

Nearly Optimal Private Convolution

Nadia Fawaz* S. Muthukrishnan† Aleksandar Nikolov‡

November 7, 2018

Abstract

We study computing the convolution of a private input x with a public input h , while satisfying the guarantees of (ϵ, δ) -differential privacy. Convolution is a fundamental operation, intimately related to Fourier Transforms. In our setting, the private input may represent a time series of sensitive events or a histogram of a database of confidential personal information. Convolution then captures important primitives including linear filtering, which is an essential tool in time series analysis, and aggregation queries on projections of the data.

We give a nearly optimal algorithm for computing convolutions while satisfying (ϵ, δ) -differential privacy. Surprisingly, we follow the simple strategy of adding independent Laplacian noise to each Fourier coefficient and bounding the privacy loss using the composition theorem from [10]. We derive a closed form expression for the optimal noise to add to each Fourier coefficient using convex programming duality. Our algorithm is very efficient – it is essentially no more computationally expensive than a Fast Fourier Transform. To prove near optimality, we use the recent discrepancy lowerbounds of [23] and derive a spectral lower bound using a characterization of discrepancy in terms of determinants.

1 Introduction

The *noise complexity* of linear queries is of fundamental interest in the theory of differential privacy. Consider a database that represents users (or events) of N different types (in the case of events, a type is a time step). We may encode the database as a vector \mathbf{x} indexed by $\{1, \dots, N\}$, where x_i gives the number of users of type i . A linear query asks for the dot product $\langle \mathbf{a}, \mathbf{x} \rangle$; a *workload* of M queries is given as a matrix \mathbf{A} , and the intended output is \mathbf{Ax} . As the database often encodes personal information, we wish to answer queries in a way that does not compromise the individuals represented in the data. We adopt the now standard notion of (ϵ, δ) -*differential privacy* [8]; informally, an algorithm is differentially private if its output distribution does not change drastically

*Technicolor, Palo Alto CA, nadia.fawaz@technicolor.com

†Rutgers University, Piscataway NJ, muthu@cs.rutgers.edu

‡Rutgers University, Piscataway NJ, anikolov@cs.rutgers.edu

when a single user/event changes in the database. This definition necessitates randomization and approximation, and, therefore, the question of the optimal accuracy of any differentially private algorithm on a workload \mathbf{A} comes into the center. We discuss accuracy in terms of *mean squared error* as a measure of approximation: the expected average of squared error over all M queries.

The queries in a workload \mathbf{A} can have different degrees of correlation, and this poses different challenges for the private approximation algorithm. In one extreme, when \mathbf{A} is a set of $\Omega(N)$ independently sampled random $\{0, 1\}$ (i.e. counting) queries, we know, by the seminal work of Dinur and Nissim [7], that any (ϵ, δ) -differentially private algorithm needs to incur at least $\Omega(N)$ squared error per query on average. On the other hand, if \mathbf{A} consists of the same counting query repeated M times, we only need to add $O(1)$ noise per query [8]. While those two extremes are well understood – the bounds cited above are tight – little is known about workloads of queries with some, but not perfect, correlation.

The *convolution*¹ of the private input \mathbf{x} with a public vector \mathbf{h} is defined as the vector \mathbf{y} where

$$y_i = \sum_{j=1}^N h_j x_{i-j \pmod{N}}.$$

This convolution map is a workload of N linear queries. Each query is a circular shift of the previous one, and, therefore, the queries are far from independent but not identical either. Convolution is a fundamental operation that arises in algebraic computations such as polynomial multiplication. It is a basic operation in signal analysis and has well known connection to Fourier transforms. Of primary interest to us, it is a natural primitive in various applications:

- linear filters in the analysis of time series data can be cast as convolutions; as example applications, linear filtering can be used to isolate cycle components in time series data from spurious variations, and to compute time-decayed statistics of the data;
- when user type in the database is specified by d binary attributes, aggregate queries such as k -wise marginals and generalizations can be represented as convolutions.

Privacy concerns arise naturally in these applications: the time series data can contain records of sensitive events, such as financial transactions, records of user activity, etc.; some of the attributes in a database can be sensitive, for example when dealing with databases of medical data.

