D'$ are neighboring if their symmetric difference $D \triangle D'$ is at most 1, i.e. if they differ in the presence/absence of a single individual.

We can represent a database $D \in \mathcal{U}^n$ as a histogram as follows: we enumerate the universe $\mathcal{U} = \{e_1, \ldots, e_N\}$ in some arbitrary but fixed way; the histogram $x$ associated with a database $D$ is a vector $x \in \mathbb{R}^N$ such that $x_i$ is equal to the number of occurrences of $e_i$ in $D$. Two useful (and closely related) facts about the histogram representation are that $\|x\|_1 = n$ when $x$ is the histogram of a size $n$ database, and $\|x - x'\|_1 = D \triangle D'$, where $x$ is the histogram of $D$ and $x'$ is the histogram of $D'$. We use the histogram representation extensively in this work.

In this paper, we work under the notion of approximate differential privacy. The definition follows.

**Definition 1** [14,15] An algorithm $\mathcal{M}$ with input domain $\mathbb{R}^N$ and output range $Y$ is $(\epsilon, \delta)$-differentially private if for every two neighboring databases $x, x'$, and every measurable $S \subseteq Y$, $\mathcal{M}$ satisfies

$$\Pr[\mathcal{M}(x) \in S] \leq e^\epsilon \Pr[\mathcal{M}(x') \in S] + \delta.$$
We also state two basic results about differential privacy. The first result is a composition theorem, which states that privacy degrades gracefully when we combine the results of several private computations.

Lemma 1 (Composition \cite{14,15}) Let \( \mathcal{M}_1 \) and \( \mathcal{M}_2 \) satisfy \((\varepsilon_1, \delta_1)\)- and \((\varepsilon_2, \delta_2)\)-differential privacy, respectively. Then the algorithm which on input \( x \) outputs \( \mathcal{M}_1(x) \) and \( \mathcal{M}_2(\mathcal{M}_1(x), x) \) satisfies \((\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)\)-differential privacy.

We also recall the basic Gaussian noise mechanism and its privacy guarantee.

Lemma 2 (Gaussian Mechanism \cite{10,12,14}) Let \( A \) be an \( m \times N \) matrix such that each of its columns has \( \ell_2 \) norm at most \( \sigma \). Furthermore, define

\[
c(\varepsilon, \delta) = \frac{0.5\sqrt{\varepsilon} + \sqrt{2\ln(1/\delta)}}{\varepsilon}.
\]

An algorithm which on input a histogram \( x \in \mathbb{R}^N \) outputs \( Ax + w \), where \( w \sim N(0, c(\varepsilon, \delta)^2 \sigma^2)^m \), satisfies \((\varepsilon, \delta)\)-differential privacy.

2.3 Linear Queries and Error Complexity

A query \( q : \mathcal{U}^* \to \mathbb{R} \) is linear if \( q(D) = \sum_{e \in D} q(e) \). We represent a set \( Q \) of \( m \) linear queries as a query matrix \( A \in \mathbb{R}^{m \times N} \); associating each query \( q \in Q \) with a row in \( A \) and each universe element \( e \in \mathcal{U} \) with a column, \( A \) is defined by \( A_{q,e} = q(e) \). The true answers to all queries in \( Q \) for a database \( D \) with histogram \( x \) are given by \( y = Ax \). The sensitivity of \( Q \) is defined as \( \max_{q \in Q, e \in \mathcal{U}} |q(e)| \).

We measure the error complexity of a mechanism \( M \) according to two different measures: average error and worst-case error. The mean squared error (MSE) of a mechanism \( M \) according to a distribution \( p \) on a set of queries \( Q \) is defined by

\[
\mathbb{E}_M \mathbb{E}_{q \sim p} |q(D) - \hat{y}_q|^2,
\]

where \( \hat{y} = M(D, Q) \). Notice that, by Jensen’s inequality, the square root of the MSE according to \( p \) is an upper bound on average absolute error according to \( p \):

\[
\mathbb{E}_M \mathbb{E}_{q \sim p} |q(D) - \hat{y}_q| \leq \sqrt{\mathbb{E}_M \mathbb{E}_{q \sim p} |q(D) - \hat{y}_q|^2}.
\]

The worst-case error of a mechanism \( M \) on a set of queries \( Q \) is defined by

\[
\mathbb{E}_M \max_{q \in Q} |q(D) - \hat{y}_q|,
\]

where \( \hat{y} \) is as before. For any distribution \( p \), if the worst-case error of \( M \) is \( \lambda \), then the MSE according to \( p \) is at most \( \lambda^2 \).

2.4 Marginals and Parities

In this paper we are concerned with marginals, which are a special case of linear queries. Let \( \binom{[d]}{k} \) denote the set of subsets of \([d]\) of size \( k \). For \( k \)-way marginals, the universe \( \mathcal{U} \) is the set\(^3\) \( \mathcal{U} = \{-1, 1\}^d \). Thus, each person

\(^3\) This is a simple notational switch from the usual \( \{0, 1\} \) vectors, which helps simplify notation.
is represented in the database $D$ by $d$ binary attributes. A $k$-way marginal query is specified by a set $S$ of $k$ attribute indexes, and a $\beta_i \in \{-1, 1\}$ for each $i \in S$, and is equal to the number of rows in the database for which the row vector $e$ restricted to the set of attributes $S$ takes the value given by $\beta$. More formally,

$$\text{marg}_{(S, \beta)}(D) = \sum_{e \in D} \bigwedge_{i \in S} (e_i = \beta_i).$$

It will be convenient to work with a slightly different set of queries that we call parity queries. In the same setting as above, a $k$-wise parity query is specified by a subset $S$ of $k$ attribute indexes. It is given by

$$\text{par}_S(D) = \sum_{e \in D} \prod_{i \in S} e_i.$$ 

We note that these $k$-wise parities correspond exactly to the degree-$k$ Fourier coefficients of the histogram $x$. Barak et al. [2] observed the following useful reduction from marginals to parities.

**Lemma 3** For any $S \in \binom{[d]}{k}$ and $\beta \in \{-1, 1\}^S$, there are $\{-\frac{1}{2^k}, \frac{1}{2^k}\}$ coefficients $\alpha_{S, \beta, T}$ for $T \subseteq S$ such that for all $D$

$$\text{marg}_{(S, \beta)}(D) = \sum_{T \subseteq S} \alpha_{S, \beta, T} \cdot \text{par}_T(D).$$

**Proof** Note that both the operators $\text{marg}_{(S, \beta)}$ and $\text{par}_S$ are linear. Thus it suffices to prove the statement for a database containing a single element in $\{-1, 1\}^d$. Finally observe that

$$\text{marg}_{(S, \beta)}(\{e\}) = \prod_{i \in S} \mathbf{1}(e_i = \beta_i) = \prod_{i \in S} \frac{1}{2}(1 + e_i \beta_i) = \frac{1}{2^k} \sum_{T \subseteq S} \left( \prod_{i \in T} e_i \beta_i \right) = \frac{1}{2^k} \sum_{T \subseteq S} \left( \prod_{i \in T} \beta_i \right) \cdot \text{par}_T(\{e\}).$$

The theorem below follows immediately.

**Theorem 3** Suppose that for a database $D$, for all $S \in \binom{[d]}{k}$, we have estimates $\hat{y}_S$ satisfying $|\hat{y}_S - \text{par}_S(D)| \leq \lambda$. Then we can efficiently construct estimates $\hat{z}_{S, \beta}$ for all $S \in \binom{[d]}{k}$, and $\beta \in \{-1, 1\}^S$, such that $|\hat{z}_{S, \beta} - \text{marg}_{(S, \beta)}(D)| \leq \lambda$. 