We give the first nearly optimal algorithm for computing convolution under (ϵ, δ) -differential privacy constraints. Our algorithm gives the lowest mean squared error achievable by adding independent (but non-uniform) Laplace noise to the Fourier coefficients of \mathbf{x} and bounding the privacy loss by the composition

¹Here we define circular convolution, but, however, as discussed in the paper, our results generalize to other types of convolution, which are defined similarly.

theorem of Dwork et al. [10]. Using complementary slackness conditions, we derive a simple closed form for the optimal amount of error that should be added in the direction of each Fourier coefficient. We prove that, for any fixed \mathbf{h} , up to polylogarithmic factors, *any* (ε, δ) -differential private algorithm incurs at least as much squared error per query as our algorithm. *Somewhat surprisingly, our result shows that the simple strategy of adding independent noise in the Fourier domain is nearly optimal for computing convolutions.* Prior to our work there were known nearly instance-optimal² (ε, δ) -differentially private algorithm for a natural class of linear queries. Additionally, our algorithm is simpler and more efficient than related algorithms for $(\varepsilon, 0)$ -differential privacy.

To prove optimality of our algorithm, we use the recent discrepancy-based noise lower bounds of Muthukrishnan and Nikolov [23]. We use a characterization of discrepancy in terms of determinants of submatrices discovered by Lovász, Spencer, and Vesztegombi, together with ideas by Hardt and Talwar, who give instance-optimal algorithms for the stronger notion of $(\varepsilon, 0)$ -differential privacy³. A main technical ingredient in our proof is a connection between the discrepancy of a matrix \mathbf{A} and the discrepancy of \mathbf{PA} where \mathbf{P} is an orthogonal projection operator.

In addition to applications to linear filtering, our algorithm allows us to approximate marginal queries encoded by w -DNFs, which generalize k -wise marginal queries. Using concentration results for the spectrum of bounded-width DNFs, we derive a non-trivial error bound for approximating w -DNF queries. The bound is independent of the DNF size.

Related work. The problem of computing private convolutions has not been considered in the literature before. However, there is a fair amount of work on the more general problem of computing arbitrary linear queries, as well as some work on special cases of convolution maps.

The problem of computing arbitrary linear maps of a private database histogram was first considered in the seminal work of Dinur and Nissim [7]. They showed that privately answering M random 0-1 queries on a universe of size N requires $\Omega(N)$ mean squared error as long as $M = \Omega(N)$, and this bound is tight. These bounds do not directly apply to our work, as a set of independent random queries is not likely to encode a circular convolution. Nevertheless, one can show, using spectral noise lower bounds, that a convolution with a random 0-1 vector h requires asymptotically as much error as N random queries. Yet, many particular convolutions of interest require much less noise. This fact motivates us to study algorithms for approximating the convolution $x * h$ which are optimal for any given h . An efficient algorithm with this kind of instance per instance (in terms of h) optimality guarantee obviates the need to develop specialized algorithms. Next we review some prior work on special instances of convolution maps and also related work on computing linear maps optimally.

Bolot et al. [3] give algorithms for various decayed sum queries: window

²Note that instance-optimality here refers to the query vector \mathbf{h} , while we still consider worst-case error over the private input \mathbf{x} .

³Note that establishing instance-optimality for (ε, δ) -differential privacy is harder from error lower bounds perspective, as the privacy definition is weaker.

sums, exponentially and polynomially decayed sums. Any decayed sum function is a type of linear filter, and, therefore, a special case of convolution. Thus, our current work gives a nearly optimal (ϵ, δ) -differentially private approximation for *any decayed sum function*. Moreover, as far as mean squared error is concerned, our algorithms give improved error bounds for the window sums problem: constant squared error per query. However, unlike [3], we only consider the offline batch-processing setting, as opposed to the online continual observation setting.

The work of Barak et al. [1] on computing k -wise marginals concerns a restricted class of convolutions (see Section 5). Moreover, Kasiviswanathan [16] show a noise lower bound for k -wise marginals which is tight in the worst case. Our work is a generalization: we are able to give nearly optimal approximations to a wider class of queries, and our lower and upper bounds nearly match for any convolution.

Li and Miklau [18, 19] proposed the class of extended matrix mechanisms, building on prior work on the matrix mechanism [17], and showed how to efficiently compute the optimal mechanism from the class. Furthermore, independently and concurrently with our work, Cormode et al. [6] considered adding optimal non-uniform noise to a fixed transform of the private database. Since our mechanism is a special instance of the extended matrix mechanism, the algorithms of Li and Miklau have at most as much error as our algorithm. However, similarly to [6], we gain significantly in efficiency by fixing a specific transform (in our case the Fourier transform) of the data and computing a closed form expression for the optimal noise magnitudes. Unlike the work of Li and Miklau and Cormode et al., we are able to show nearly tight *lower bounds* for *any* differentially private algorithm (not just the extended matrix mechanism) and any set of convolution queries. Therefore, we can show that the choice of the Fourier transform comes without loss of generality for any set of convolution queries.