© Springer
Proof We set \( \hat{z}_{(S, \beta)} = \sum_{T \subseteq S} \alpha_{S, \beta, T} \cdot \hat{y}_T \). Thus, by the triangle inequality,
\[
|\hat{z}_{S, \beta} - \text{marg}_{(S, \beta)}(D)| = \left| \sum_{T \subseteq S} \alpha_{S, \beta, T} \cdot (\hat{y}_T - \text{par}_T(D)) \right|
\leq \sum_{T \subseteq S} |\alpha_{S, \beta, T}| \cdot |\hat{y}_T - \text{par}_T(D)|
\leq \sum_{T \subseteq S} \frac{1}{2^k} \cdot \lambda = \lambda.
\]

\[\square\]

It will also be useful to have a version of this result for mean squared error.

**Theorem 4** Let \( p \) be a distribution over \( k \)-way marginals. Then there exists a distribution \( p' \) over \( k \)-wise parities such that the following holds. Given estimates \( \hat{y}_S \) such that
\[\mathbb{E}_{S \sim p'}[|\hat{y}_S - \text{par}_S(D)|^2] \leq \lambda^2,\]
we can efficiently construct estimates \( \hat{z}_{S, \beta} \) such that
\[\mathbb{E}_{(S, \beta) \sim p}[|\hat{z}_{S, \beta} - \text{marg}_{(S, \beta)}(D)|^2] \leq \lambda^2.\]

**Proof** We define \( p' \) as follows: we sample an \((S, \beta)\) in \( p \) and sample a random \( T \subseteq S \). The estimate \( \hat{z}_{S, \beta} \) is simply defined to be \( \sum_{T \subseteq S} \alpha_{S, \beta, T} \cdot \hat{y}_T \). Now for any \((S, \beta)\)
\[
|\hat{z}_{S, \beta} - \text{marg}_{(S, \beta)}(D)|^2 = \left| \sum_{T \subseteq S} \alpha_{S, \beta, T} \cdot (\hat{y}_T - \text{par}_T(D)) \right|^2
\leq \left( \sum_{T \subseteq S} |\alpha_{S, \beta, T}|^2 \right) \cdot \left( \sum_{T \subseteq S} |\hat{y}_T - \text{par}_T(D)|^2 \right)
= 2^{-k} \cdot \left( \sum_{T \subseteq S} |\hat{y}_T - \text{par}_T(D)|^2 \right),
\]
where the inequality follows by Cauchy–Schwarz. Finally observe that when \((S, \beta)\) is drawn according to \( p \), each of the terms in the summation in the last term is distributed according to \( p' \). By linearity of expectation, the claim follows. \[\square\]

The problem of answering all \( k \)-wise parities for a database with \( d \) attributes can be easily reduced to that of answering all 2-wise parities over a database with \( d^{\lceil k/2 \rceil} + d^{\lfloor k/2 \rfloor} \) attributes, by creating a new attribute for all \( \lceil k/2 \rceil \)-wise and all \( \lfloor k/2 \rfloor \)-wise parities. Clearly if \( S \) is the disjoint union of \( S_1 \) and \( S_2 \), then \( \prod_{i \in S_1} e_i = \prod_{i \in S_1} e_i \prod_{i \in S_2} e_i \) so that a \( k \)-wise parity can be recovered as a 2-wise parity on \( 2d^{\lfloor k/2 \rfloor} \) attribute databases.

Thus in the rest of the paper, we will concern ourselves with 2-wise parity queries. When the database is in its histogram representation, these queries are represented by a matrix \( A \) with rows indexed by sets \( S \subseteq [d] \) and columns indexed by \( e \in \{-1, +1\}^d \), with \( a_{S,e} = \text{par}_S(\{e\}) \).
2.5 Convex Geometry

For a bounded set $S \subseteq \mathbb{R}^m$, we use the notation $\text{diam}(S)$ for the diameter of $S$ in $\ell_2$ norm, i.e. $\max_{x, x' \in S} \|x - x'\|_2$.

A convex body is a convex compact subset $K \subseteq \mathbb{R}^m$. A convex body $K$ is (centrally) symmetric if $-K = K$. The Minkowski norm $\|x\|_K$ induced by a symmetric convex body $K$ is defined as $\|x\|_K = \min\{r \in \mathbb{R} : x \in rK\}$.

The polar body $K^\circ$ of a convex body $K$ is defined by $K^\circ = \{y : \langle y, x \rangle \leq 1 \forall x \in K\}$.

The polar body is closely related to the standard duality between hyperplanes and points in $\mathbb{R}^m$. Recall that the duality takes a hyperplane $H = \{x : \langle y, x \rangle = 1\}$ to the point $y = D(H)$. Then the boundary of $K^\circ$ is the union of the points $D(H)$ over all supporting hyperplanes $H$ of $K$.

The Minkowski norm induced by the polar body $K^\circ$ of $K$ is the dual norm of $\|x\|_K$ and also has the form $\|y\|_{K^\circ} = \max_{x \in K} \langle x, y \rangle$. The dual norm $\|y\|_{K^\circ}$ is also known as the support function of $K$ and is often denoted as $h_K(y)$. The dual norm is useful in arguing about the width of $K$. In particular, if $u$ is a vector of unit Euclidean norm, then $\|u\|_{K^\circ}$ is equal to the Euclidean distances from the origin 0 to the supporting hyperplane of $K$ orthogonal to the line through $u$. For a general point $y \in \mathbb{R}^m$, we call $\|y\|_{K^\circ}$ the width of $K$ with respect to $y$.

The mean (Gaussian) width $\ell^*(K)$ of $K$ is the expected width of $K$ with respect to a random Gaussian, i.e. $\ell^*(K) = \mathbb{E} \|g\|_{K^\circ}$, where $g \sim N(0, 1)^m$.

For convex symmetric $K$, the induced norm and the dual norm satisfy Hölder’s inequality:

$$|\langle x, y \rangle| \leq \|x\|_K \|y\|_{K^\circ}. \tag{1}$$

Geometric approaches to differential privacy rely on studying the geometry of the sensitivity polytope $K = AB_1$, where $A$ is the query matrix of family $Q$ of linear queries. The body $K$ is the symmetric convex hull of all possible vectors of answers $y$ to the queries $Q$ for a database of size 1. The name sensitivity polytope is motivated by the fact that $K$ is the smallest convex body such that for any neighboring databases $x, x', Ax - Ax' \in K$. Since the queries are linear, it is easy to see that $nK = AB_1(n)$ is the symmetric convex hull of all possible vectors of answers $y$ for a database of size at most $n$.

3 The Projection Algorithm and Relaxations

A central tool in the present work is an algorithm for linear queries, first proposed in [29], which is simply the well-known Gaussian noise mechanism combined with a post-processing step. The post-processing, a projection onto $nK$, is the computationally expensive step of the algorithm. Here, in order to implement this step efficiently,
we modify the algorithm from [29] to project onto a relaxation of $nK$, and we compute an approximate projection using the Frank–Wolfe convex minimization algorithm.

3.1 Frank–Wolfe

In this section we recall the classical constrained convex minimization algorithm of Frank and Wolfe [18], which allows us to reduce computing an approximate projection onto a convex body to solving a small number of linear maximization problems. The algorithm is presented as Algorithm 1.

### Algorithm 1 FRANKWOLFE

**Input** convex body $F \subseteq \mathbb{R}^m$; point $r \in \mathbb{R}^m$; number of iterations $T$

Let $q^{(0)} \in F$ be arbitrary.

for $t = 1$ to $T$

Let $v^{(t)} = \arg \max_{v \in F} \langle r - q^{(t-1)}, v \rangle$.

Let $\alpha^{(t)} = \arg \min_{\alpha \in [0,1]} \| r - \alpha q^{(t-1)} - (1 - \alpha) v^{(t)} \|_2$.

Set $q^{(t)} = \alpha^{(t)} q^{(t-1)} + (1 - \alpha^{(t)}) v^{(t)}$.

end for

**Output** $q^{(T)}$.

We use the following bound on the convergence rate of the Frank–Wolfe algorithm. It is a refinement of the original analysis of Frank and Wolfe, due to Clarkson.

**Theorem 5** (Frank–Wolfe [8,18]) Let $q^* = \arg \min_{q \in F} \| r - q \|_2^2$ be the projection of $r$ onto the convex body $F$. Then the point $q^{(T)}$ computed by $T$ iterations of FRANKWOLFE satisfies

$$
\| r - q^{(T)} \|_2^2 \leq \| r - q^* \|_2^2 + \frac{4 \text{diam}(F)^2}{T + 3}.
$$

Moreover, $q^{(T)}$ lies in the convex hull of $q^{(0)}, v^{(1)}, \ldots, v^{(T)}$.

The expensive step in each iteration is computing $v^{(t)}$, which requires solving a linear optimization problem over $F$. Computing $\alpha^{(t)}$ is a quadratic optimization problem in a single variable, and has a closed-form solution. The sparsity property of $q^{(T)}$ (the claim after ‘moreover’) turns out to be useful in reductions from worst-case to average error via boosting.

3.2 Projection onto a Relaxation

Let us consider a query matrix $A$ which is given only implicitly, e.g. the $k$-way parities matrix. More generally, we have the following definition.

**Definition 2** An $m \times N$ query matrix $A$ for a set of linear queries $Q(|Q| = m)$ over a universe $\mathcal{U}(|\mathcal{U}| = N)$ is efficiently represented if for each $q \in Q$, and each $e \in \mathcal{U}$, $A_{q,e}$ can be computed in time polynomial in $m$ and $\log N$. 
Given an efficiently represented \( A \), can we approximate \( Ax \) with additive error close to \( \sqrt{n} \) in time \( \text{poly}(n, \log N) \)? Using the Frank–Wolfe algorithm, and the geometric methods of Nikolov et al. [29], this problem can be reduced to \( \text{poly}(n, \log N, \text{diam}(K)) \) calls to a procedure solving the optimization problem \( \max_{v \in K} \langle u, v \rangle \), where \( K = AB_1 \). While this may be a hard problem to solve, fortunately, the analysis of the algorithm in [29] is flexible and it is enough to be able to solve the problem for a relaxation \( L \) of \( K \). Moreover, we need a relatively weak guarantee on \( L \): it should have Gaussian width comparable to that of \( K \), and diameter that is polynomially bounded.

Next we define this modification of the algorithm and the notion of relaxation that is useful to us.

**Algorithm 2** \( \text{RELAXEDPROJ}_L \)

**Input** (Public) efficiently represented query matrix \( A = (a_i)_{i=1}^N \in [-1, 1]^{m \times N} \); a convex body \( L \subseteq \mathbb{R}^m \); distribution \( p = (p_i)_{i=1}^m \) on \([m]\); number of iterations \( T \).

**Input** (Private) database \( x \in \mathbb{R}^N, \|x\|_1 = n \)

Let \( P = \text{diag}(p) \).

Let \( c(\epsilon, \delta) = \frac{1 + \sqrt{2 \ln(1/\delta)}}{\epsilon} \).

Sample \( w \sim N(0, c(\epsilon, \delta)^2 m)^m \);

Let \( \tilde{y} = P^{1/2} Ax + w \).

Let \( \bar{y} \) be the output of \( T \) iterations of \( \text{FRANKWOLFE} \) with input the convex body \( F = nP^{1/2}L \) and the point \( r = \tilde{y} \).

**Output** \( \hat{y} = P^{-1/2} \bar{y} \) (with the convention \( 0^{-1/2} = 0 \)).

**Definition 3** A convex body \( L \subseteq \mathbb{R}^m \) is an efficient relaxation of \( K = AB_1 \subseteq \mathbb{R}^m \), where \( A \in \mathbb{R}^{m \times N} \), if \( K \subseteq L \), and for any \( u \in \mathbb{R}^m \) and any \( \beta > 0 \), the optimal solution of the maximization problem \( \max_{v \in L} \langle u, v \rangle \) can be approximated to within \( \beta \) in time polynomial in \( \log \frac{1}{\beta}, m, \log \|u\|_{\infty}, \) and \( \log N \).

Notice that if \( L \) is an efficient relaxation of \( K \), then \( QL \) is an efficient relaxation of \( QK \) for any matrix \( Q \) with sufficiently well-bounded entries.

**Theorem 6** Let \( p \) be a probability distribution on \([m]\) and let \( P = \text{diag}(p) \). Let \( L \) be an efficient relaxation of \( K = AB_1 \), and finally let

\[
T = \frac{4n \text{diam}(P^{1/2}L)^2}{c(\epsilon, \delta)\ell^*(L)}.
\]

Then algorithm \( \text{RELAXEDPROJ}_L \)

1. satisfies \( (\epsilon, \delta) \)-differential privacy;
2. can be implemented in time polynomial in \( m, n, \log N, \) and \( \text{diam}(P^{1/2}L) \);
3. outputs a point \( \hat{y} \) in \( n \cdot \Pi_{\text{supp}(p)} L \), where \( \Pi_{\text{supp}(p)} \) is a coordinate projection onto the support of \( p \);
4. has MSE with respect to \( p \) at most

\[
\mathbb{E} \sum_{i=1}^m p_i |y_i - \hat{y}_i|^2 = O(c(\epsilon, \delta)n\ell^*(P^{1/2}L)).
\]
5. there exists a constant $C$ s.t. for any $t > 0$, with probability at least $1 - \exp(t^2/4\ diam(P^{1/2}L)^2)$,

$$\sum_{i=1}^{m} p_i |y_i - \hat{y}_i|^2 \leq C \cdot c(\varepsilon, \delta)n(\ell^*(P^{1/2}L) + t).$$

**Proof** We first prove claim 1. Since $A \in [-1, 1]^{m \times N}$, and $\sum p_i = 1$, for any column $a_j$ of $A$, we have $\|P^{1/2}a_j\|_2 \leq 1$. Then by Lemma 2, $\tilde{y}$ is $(\varepsilon, \delta)$-differentially private. The output $\hat{y}$ is a function only of $\tilde{y}$ and not of the private data $x$, and is therefore $(\varepsilon, \delta)$-differentially private by Lemma 1.

It is easy to verify that $\tilde{y}$ can be computed in time polynomial in $m$, $n$, and $\log N$ given an efficiently represented $A$. Then claim 2 follows since, for an efficient relaxation $L$, each step of FrankWolfe can be implemented in time polynomial in $m$, $\log \diam(P^{1/2}L)$, and $N$ for the chosen value of $T$.

To prove claim 3, note that (with the convention used in the algorithm that $0^{1/2} = 0$) $P^{-1/2}P^{1/2}$ is in fact the coordinate projection $\Pi_{\text{supp}(p)}$, and $\hat{y} \in P^{-1/2}(P^{1/2}K)$.

The central claim is the MSE bound in claim 4. The proof of this bound follows essentially from [29] and appears to be a standard method of analyzing least squares estimation in statistics. The key observation we make in this work is that an efficient relaxation with well-bounded mean width is sufficient for the proof to go through. We give the full proof next for completeness.