In the setting of $(\epsilon, 0)$ -differential privacy, Hardt and Talwar [15] prove nearly optimal upper and lower bounds on approximating \mathbf{Ax} for any matrix \mathbf{A} . Recently, their results were improved, and made unconditional by Bhaskara et al. [2]. Prior to our work a similar result was not known for the weaker notion of approximate privacy, i.e. (ϵ, δ) -differential privacy. Subsequently to our work, our results were generalized by Nikolov, Talwar, and Zhang [24] to give nearly optimal algorithms for computing any linear map A under (ϵ, δ) -differential privacy. Their work combined our use of hereditary discrepancy bounds on error through the determinant lower bound with results from asymptotic convex geometry. The algorithms from [2, 15] are computationally expensive, as they need to sample from a high-dimensional convex body⁴. Even the more efficient algorithm from [24] has running time $\Omega(N^3)$, as it needs to approximate the minimum enclosing ellipsoid of an N -dimensional convex body. By contrast our algorithm’s running time is dominated by the running time of the Fast Fourier

⁴One of the best known algorithms is due to Lovász and Vempala [21] and, ignoring other parameters, makes $\Theta(N^3)$ calls to a separation oracle, each of which would require solving a linear programming feasibility problem.

Transform, i.e. $O(N \log N)$, making it more suitable for practical applications. Also, for some sets of queries, such as running sums, our analysis gives tighter bounds than the analysis of the algorithm in [24].

A related line of work seeks to exploit sparsity assumptions on the private database in order to reduce error; as we do not limit the database size, our results are not directly comparable. Using our histogram representation, database size corresponds to the norm $\|\mathbf{x}\|_1$ where \mathbf{x} is the database in histogram representation. For general linear queries, the multiplicative weights algorithm of Hardt and Rothblum achieves mean squared error $O(n\sqrt{\log N})$ for $\|x\|_1 \leq n$. This bound is nearly tight for random queries, but can be loose for special queries of interest. For example, running sums require noise $O(\log^{O(1)} N)$, which is less than n except for n very small in the universe size. In general, algorithms which bound database size in order to bound error become less useful when database size is large compared to the total number of queries, and for very large databases algorithms such as ours are still of interest. This is true also for the line of algorithms for marginal queries which give error an arbitrary small constant fraction of the database size [5, 13, 14, 25]. Note further that the optimal error for a *subset of all marginal queries* may be less than linear in database size, and our algorithms will give near optimal error for the specific subset of interest.

Organization. We begin with preliminaries on differential privacy and convolution operators. In section 3 we derive our main lower bound result, and in section 4 we describe and analyze our nearly optimal algorithm. In section 5 we describe applications of our main results.

2 Preliminaries

Notation. \mathbb{N} , \mathbb{R} , and \mathbb{C} are the sets of non-negative integers, real, and complex numbers respectively. By \log we denote the logarithm in base 2 while by \ln we denote the logarithm in base e . Matrices and vectors are represented by boldface upper and lower cases, respectively. \mathbf{A}^T , \mathbf{A}^* , \mathbf{A}^H stand for the transpose, the conjugate and the transpose conjugate of \mathbf{A} , respectively. The trace and the determinant of \mathbf{A} are respectively denoted by $\text{tr}(\mathbf{A})$ and $\det(\mathbf{A})$. \mathbf{A}_m denotes the m -th row of matrix \mathbf{A} , and $\mathbf{A}_{:n}$ its n -th column. $\mathbf{A}_{|S}$, where \mathbf{A} is a matrix with N columns and $S \subseteq [N]$, denotes the submatrix of \mathbf{A} consisting of those columns corresponding to elements of S . $\lambda_{\mathbf{A}}(1), \dots, \lambda_{\mathbf{A}}(n)$ represent the eigenvalues of an $n \times n$ matrix \mathbf{A} . \mathbf{I}_N is the identity matrix of size N . $\mathbb{E}[\cdot]$ is the statistical expectation operator. $Lap(x, s)$ denotes the Laplace distribution centered at x with scale s , i.e. the distribution of the random variable $x + \eta$ where η has probability density function $p(y) \propto \exp(-|y|/s)$.