Let $\tilde{w} = \tilde{y} - P^{1/2}y$ and recall $w = \check{y} - P^{1/2}y$. Since $\check{y}_i = p_i^{-1/2}\hat{y}_i$ for each $i$ (again using the convention $0^{-1/2} = 0$),

$$\sum_{i=1}^{m} p_i |y_i - \check{y}_i|^2 = \sum_{i=1}^{m} |p_i^{1/2}y_i - \check{y}_i|^2 = \|P^{1/2}y - \check{y}\|_2^2 = \|\tilde{w}\|_2^2.$$

Therefore it is enough to bound $\mathbb{E}_w\|\tilde{w}\|_2^2$.

The bound, proved below, is based on Hölder’s inequality and the following fact:

$$\|\tilde{w}\|_2^2 = \langle \tilde{w}, w \rangle + \langle \tilde{w}, \tilde{w} - w \rangle \leq 2\langle \tilde{w}, w \rangle + \nu, \quad (2)$$

where $\nu$ is defined to be $c(\varepsilon, \delta)n\ell^*(P^{1/2}L)$ and measures the quality of the approximation to the projection of $\check{y}$ onto $n P^{1/2}L$ returned by $T$ rounds of the Frank–Wolfe algorithm. In particular, by Theorem 5, for the choice of $T$ in the statement of the theorem, we have

$$\|\check{y} - \tilde{y}\|_2^2 \leq \min_{z \in \text{supp}(p)} \|\check{y} - z\|_2^2 + \nu. \quad (3)$$

Since $P^{1/2}y \in n P^{1/2}K \subseteq n P^{1/2}L$, (3) implies $\|w - \tilde{w}\|_2^2 \leq \|w\|_2^2 + \nu$. The inequality (2) then follows from $\langle \tilde{w}, \tilde{w} - w \rangle \leq \langle \tilde{w}, w \rangle + \nu$, which is proved as follows:

$$\langle \tilde{w}, w \rangle = \|w\|_2^2 + \langle w, \tilde{w} - w \rangle$$

$$\geq \|w - \tilde{w}\|_2^2 - \nu + \langle w, \tilde{w} - w \rangle$$

$$= \langle \tilde{w}, \tilde{w} - w \rangle - \nu.$$
Observe that $\tilde{y}, P^{1/2} y \in n P^{1/2} L$, and, therefore, by the triangle inequality, $\|\tilde{w}\|_{P^{1/2} L} \leq 2n$. This observation, inequality (2), and Hölder’s inequality (1) imply

$$\|\tilde{w}\|_2^2 \leq 2\langle \tilde{w}, w \rangle + v$$
$$\leq 2\|\tilde{w}\|_{P^{1/2} L} \|w\|_{(P^{1/2} L)\circ} + v$$
$$\leq 4n\|w\|_{(P^{1/2} L)\circ} + v.$$ 

Since $w \sim N(0, c(\varepsilon, \delta)^2)^m$, $\frac{1}{c(\varepsilon, \delta)} w \sim N(0, 1)^m$, and, therefore, $\mathbb{E} \|w\|_{(P^{1/2} L)\circ} = c(\varepsilon, \delta)\ell^*(P^{1/2} L)$. Recalling that $v = c(\varepsilon, \delta)n\ell^*(P^{1/2} L)$, we write

$$\mathbb{E} \sum_{i=1}^m p_i |y_i - \hat{y}_i|^2 = \mathbb{E} \|\tilde{w}\|_2^2 \leq 4n\mathbb{E} \|w\|_{(P^{1/2} L)\circ} + v$$
$$= O(c(\varepsilon, \delta)n\ell^*(P^{1/2} L)),$$

completing the proof of claim 4.

For the proof of claim 5, observe that the function $\|g\|_{(P^{1/2} L)\circ}$ is a Lipschitz function of $g$ with Lipschitz constant $\frac{1}{2} \text{diam}(P^{1/2} L)$. Indeed,

$$\|g\|_{(P^{1/2} L)\circ} - |g'|_{(P^{1/2} L)\circ} = \max_{v \in P^{1/2} L} \langle v, g \rangle - \max_{v \in P^{1/2} L} \langle v, g' \rangle$$
$$= \max_{v \in P^{1/2} L} |\langle v, g - g' \rangle|$$
$$\leq \frac{1}{2} \text{diam}(P^{1/2} L) \cdot \|g - g'\|_2.$$

Then the tail bound in claim 5 follows from the Gaussian isoperimetric inequality (see e.g. [11]).

\[ \square \]

4 Grothendieck’s Inequality and Marginals

In this section we instantiate Theorem 6 with an efficient relaxation $L$ of $K = AB_1$, where $A$ is the pairwise parity queries matrix. It will be convenient to extend $A$ so that it has one row for each pair $(i, j) \in [d]^2$ (rather than one row for each 2-element set $\{i, j\} \in \binom{[d]}{2}$): this corresponds to just taking two copies of the original query matrix. We will work with the extended $A$ and the corresponding $K = AB_1$ for the remainder of the section.

Notice that $K = \text{conv}\{ \pm a_e \} = \text{conv}\{ \pm e \otimes e : e \in \{ \pm 1 \}^d \}$. Let us consider the convex body $L_0 = \text{conv}\{ w \otimes z : w, z \in \{ \pm 1 \}^d \}$. It is immediate that $K \subseteq L_0$.

**Lemma 4** For all $k$, $K \subseteq L_0$, and moreover, for any distribution $p$ on $[d]^2$ and $P = \text{diag}(p)$, $\ell^*(P^{1/2} L_0) \leq d^{1/2}$ and $\text{diam}(P^{1/2} L_0) \leq 1$.

**Proof** By the definition of $\ell^*$ and since a linear function is always maximized at an extreme point of a convex set,

$$\ell^*(P^{1/2} L_0) = \mathbb{E}_g \max_{v \in P^{1/2} L_0} \langle g, v \rangle = \mathbb{E}_g \max_{w, z \in \{ \pm 1 \}^d} \langle g, P^{1/2} (w \otimes z) \rangle,$$

\[ \square \] Springer
where expectations are taken over \( g \sim N(0, 1)^m \). Let us fix some \( w \) and \( z \). Since \( w \otimes z \) is a vector in \( \{ \pm 1 \}^{d/2} \times [d] \), \( \| P^{1/2}(w \otimes z) \|_2^2 = \sum p_i = 1 \) (this proves the bound on \( \text{diam}(P^{1/2}L_0) \)). By stability of Gaussians, \( \langle g, P^{1/2}(w \otimes z) \rangle \sim N(0, 1) \). Then

\[
\max_{w, z \in \{ \pm 1 \}^d} \langle g, P^{1/2}(w \otimes z) \rangle
\]

is the maximum of \( 2^d \) standard Gaussian random variables. By standard arguments, the expectation of this maximum is at most \( O(\sqrt{\log 2^d}) = O(d^{1/2}). \)

The relaxation \( L_0 \) is not efficient, as maximizing a linear function over \( L_0 \) is \( \text{NP} \)-hard.\(^5\) However, we can view the problem of maximizing a linear function over \( L_0 \) as the problem of computing the \( \| \cdot \|_{\infty \to 1} \) norm of an associated matrix and this norm is well approximated by the optimum of a convex relaxation \([1, 20]\). This connection, which we explain next, allows us to relax \( L_0 \) further to an efficient relaxation.

**Definition 4** For given \( d \), the convex body \( L \) consists of all vectors \( h \in \mathbb{R}^{d \times [d]} \) such that there exist sequences of unit vector \( (u_i)_{i \in [d]}, (v_j)_{j \in [d]} \) in \( \ell_2 \) for which \( h(i, j) = \langle u_i, v_j \rangle \) for all \( i, j \in [d] \).

We show that \( L \) is an efficient relaxation using semidefinite programming techniques, and therefore it can be used in \( \text{RELAXEDPROJ}_L \). Then, in order to give error guarantees for \( \text{RELAXEDPROJ}_L \), it would be enough to show that \( \ell^*(P^{1/2}L) \) is not much larger than \( \ell^*(P^{1/2}L_0) \) for any distribution \( p \). A much stronger property—\( L_0 \subseteq L \subseteq CL_0 \) for a constant \( C \)—is implied by Grothendieck’s inequality, a classical result in functional analysis. The following formulation of the inequality is due to Lin-denstrauss and Pelczynski [28].

**Theorem 7** [20] There exists an absolute constant \( C \) such that for any \( d \times d \) real matrix \( M \),

\[
\max_{(u_i)_{i=1}^d, (v_j)_{j=1}^d} \sum_{i, j} M_{ij} \langle u_i, v_j \rangle \leq C \max_{w, z \in \{ \pm 1 \}^d} w^T M z,
\]

where the maximum on the left-hand side ranges over sequences of unit vectors \( (u_i)_{i=1}^d, (v_j)_{j=1}^d \) in \( \ell_2 \).

The following lemma follows from Theorem 7 and the existence of efficient optimization algorithms for semidefinite programming [21].

**Lemma 5** \( L \) is an efficient relaxation of \( L_0 \) and therefore of \( K \). Moreover, there exists a constant \( C \) such that for every square matrix \( Q \), we have \( \ell^*(QL) \leq C \ell^*(QL_0) \) and \( \text{diam}(QL) \leq \sqrt{\text{tr}(Q^2)} \).

**Proof** Let us first prove that \( L \) is an efficient relaxation of \( L_0 \). To show that \( L_0 \subseteq L \), it is enough to argue that all extreme points of \( L_0 \) are in \( L \). Take any \( w \otimes z \in L_0 \). Define

\(^5\) For example, there is an easy reduction from the maximum cut problem. We omit the details.

\(^6\) The \( p \) to \( q \) norm of a matrix \( M \) is defined as \( \| M \|_{p \to q} = \max_{x : \| x \|_p = 1} \| M x \|_q \).
the unit vectors \((u_i)\) and \((v_j)\) to be just the one-dimensional vectors \((w_i)\), \((z_j)\); since 
\((w \otimes z)_{(i,j)} = w_i z_j\), we have shown the inclusion \(w \otimes z \in L\).

Let us now argue that linear functions can be efficiently optimized over \(L\). Given a point \(h \in \mathbb{R}^{d \times d}\), define the \(d \times d\) matrix \(H\) by \(H_{i,j} = h_{(i,j)}\). Then \(L\) is in a one-to-one correspondence with the convex set of matrices \(H \in \mathbb{R}^{d \times d}\) that can be extended to a positive semidefinite \(H' \in \mathbb{R}^{2d \times 2d}\). This shows that for any \(g \in \mathbb{R}^{d \times d}\), the maximization problem

\[
\max_{h \in L} \langle g, h \rangle = \max_{h \in L} \tr(G^T H) = \max_{(u_i), (v_j)} \sum_{i,j} G_{i,j} \langle u_i, v_j \rangle
\]

is a semidefinite program, and therefore can be solved to within arbitrary accuracy in polynomial time [21]. Above \(i\) and \(j\) range over \([d]\), and \((u_i), (v_j)\) are sequences of unit vectors in Hilbert space. Also, \(G\) is defined by \(G_{i,j} = g_{(i,j)}\).

To bound \(\ell^*(QL)\), it is enough to show that for any \(g \in \mathbb{R}^{d \times d}\), \(\|g\|_{(QL)}^\circ \leq C\|g\|_{(QL_0)}^\circ\). (In fact by duality, this establishes the stronger result \(L \subseteq CL_0\).) We have

\[
\|g\|_{(QL_0)}^\circ = \max_{w, z \in \{\pm 1\}^d} \langle g, Q(w \otimes z) \rangle = \max_{w, z \in \{\pm 1\}^d} \langle Q^T g, w \otimes z \rangle.
\]

Define \(g' = Q^T g\), and define the \(d \times d\) matrix \(G'\) by \(G'_{i,j} = g'_{(i,j)}\), where \(i\) and \(j\) range over \([d]\). Then we have

\[
\|g\|_{(QL_0)}^\circ = \max_{w, z \in \{\pm 1\}^d} \tr((G')^T w z^T) = \max_{w, z \in \{\pm 1\}^d} \tr(z w^T G') = \max_{w, z \in \{\pm 1\}^d} w^T G' z. \tag{4}
\]

By an analogous argument, we derive the identity

\[
\|g\|_{(QL)}^\circ = \max_{(u_i), (v_j)} \sum_{s, t \in [d]^{k/2}} G'_{i,j} \langle u_i, v_j \rangle, \tag{5}
\]

where \((u_i)\) and \((v_j)\) are sequences of \(d\) unit vectors in Hilbert space. The first part of the lemma then follows from (4), (5), and Theorem 7.

For the diameter bound, we note that for any point \(h\) in \(L\), each coordinate of \(h\) is the dot product of two unit vectors and hence bounded in absolute value by 1. \(\square\)

Combining the results above gives our main theorem.

**Theorem 8** There exists an \((\varepsilon, \delta)\)-differentially private mechanism \(M\) that, given any (public) distribution \(p\) on 2-wise parity queries and (private) database \(D\), computes answers \((\hat{y}_S)_{S : |S| = 2} = M(p, D)\) with MSE with respect to \(p\)

\[
\sqrt{\mathbb{E}_M \mathbb{E}_{S \sim p} | \text{par}_S(D) - \hat{y}_S |^2} \leq C \cdot c(\varepsilon, \delta)^{1/2} \sqrt{nd}^{1/4},
\]

for a universal constant \(C\). Additionally, for any \(t > 0\), with probability at least \(1 - \exp(-t^2/4)\),

\[
\mathbb{E}_{S \sim p} | \text{par}_S(D) - \hat{y}_S |^2 > C \cdot c(\varepsilon, \delta)n(d^{1/2} + t).
\]
Moreover, $\mathcal{M}$ runs in time $\text{poly}(d, n)$.

**Proof** The mechanism $\mathcal{M}$ runs $\text{RELAXEDPROJ}_L$ with the choice of the number of iterations $T$ as in Theorem 6. By Theorem 6 and Lemma 5, $\mathcal{M}$ runs in time polynomial in $m = d^2$ and $n$. Also by Theorem 6, $\mathcal{E}_s \mathbb{E}_{S \sim p} |\text{par}_S(D) - \hat{y}_S|^2 = O(c(\varepsilon, \delta) n \ell^*(P^{1/2}L))$. By Lemmas 4 and 5, $\ell^*(P^{1/2}L) \leq C \ell^*(P^{1/2}L_0) = O(d^{1/2})$. Plugging this into (6) and taking square roots completes the analysis of the expected MSE. The tail bound follows analogously.

5 Worst-Case Error and Boosting

At a relatively small cost in error and computational complexity, we can strengthen the guarantees of Theorem 8 from MSE bounds for every query distribution to worst-case error bounds. We do this via the private boosting framework of Dwork et al. [17].