2.1 Convolution

In this section, we first give the definition of circular convolution. We then recall important results on the Fourier eigen-decomposition of convolution. Gen-

eralization to other notions of convolution and applications are discussed in Section 5.

Let $x = \{x_0, \dots, x_{N-1}\}$ be a real input sequence of length N , and $h = \{h_0, \dots, h_{N-1}\}$ a sequence of length N . The circular convolution of x and h is the sequence $y = x * h$ of length N defined by

$$y_k = \sum_{n=0}^{N-1} x_n h_{(k-n) \bmod N}, \forall k \in \{0, \dots, N-1\}. \quad (1)$$

Definition 1. The $N \times N$ circular convolution matrix \mathbf{H} is defined as

$$\mathbf{H} = \begin{bmatrix} h_0 & h_{N-1} & h_{N-2} & \dots & h_1 \\ h_1 & h_0 & \ddots & \ddots & \vdots \\ h_2 & \ddots & \ddots & \ddots & h_{N-2} \\ \vdots & \ddots & \ddots & h_0 & h_{N-1} \\ h_{N-1} & \dots & h_2 & h_1 & h_0 \end{bmatrix}_{N \times N}.$$

This matrix is a circulant matrix with first column $\mathbf{h} = [h_0, \dots, h_{N-1}]^T \in \mathbb{R}^N$, and its subsequent columns are successive cyclic shifts of its first column. Note that \mathbf{H} is a normal matrix ($\mathbf{H}\mathbf{H}^H = \mathbf{H}^H\mathbf{H}$).

Define the column vectors $\mathbf{x} = [x_0, \dots, x_{N-1}]^T \in \mathbb{R}^N$, and $\mathbf{y} = [y_0, \dots, y_{N-1}]^T \in \mathbb{R}^N$. The circular convolution (1) can be written in matrix notation $\mathbf{y} = \mathbf{H}\mathbf{x}$. In Section 2.2, we recall that circular convolution can be diagonalized in the Fourier basis.

2.2 Fourier Eigen-decomposition of Convolution

In this section, we recall the definition of the Fourier basis, and the eigen-decomposition of circular convolution in this basis.

Definition 2. The normalized Discrete Fourier Transform (DFT) matrix of size N is defined as

$$\mathbf{F}_N = \left\{ \frac{1}{\sqrt{N}} \exp \left(-\frac{j2\pi m n}{N} \right) \right\}_{m,n \in \{0, \dots, N-1\}}. \quad (2)$$

Note that \mathbf{F}_N is symmetric ($\mathbf{F}_N = \mathbf{F}_N^T$) and unitary ($\mathbf{F}_N \mathbf{F}_N^H = \mathbf{F}_N^H \mathbf{F}_N = \mathbf{I}_N$).

We denote by $\mathbf{f}_m = [1, e^{\frac{j2\pi m}{N}}, \dots, e^{\frac{j2\pi m (N-1)}{N}}]^T \in \mathbb{C}^N$ the m -th column of the inverse DFT matrix \mathbf{F}_N^H . Or alternatively, \mathbf{f}_m^H is the m -th row of \mathbf{F}_N . The normalized DFT of a vector \mathbf{h} is simply given by $\hat{\mathbf{h}} = \mathbf{F}_N \mathbf{h}$.

Theorem 1 ([12]). Any circulant matrix \mathbf{H} can be diagonalized in the Fourier basis \mathbf{F}_N : the eigenvectors of \mathbf{H} are given by the columns $\{\mathbf{f}_m\}_{m \in \{0, \dots, N-1\}}$ of

the inverse DFT matrix \mathbf{F}_N^H , and the associated eigenvalues $\{\lambda_m\}_{m \in \{0, \dots, N-1\}}$ are given by $\sqrt{N}\hat{\mathbf{h}}$, i.e. by the DFT of the first column \mathbf{h} of \mathbf{H} :

$$\forall m \in \{0, \dots, N-1\}, \quad \mathbf{H}\mathbf{f}_m = \lambda_m \mathbf{f}_m$$

$$\text{where} \quad \lambda_m = \sqrt{N}\hat{h}_m = \sum_{n=0}^{N-1} h_n e^{-\frac{j2\pi m n}{N}}.$$

Equivalently, in the Fourier domain, the circular convolution matrix \mathbf{H} becomes a diagonal matrix $\hat{\mathbf{H}} = \text{diag}\{\sqrt{N}\hat{\mathbf{h}}\}$.