5.1 The Boosting for Queries Framework

The boosting for queries framework of Dwork et al. assumes black-box access to a base synopsis generator: a private mechanism that, given a set of queries sampled from some probability distribution and a private database, produces a data structure (the synopsis) that can be used to answer a strong majority of the queries with error at most $\lambda$. The boosting algorithm runs the synopsis generator several times and produces a new data structure that can be used to answer all queries with error $\lambda + \mu$, where $\mu$ is a term that scales with the bit size of the synopsis produced by the base generator. Next we define a base generator formally and give the statement of the main result from [17].

**Definition 5** A mechanism $\mathcal{M}$ is a $(\kappa, \lambda, \beta)$-base synopsis generator for a set of queries $Q$, if there exists a reconstruction algorithm $R$ such that the following holds. For any distribution $p$ on $Q$, and any private database $D$, when $S$ is a multiset of $\kappa$ queries sampled independently with replacement from $Q$, $\hat{D} = \mathcal{M}(S, D)$ satisfies

$$\Pr_{\mathcal{M}, S}[p(|q : |q(D) - R(\hat{D}, q)| \geq \lambda]) > 1/3] < \beta.$$ 

**Theorem 9** [17] Let $Q$ be a set of $|Q| = m$ linear queries with sensitivity 1, and let $T = C \log m$ for a large enough constant $C$. There exists a mechanism $\mathcal{M}$ that, given access to an $(\varepsilon_0, \delta_0)$-differentially private $(\kappa, \lambda, \beta)$-base synopsis generator $\mathcal{M}^{\text{base}}$, satisfies $(\varepsilon + T \varepsilon_0, T(\kappa \beta + \delta_0))$-differential privacy and, for any private database $D$, in time polynomial in $m$ and the running time of $\mathcal{M}^{\text{base}}$, with probability at least $(1 - T \beta)$ outputs answers $(q^*(D))_{q \in Q}$ such that

$$\forall q \in Q : |q^*(D) - q(D)| \leq \lambda + \mu,$$
for 
\[
\mu = O\left(\frac{\sqrt{\kappa} \log^{3/2} m \sqrt{\log 1/\beta}}{\varepsilon}\right).
\]

The term \( \mu \) in Theorem 9 is an error overhead due to the privacy requirements of the boosting algorithm. To minimize this overhead, we need to make the number \( \kappa \) of queries given to the base generator as small as possible. A generalization result proved for the uniform distribution in [16] and extended to arbitrary distributions in [17] shows that it is sufficient to make \( \kappa \) only a constant factor larger than the bit size of the synopsis. We reproduce a version of this argument with a slightly weaker assumption.

**Lemma 6** Suppose there exists a mechanism \( \mathcal{M} \) and a reconstruction algorithm \( \mathcal{R} \) such that, given any distribution \( \tilde{\nu} \) on queries \( Q \), and a private database \( D \), \( \hat{D} = \mathcal{M}(\tilde{\nu}, D) \) satisfies the MSE bound

\[
\sqrt{\sum_{q \in Q} \tilde{\nu}(q)|q(D) - \mathcal{R}(\hat{D}, q)|^2} \leq \lambda,
\]

with probability \( 1 - \beta \), and, moreover, for all \( D \), \( \hat{D} \) can be represented by a string of \( s \) bits. Then \( \mathcal{M} \) is a \( (O(s + \log 1/\beta), O(\lambda), 2\beta) \)-base synopsis generator for \( Q \).

**Proof** Let \( p \) be an arbitrary distribution on \( Q \), and let \( S \) be a multiset of \( \kappa \) queries sampled independently with replacement from \( Q \). Let \( \tilde{\nu} \) be the empirical distribution given by \( S \), i.e. \( \tilde{\nu}(q) \) is equal to the number of copies of \( q \) in \( S \) divided by \( \kappa \). Define \( Q_{\text{bad}}(\hat{D}, D) = \{ q : |q(D) - \mathcal{R}(\hat{D}, q)| \geq \sqrt{6}\lambda \} \). Fix some \( \hat{D} \) such that \( p(Q_{\text{bad}}(\hat{D}, D)) > 1/3 \). Then \( \mathbb{E}_S \tilde{\nu}(Q_{\text{bad}}(\hat{D}, D)) > 1/3 \) and by Chernoff’s inequality

\[
\Pr_S[\tilde{\nu}(Q_{\text{bad}}(\hat{D}, D)) \leq 1/6] \leq 2^{-\kappa/C}
\]

for a constant \( C \). Setting \( \kappa = C(s + \log 1/\beta) \), we get that \( \tilde{\nu}(Q_{\text{bad}}(\hat{D}, D)) \leq 1/6 \) with probability at most \( \beta 2^{-s} \); by a union bound over all choices of \( \hat{D} \), with probability \( 1 - \beta \) over the random choice of \( S \), \( \tilde{\nu}(Q_{\text{bad}}(\hat{D}, D)) > 1/6 \) for all \( \hat{D} \) such that \( p(Q_{\text{bad}}(\hat{D}, D)) > 1/3 \).

By Markov’s inequality and the MSE guarantee of \( \mathcal{M} \), with probability \( 1 - \beta \) over the randomness of \( \mathcal{M} \), \( \hat{D} = \mathcal{M}(\tilde{\nu}, D) \) satisfies \( \tilde{\nu}(\{ q : |q(D) - \mathcal{R}(\hat{D}, q)| \geq \sqrt{6}\lambda \}) \leq 1/6 \). Therefore, except with probability \( \beta \), for \( \hat{D} = \mathcal{M}(S, D) \), \( \tilde{\nu}(Q_{\text{bad}}(\hat{D}, D)) \leq 1/6 \). From this fact and the discussion above, we conclude that with probability \( 1 - 2\beta \) (over both the randomness of \( \mathcal{M} \) and the random choice of \( S \)), \( p(Q_{\text{bad}}(\hat{D}, D)) \leq 1/3 \) for \( \hat{D} = \mathcal{M}(\tilde{\nu}, D) \).

**5.2 Generating a Concise Synopsis**

Lemma 6 and Theorem 9 together imply that the additional error \( \mu \) incurred by boosting the MSE guarantee to a worst-case guarantee can be made nearly as small as \( \sqrt{s} \), where \( s \) is the size of the base synopsis in bits.

\( \odot \) Springer
Next we show how to modify RELAXEDPROJ so that it produces a synopsis small enough to make this additional error comparable to the MSE bound which we have already proved.

Let us consider the body $L$ of Definition 4. Without modification, RELAXEDPROJ$_L$, called with a distribution $\tilde{\rho}$, outputs a point $\hat{y} \in n \cdot \Pi_{\text{supp}(\tilde{\rho})} L$. Thus for any $i, j \in [d]$, such that $(i, j) \in \text{supp}(\tilde{\rho})$, $\hat{y}_{(i,j)} = n(u_i, v_j)$, where $u_i, v_j \in \mathbb{R}^m$ are unit vectors. It is a relatively standard fact that these vectors can be projected down to about $O(\log m)$ dimensions such that each of the $m$ dot products is preserved up to a small constant additive error. We will need subconstant error and will thus have to take many more dimensions. We first give a formal statement of the guarantee given by the Johnson–Lindenstrauss lemma.