Corollary 1. Consider the circular convolution $\mathbf{y} = \mathbf{H}\mathbf{x}$ of \mathbf{x} and \mathbf{y} . Let $\hat{\mathbf{x}} = \mathbf{F}_N \mathbf{x}$ and $\hat{\mathbf{h}} = \mathbf{F}_N \mathbf{h}$ denote the normalized DFT of \mathbf{x} and \mathbf{h} . In the Fourier domain, the circular convolution becomes a simple entry-wise multiplication of the components of $\sqrt{N}\hat{\mathbf{h}}$ with the components of $\hat{\mathbf{x}}$: $\hat{\mathbf{y}} = \mathbf{F}_N \mathbf{y} = \hat{\mathbf{H}} \hat{\mathbf{x}}$.

2.3 Privacy Model

2.3.1 Differential Privacy

Two real-valued input vectors $\mathbf{x}, \mathbf{x}' \in [0, 1]^N$ are *neighbors* when $\|\mathbf{x} - \mathbf{x}'\|_1 \leq 1$.

Definition 3. A randomized algorithm \mathcal{A} satisfies (ϵ, δ) -differential privacy if for all neighbors $\mathbf{x}, \mathbf{x}' \in [0, 1]^N$, and all measurable subsets T of the support of \mathcal{A} , we have

$$\Pr[\mathcal{A}(\mathbf{x}) \in T] \leq e^\epsilon \Pr[\mathcal{A}(\mathbf{x}') \in T] + \delta,$$

where probabilities are taken over the randomness of \mathcal{A} .

2.3.2 Laplace Noise Mechanism

Definition 4. A function $f : [0, 1]^N \rightarrow \mathbb{C}$ has sensitivity s if s is the smallest number such that for any two neighbors $\mathbf{x}, \mathbf{x}' \in [0, 1]^N$,

$$|f(\mathbf{x}) - f(\mathbf{x}')| \leq s.$$

Theorem 2 ([8]). Let $f : [0, 1]^N \rightarrow \mathbb{C}$ have sensitivity s . Suppose that on input \mathbf{x} , algorithm \mathcal{A} outputs $f(\mathbf{x}) + z$, where $z \sim \text{Lap}(0, s/\epsilon)$. Then \mathcal{A} satisfies $(\epsilon, 0)$ -differential privacy.

2.3.3 Composition Theorems

An important feature of differential privacy is its robustness: when an algorithm is a “composition” of several differentially private algorithms, the algorithm itself also satisfies differential privacy constraints, with the privacy parameters degrading smoothly. The results in this subsection quantify how the privacy parameters degrade.

The first composition theorem is an easy consequence of the definition of differential privacy:

Theorem 3 ([8]). *Let \mathcal{A}_1 satisfy $(\varepsilon_1, \delta_1)$ -differential privacy and \mathcal{A}_2 satisfy $(\varepsilon_2, \delta_2)$ -differential privacy, where \mathcal{A}_2 could take the output of \mathcal{A}_1 as input. Then the algorithm which on input \mathbf{x} outputs the tuple $(\mathcal{A}_1(\mathbf{x}), \mathcal{A}_2(\mathcal{A}_1(\mathbf{x}), \mathbf{x}))$ satisfies $(\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)$ -differential privacy.*

In a more recent paper, Dwork et al. proved a more sophisticated composition theorem, which often gives asymptotically better bounds on the privacy parameters. Next we state their theorem.

Theorem 4 ([10]). *Let $\mathcal{A}_1, \dots, \mathcal{A}_k$ be such that algorithm \mathcal{A}_i satisfies $(\varepsilon_i, 0)$ -differential privacy. Then the algorithm that on input \mathbf{x} outputs the tuple $(\mathcal{A}_1(\mathbf{x}), \dots, \mathcal{A}_k(\mathbf{x}))$ satisfies (ε, δ) -differential privacy for any $\delta > 0$ and*

$$\varepsilon \geq \sqrt{2 \ln \left(\frac{1}{\delta} \right) \sum_{i=1}^m \varepsilon_i^2}.$$

2.4 Accuracy

In this paper we are interested in differentially private algorithms for the *convolution problem*. In the convolution problem, we are given a *public* sequence $h = \{h_1, \dots, h_N\}$ and a *private* sequence $x = \{x_1, \dots, x_N\}$. Our goal is to design an algorithm \mathcal{A} that is (ε, δ) -differentially private with respect to the private input x (taken as column vector \mathbf{x}), and approximates the convolution $h * x$. More precisely,

Definition 5. *Given a vector $\mathbf{h} \in \mathbb{R}^N$ which defines a convolution matrix \mathbf{H} , the mean (expected) squared error (MSE) of an algorithm \mathcal{A} is defined as*

$$\text{MSE} = \sup_{\mathbf{x} \in \mathbb{R}^N} \frac{1}{N} \mathbb{E}[\|\mathcal{A}(\mathbf{x}) - \mathbf{H}\mathbf{x}\|_2^2].$$

Note that MSE measures the mean expected squared error *per output component*.