**Lemma 7** Let $u$ and $v$ be unit vectors in $\mathbb{R}^M$ and let $\Pi$ be a $M' \times M$ matrix with entries drawn independently from $N(0, 1/M')$ for $M' < M$. Then $\mathbb{E}[(\langle \Pi u, \Pi v \rangle) = \langle u, v \rangle$ and for any $t \in (0, 1)$,

$$\Pr[|\langle \Pi u, \Pi v \rangle - \langle u, v \rangle| > 3t] \leq 6 \cdot \exp(-M't^2/6).$$

**Proof** By the Johnson–Lindenstrauss Lemma (e.g. [9]), for any vector $w$,

$$\Pr[\|\Pi w\|^2 - \|w\|^2 \geq t\|w\|^2] \leq 2\exp(-M't^2/6).$$

Conditioning on the event $\|\Pi w\|^2 - \|w\|^2 \leq t\|w\|^2$ for $w \in \{u, v, (u + v)\}$, and observing that $\|u\|^2, \|v\|^2 = 1$ and $\|u + v\|^2 \leq 4$, we write

$$2|\langle \Pi u, \Pi v \rangle - \langle u, v \rangle| = |(\|\Pi u + \Pi v\|^2 - \|\Pi u\|^2 - \|\Pi v\|^2)$$

$$- (\|u + v\|^2 - \|u\|^2 - \|v\|^2)|$$

$$\leq t\|u + v\|^2 + t\|u\|^2 + t\|v\|^2$$

$$\leq 6t.$$

This completes the proof. \hfill \Box

In our setting, we wish to preserve $m = O(d^2)$ dot products approximately, with probability $(1 - \beta)$. Suppose that for a parameter $\chi$, we set $t = \chi d^{\frac{1}{2}}/\sqrt{n}$, and $M' = 12 \cdot (2\log d + \log 1/\beta)/t^2$. Then, by a union bound, with probability $(1 - \beta)$, a random projection $\Pi$ will satisfy $|\langle \Pi u_i, \Pi v_j \rangle - \langle u_s, v_t \rangle| \leq 6t$ simultaneously for all $u_i, v_j$. Also note that this gives us

$$M' = 12 \cdot (2\log d + \log 1/\beta)/t^2$$

$$= 12n \cdot (2\log d + \log 1/\beta)/(\chi^2 d^{1/2}).$$

We will let our synopsis be defined as the collection of vectors $\{\Pi u_i\}, \{\Pi v_j\}$, with each coordinate being represented to $(\log n + \log M')$ bits of precision. Note that this truncation at the $(\log n + \log M')$th bit adds at most a $1/n$ additive error to the dot
product. Also recalling that \( \hat{y}_{(i,j)} = n \langle u_i, v_j \rangle \), we set \( \hat{y}'_{(i,j)} \) to be \( n \langle \Pi u_i, \Pi v_j \rangle \). Thus except with probability \( \beta \), every pair \((i, j)\) satisfies

\[
|\hat{y}'_{(i,j)} - \hat{y}_{(i,j)}| \leq nt = \chi \cdot \sqrt{n}d^{1/4}.
\]

Theorem 8 then implies the following.

**Lemma 8** Suppose that \( \beta \in (\exp(-n), d^{-1/2}n^{-2}) \), and \( n \leq d^{1/2} \). For any \( \chi > 1 \), there exists a mechanism \( \mathcal{M} \) and a reconstruction algorithm \( \mathcal{R} \) such that, given any distribution \( \tilde{p} \) on 2-wise parity queries, and a private database \( D \), \( \hat{D} = \mathcal{M}(\tilde{p}, D) \) satisfies the MSE bound

\[
\sqrt{\sum_q \tilde{p}(q)|q(D) - \mathcal{R}(\hat{D}, q)|^2} \leq C \cdot \sqrt{n} \cdot d^{1/4}(c(\epsilon, \delta)^{1/2} + \chi),
\]

with probability \( 1 - \beta \), for some absolute constant \( C \). Moreover, for all \( D \), \( \hat{D} \) can be represented by a string of \( 48n \cdot d^{1/2} \cdot \log(1/\beta) / \chi^2 \) bits.

A similar compression result holds for the general case of arbitrary queries. Assume we have a set of \( m \) queries \( Q \) with coefficients in \([-1, 1] \), and let \( L \) be an efficient relaxation for the corresponding sensitivity polytope \( K \) so that \( L \subseteq [-r, r]^m \). Theorem 5 implies that after \( T = O(\text{diam}(nL)^2 / \nu) \) iterations, the Frank–Wolfe algorithm approximates the optimal projection onto \( nL \) within an additive \( \nu \) in squared distance. This translates to additional squared error \( \nu \). Moreover, the output after \( T \) iterations is a convex combination of \( T \) extreme points of \( L \). If we have an \( L \) such that each extreme point of \( L \) has a concise representation, we get a concise representation for \( \tilde{y} \) as well. The representation consists of the \( T \) extreme points, each represented as \( s \) bits, and their coefficients in the convex combination that gives \( \tilde{y} \), rounded off to sufficient accuracy. Overall, the total number of bits used is \( O(Ts + T \log(Tr / \nu)) \).

### 5.3 Putting it Together

Combining Lemma 6 with Lemma 8, we conclude that if \( \beta \in (\exp(-n), d^{-1/2}n^{-2}) \), and \( n \leq d^{1/2} \), then for any \( \chi > 1 \), there is a mechanism with running time polynomial in \( n \) and \( d \) that is a \((\kappa, \lambda, \beta)\)-base synopsis generator, with

\[
\kappa = 48n \cdot d^{1/2} \cdot \log(1/\beta)(\log n) / \chi^2.
\]

\[
\lambda = C \cdot \sqrt{n} \cdot d^{1/4}(c(\epsilon, \delta)^{1/2} + \chi).
\]

Plugging this into Theorem 9, we get our main result for worst-case error.

**Theorem 10** Let \( 2^{-n} \leq \delta \leq n^{-2} \) and \( d \leq \exp(n) \). There exists an \((\epsilon, \delta)\)-differentially private mechanism \( \mathcal{M} \) that, given any database \( D \), with constant probability computes answers \( (\hat{y}_S)_{S:|S|=2} = \mathcal{A}(D) \) with worst-case error \( \max_S |\text{par}_S(D) - \hat{y}_S| \) bounded by
\[ O\left(\sqrt{n} \cdot d^{1/4} \cdot (\log d + \log 1/\delta)^{1/2} (\log n)^{1/4} (\log d)^{3/4} / \varepsilon^{1/2}\right). \]

Moreover, \( \mathcal{M} \) runs in time \( \text{poly}(d, n) \).

Proof We set \( T = C \log m = C \log d, \beta = \delta / (\kappa T) \), and use the \((\kappa, \lambda, \beta)\)-base synopsis generator with privacy parameters \((\varepsilon / T, \delta / T)\). If \( n \geq d^{1/2} \), then the required error bound is achieved by the Gaussian noise mechanism. Thus we can assume that \( n \leq d^{1/2} \). The error resulting from Theorem 9 is \( \lambda + \mu \), where

\[
\lambda = C_1 \cdot \sqrt{n} \cdot d^{1/4} (c(\varepsilon / T, \delta / T)^{1/2} + \chi),
\]

\[
\mu = C_2 \cdot \sqrt{\kappa} (\log d)^{3/2} \sqrt{(\log \kappa + \log T + \log 1/\delta)} / \varepsilon.
\]

Note that \( T \leq \kappa \) and \( \log \kappa = O(\log d) \). Also assuming that \( d \leq \exp(n) \) and \( \delta \geq \exp(-n) \), \( \log 1/\beta \) is \( O(\log n) \). We then get

\[
\mu = C'_2 \sqrt{n} \cdot d^{1/4} \cdot (\log d + \log 1/\delta) \sqrt{\log n} (\log d)^{3/2} / (\varepsilon \chi).
\]

We can now choose

\[
\chi = \left((\log d + \log 1/\delta) \sqrt{\log n} (\log d)^{3/2} / (\varepsilon)\right)^{1/2}.
\]

Observe that \( \chi > c(\varepsilon / T, \delta / T)^{1/2} \). Thus the \( \mu \) term dominates and is equal to

\[
\mu = C'_2 \sqrt{n} \cdot d^{1/4} \cdot \chi.
\]

\[
= O\left(\sqrt{n} \cdot d^{1/4} \cdot (\log d + \log 1/\delta)^{1/2} (\log n)^{1/4} (\log d)^{3/4} / \varepsilon^{1/2}\right).
\]

The claim follows.

\( \square \)

Observe that the error bound of Theorem 10 is, up to logarithmic factors, \( \sqrt{nd}^{1/4} \).

Acknowledgments We would like to thank Moritz Hardt and Anupam Gupta for several helpful discussions.

References

1. Alon, N., Naor, A.: Approximating the cut-norm via Grothendieck’s inequality. In: ACM Symposium on Theory of Computing, pp. 72–80. ACM Press, New York (2004)
2. Barak, B., Chaudhuri, K., Dwork, C., Kale, S., McSherry, F., Talwar, K.: Privacy, accuracy, and consistency too: a holistic solution to contingency table release. In: Libkin, L. (ed.) Proceedings of ACM PODS, pp. 273–282. ACM Press, New York (2007)
3. Blum, A., Ligett, K., Roth, A.: A learning theory approach to non-interactive database privacy. In: STOC ‘08: Proceedings of the 40th Annual ACM Symposium on Theory of Computing, pp. 609–618. ACM Press, New York (2008)
4. Bun, M., Ullman, J., Vadhan, S.: Fingerprinting codes and the price of approximate differential privacy. arXiv preprint http://arxiv.org/abs/1311.3158 (2013)
5. Buss, S.R., Grigoriev, D., Impagliazzo, R., Pitassi, T.: Linear gaps between degrees for the polynomial calculus modulo distinct primes. J. Comput. Syst. Sci. 62(2), 267–289 (2001)

6. Chandrasekaran, K., Thaler, J., Ullman, J., Wan, A.: Faster private release of marginals on small databases. CoRR http://arxiv.org/abs/1304.3754 (2013)

7. Cheraghchi, M., Klivans, A., Kothari, P., Lee, H.K.: Submodular functions are noise stable. In: SODA ’12 Proceedings of the Twenty-Third Annual ACM–SIAM. Symposium on Discrete Algorithms (SODA), pp. 1586–1592 (2012)

8. Clarkson, K.: Coresets, sparse greedy approximation, and the Frank–Wolfe algorithm. ACM Trans. Algorithms (TALG) 6(4), 63 (2010)

9. Dasgupta, S., Gupta, A.: An elementary proof of a theorem of Johnson and Lindenstrauss. Random Struct. Algorithms 22, 60–65 (2003)

10. Dinur, I., Nissim, K.: Revealing information while preserving privacy. In: Proceedings of 22nd ACM Symposium on Principles of Database Systems, pp. 202–210. ACM Press, New York (2003)

11. Dubhashi, D.P., Panconesi, A.: Concentration of Measure for the Analysis of Randomized Algorithms. Cambridge University Press, New York (2009)

12. Dwork, C., Nissim, K.: Privacy-preserving datamining on vertically partitioned databases. In: Advances in Cryptology, CRYPTO’04. Lecture Notes in Computer Science, vol. 3152, pp. 528–544. Springer, Berlin (2004)

13. Dwork, C., Roth, A.: The algorithmic foundations of differential privacy. Theor. Comput. Sci. 9(3–4), 211–407 (2013)

14. Dwork, C., Mosherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Halevi, S., Rabin, T. (eds.) Theory of Cryptography. Lecture Notes in Computer Science, vol. 3876, pp. 265–284. Springer, Berlin Heidelberg (2006)

15. Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., Naor, M.: Our data, ourselves: privacy via distributed noise generation. In: Vaudena, S. (ed.) EUROCRYPT. Lecture Notes in Computer Science, vol. 4004, pp. 486–503. Springer, Heidelberg (2006)

16. Dwork, C., Naor, M., Reingold, O., Rothblum, G.N., Vadhan, S.: On the complexity of differentially private data release: efficient algorithms and hardness results. In: Proceedings of the 41st ACM Symposium on Theory of Computing, pp. 381–390. ACM Press, New York (2009)

17. Dwork, C., Rothblum, G.N., Vadhan, S.: Boosting and differential privacy. In: 2010 51st Annual IEEE Symposium on Foundations of Computer Science (FOCS), pp. 51–60. IEEE, Las Vegas (2010)

18. Frank, M., Wolfe, P.: An algorithm for quadratic programming. Nav. Res. Logist. Q. 3(1–2), 95–110 (1956)

19. Grigoriev, D.: Linear lower bound on degrees of positivstellensatz calculus proofs for the parity. Theor. Comput. Sci. 259(1–2), 613–622 (2001)

20. Grothendieck, A.: Résumé de la théorie métrique des produits tensoriels topologiques. Bol. Soc. Mat. Sao Paulo 8(1–79), 88 (1953)

21. Grötschel, M., Lovász, L., Schrijver, A.: The ellipsoid method and its consequences in combinatorial optimization. Combinatorica 1(2), 169–197 (1981)

22. Gupta, A., Hardt, M., Roth, A., Ullman, J.: Privately releasing conjunctions and the statistical query barrier. In: STOC, pp. 803–812. ACM Press, New York (2011)

23. Hardt, M., Rothblum, G.: A multiplicative weights mechanism for privacy-preserving data analysis. In: Proceedings of the Foundations of Computer Science (FOCS). IEEE, Las Vegas (2010)

24. Hardt, M., Talwar, K.: On the geometry of differential privacy. In: Proceedings of the 42nd ACM Symposium on Theory of Computing, pp. 705–714. ACM Press, New York (2010)

25. Hardt, M., Rothblum, G.N., Servedio, R.A.: Private data release via learning thresholds. In: Proceedings of the Twenty-Third Annual ACM–SIAM Symposium on Discrete Algorithms, SODA’12, pp. 168–187. SIAM, Kyoto (2012). http://dl.acm.org/citation.cfm?id=2095116.2095131

26. Hardt, M., Ligett, K., McSherry, F.: A simple and practical algorithm for differentially private data release. In: NIPS, pp. 2348–2356 (2012)

27. Kasiviswanathan, S., Rudelson, M., Smith, A., Ullman, J.: The price of privately releasing contingency tables and the spectra of random matrices with correlated rows. In: Proceedings of the 42nd ACM symposium on Theory of computing, pp. 775–784. ACM Press, New York (2010)

28. Lindenstrauss, J., Pelszycinski, A.: Absolutely summing operators in $\ell^p$-spaces and their applications. Stud. Math. 29(3), 275–326 (1968)
29. Nikolov, A., Talwar, K., Zhang, L.: The geometry of differential privacy: the sparse and approximate cases. In: Proceedings of the 45th Annual ACM Symposium on Theory of Computing, STOC ’13, pp. 351–360. ACM Press, New York (2013)
30. O’Donnell, R., Zhou, Y.: Approximability and proof complexity. In: SODA, pp. 1537–1556. (2013)
31. Roth, A., Roughgarden, T.: Interactive privacy via the median mechanism. In: Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC ’10, pp. 765–774. ACM Press, New York, (2010)
32. Thaler, J., Ullman, J., Vadhan, S.P.: Faster algorithms for privately releasing marginals. ICALP 1, 810–821 (2012)
33. Ullman, J., Vadhan, S.: PCPs and the hardness of generating private synthetic data. In: Proceedings of the 8th Conference on Theory of Cryptography, TCC 2011, Providence, RI (2011)