3 Lower Bounds

In this section we derive a spectral lower bound on mean squared error of differentially private approximation algorithms for circular convolution. We prove that this bound is nearly tight for every fixed \mathbf{h} in the following section. The lower bound is state as Theorem 5.

Theorem 5. *Let $\mathbf{h} \in \mathbb{R}^N$ be an arbitrary real vector and let us relabel the Fourier coefficients of \mathbf{h} so that $|\hat{h}_0| \geq \dots \geq |\hat{h}_{N-1}|$. For all sufficiently small ε and δ , the expected mean squared error MSE of any (ε, δ) -differentially private algorithm \mathcal{A} that approximates $\mathbf{h} * \mathbf{x}$ is at least*

$$\text{MSE} = \Omega \left(\max_{K=1}^N \frac{K^2 \hat{h}_{K-1}^2}{N \log^2 N} \right). \quad (3)$$

For the remainder of the paper, we define the notation $\text{specLB}(\mathbf{h})$ for the right hand side of (3), i.e. $\text{specLB}(\mathbf{h}) = \max_{K=1}^N \frac{K^2 \hat{h}_{K-1}^2}{N \log^2 N}$.

The proof of Theorem 5 is based on recent work [23] connecting combinatorial discrepancy and privacy. Adapting a strategy due to Hardt and Talwar [15], we instantiate the basic discrepancy lower bound for any matrix \mathbf{PA} , where \mathbf{P} is a projection matrix, and use the maximum of these lower bounds. However, we need to resolve several issues that arise in the setting of (ε, δ) -differential privacy. While projection works naturally with the volume-based lower bounds of Hardt and Talwar, the connection between the discrepancy of \mathbf{A} and \mathbf{PA} is not immediate, since discrepancy is a combinatorially defined quantity. Our main technical contribution in this section is analyzing the discrepancy of \mathbf{PA} via the determinant lower bound of Lovász, Spencer, Vesztergombi. This approach was generalized and extended by Nikolov, Talwar, and Zhang [24] to show nearly optimal lower bounds for arbitrary linear maps.

We start our presentation with preliminaries from prior work and then we develop our lower bounds for convolutions.

3.1 Discrepancy Preliminaries

We define (ℓ_2) hereditary discrepancy as

$$\text{herdisc}(\mathbf{A}) = \max_{W \subseteq [N]} \min_{\mathbf{v} \in \{-1, +1\}^W} \|\mathbf{A}\mathbf{v}\|_2.$$

The following result connects discrepancy and differential privacy:

Theorem 6 ([23]). *Let \mathbf{A} be an $M \times N$ complex matrix and let \mathcal{A} be an (ε, δ) -differentially private algorithm for sufficiently small constant ε and δ . There exists a constant C and a vector $\mathbf{x} \in \{0, 1\}^N$ such that $\mathbb{E}[\|\mathcal{A}(\mathbf{x}) - \mathbf{A}\mathbf{x}\|_2^2] \geq C \frac{\text{herdisc}(\mathbf{A})^2}{\log^2 N}$.*

The determinant lower bound for hereditary discrepancy due to Lovász, Spencer, and Vesztergombi gives us a spectral lower bound on the noise required for privacy.

Theorem 7 ([20]). *There exists a constant C' such that for any complex $M \times N$ matrix \mathbf{A} , $\text{herdisc}(\mathbf{A}) \geq C' \max_{K, \mathbf{B}} \sqrt{K} |\det(\mathbf{B})|^{1/K}$, where K ranges over $[\min\{M, N\}]$ and \mathbf{B} ranges over $K \times K$ submatrices of \mathbf{A} .*

Corollary 8. *Let \mathbf{A} be an $M \times N$ complex matrix and let \mathcal{A} be an (ε, δ) -differentially private algorithm for sufficiently small constant ε and δ . There exists a constant C and a vector $\mathbf{x} \in \{0, 1\}^N$ such that, for any $K \times K$ submatrix \mathbf{B} of \mathbf{A} , $\mathbb{E}[\|\mathcal{A}(\mathbf{x}) - \mathbf{A}\mathbf{x}\|_2^2] \geq C \frac{K |\det(\mathbf{B})|^{2/K}}{\log^2 N}$.*

3.2 Proof of Theorem 5

We exploit the power of the determinant lower bound of Corollary 8 by combining the simple but very useful observation that projections do not increase

mean squared error with a lower bound on the maximum determinant of a submatrices of a rectangular matrix. We present these two ingredients in sequence and finish the section with a proof of Theorem 5.

Lemma 1. *Let \mathbf{A} be an $M \times N$ complex matrix and let \mathcal{A} be an (ε, δ) -differentially private algorithm for sufficiently small constant ε and δ . There exists a constant C and a vector $\mathbf{x} \in \{0, 1\}^N$ such that for any $L \times M$ projection matrix \mathbf{P} and for any $K \times K$ submatrix \mathbf{B} of \mathbf{PA} , $\mathbb{E}[\|\mathcal{A}(\mathbf{x}) - \mathbf{Ax}\|_2^2] \geq C \frac{K |\det(\mathbf{B})|^{2/K}}{\log^2 N}$.*

Proof. We show that there exists an (ε, δ) -differentially private algorithm \mathcal{B} that satisfies

$$\mathbb{E}[\|\mathcal{B}(\mathbf{x}) - \mathbf{PAx}\|_2^2] \leq \mathbb{E}[\|\mathcal{A}(\mathbf{x}) - \mathbf{Ax}\|_2^2]. \quad (4)$$

Then we can apply Corollary 8 to \mathcal{B} and \mathbf{PA} to prove the corollary.

The algorithm \mathcal{B} on input \mathbf{x} outputs \mathbf{Py} where $\mathbf{y} = \mathcal{A}(\mathbf{x})$. Since \mathcal{B} is a function of $\mathcal{A}(\mathbf{x})$ only, it satisfies (ε, δ) -differential privacy by Theorem 3. It satisfies (4) since for any \mathbf{y} and any projection matrix \mathbf{P} it holds that $\|\mathbf{P}(\mathbf{y} - \mathbf{Ax})\|_2 \leq \|\mathbf{y} - \mathbf{Ax}\|_2$. \square

Our main technical tool is a linear algebraic fact connecting the determinant lower bound for \mathbf{A} and the determinant lower bound for any projection of \mathbf{A} .

Lemma 2. *Let \mathbf{A} be an $M \times N$ complex matrix with singular values $\lambda_1 \geq \dots \geq \lambda_N$ and let \mathbf{P} be a projection matrix onto the span of the left singular vectors corresponding to $\lambda_1, \dots, \lambda_K$. There exists a constant C and $K \times K$ submatrix \mathbf{B} of \mathbf{PA} such that*

$$|\det(\mathbf{B})|^{1/K} \geq C \sqrt{\frac{K}{N}} \left(\prod_{i=1}^K \lambda_i \right)^{1/K}$$

Proof. Let $\mathbf{C} = \mathbf{PA}$ and consider the matrix $\mathbf{D} = \mathbf{CC}^H$. It has eigenvalues $\lambda_1^2, \dots, \lambda_K^2$, and therefore

$$\det(\mathbf{D}) = \prod_{i=1}^K \lambda_i^2.$$

On the other hand, by the Binet-Cauchy formula for the determinant, we have

$$\begin{aligned} \det(\mathbf{D}) &= \det(\mathbf{CC}^H) \\ &= \sum_{S \in \binom{[N]}{K}} \det(\mathbf{C}|_S)^2 \\ &\leq \binom{N}{K} \max_{S \in \binom{[N]}{K}} \det(\mathbf{C}|_S)^2. \end{aligned}$$

Rearranging and raising to the power $1/2K$, we get that there exists a $K \times K$ submatrix of \mathbf{C} such that

$$|\det(\mathbf{B})|^{1/K} \geq \binom{N}{K}^{-1/2K} \left(\prod_{i=1}^K \lambda_i \right)^{1/K}.$$

Using the bound $\binom{N}{K} \leq \left(\frac{Ne}{K}\right)^K$ completes the proof. \square

We can now prove our main lower bound theorem by combining Lemma 1 and Lemma 2.

of Theorem 5. As usual, we will express $\mathbf{h} * \mathbf{x}$ as the linear map $\mathbf{H}\mathbf{x}$, where \mathbf{H} is the convolution matrix for \mathbf{h} . By Lemma 1, it suffices to show that for each K , there exists a projection matrix \mathbf{P} and a $K \times K$ submatrix \mathbf{B} of \mathbf{PH} such that $|\det(\mathbf{B})|^{1/K} \geq \Omega(\sqrt{K}|\hat{h}_K|)$. Recall that the eigenvalues of \mathbf{H} are $\sqrt{N}\hat{h}_0, \dots, \sqrt{N}\hat{h}_{N-1}$, and, therefore, the i -th singular value of \mathbf{H} is $\sqrt{N}|\hat{h}_{i-1}|$. By Lemma 2, there exists a constant C , a projection matrix P , and a submatrix \mathbf{B} of \mathbf{PH} such that

$$|\det(\mathbf{B})|^{1/K} \geq C \sqrt{\frac{K}{N}} \left(\prod_{i=0}^{K-1} \sqrt{N}|\hat{h}_i| \right)^{1/K} \geq C \sqrt{K}|\hat{h}_K|.$$

This completes the proof. \square

4 Upperbounds

Standard (ε, δ) -privacy techniques such as input perturbation or output perturbation in the time or in the frequency domain lead to mean squared error, at best, proportional to $\|\mathbf{h}\|_2^2$.

Next we describe an algorithm which is nearly optimal for (ε, δ) -differential privacy. This algorithm is derived by formulating the error of a natural class of private algorithms as a convex program and finding a closed form solution. An alternative solution that partitions the spectrum of \mathbf{H} geometrically is described in Appendix A. The class of algorithms we consider is those which add independent Gaussian noise to the Fourier coefficients of the private input \mathbf{x} . Interestingly, we show that this simple strategy is nearly optimal for computing convolution maps.

Consider the class of algorithms, which first add independent Laplacian noise variables $z_i = \text{Lap}(0, b_i)$ to the Fourier coefficients \hat{x}_i to compute $\tilde{x}_i = \hat{x}_i + z_i$, and then output $\tilde{\mathbf{y}} = \mathbf{F}_N^H \hat{\mathbf{H}} \tilde{\mathbf{x}}$. This class of algorithms is parameterized by the vector $\mathbf{b} = (b_0, \dots, b_{N-1})$; a member of the class will be denoted $\mathcal{A}(\mathbf{b})$ in the sequel. The question we address is: For given $\varepsilon, \delta > 0$, how should the noise parameters \mathbf{b} be chosen such that the algorithm $\mathcal{A}(\mathbf{b})$ achieves (ε, δ) -differential privacy in \mathbf{x} for ℓ_1 neighbors, while minimizing the mean squared error MSE? It turns out that by convex programming duality we can derive a closed form expression for the optimal \mathbf{b} , and moreover, the optimal $\mathcal{A}(\mathbf{b})$ is nearly optimal *among all (ε, δ) -differentially private algorithms*. The optimal parameters are used in Algorithm 1.

Theorem 9. *Algorithm 1 satisfies (ε, δ) -differential privacy, and achieves expected mean squared error*

$$\text{MSE} = 4 \frac{\ln(1/\delta)}{\varepsilon^2 N} \|\hat{\mathbf{h}}\|_1^2. \quad (5)$$

Algorithm 1 FOURIER MECHANISM

```

Set  $\gamma = \frac{2 \ln(1/\delta) \|\hat{\mathbf{h}}\|_1}{\varepsilon^2 N}$ 
Compute  $\hat{\mathbf{x}} = \mathbf{F}_N \mathbf{x}$  and  $\hat{\mathbf{h}} = \mathbf{F}_N \mathbf{x}$ .
for all  $i \in \{0, \dots, N-1\}$  do
  if  $|\hat{h}_i| > 0$  then
    Set  $z_i = \text{Lap}\left(\sqrt{\frac{\gamma}{|\hat{h}_i|}}\right)$ 
  else if  $|\hat{h}_i| = 0$  then
    Set  $z_i = 0$ 
  end if
  Set  $\tilde{x}_i = \hat{x}_i + z_i$ .
  Set  $\tilde{y}_i = \sqrt{N} \hat{h}_i \tilde{x}_i$ .
end for
Output  $\tilde{\mathbf{y}} = \mathbf{F}_N^H \tilde{\mathbf{y}}$ 

```

Moreover, Algorithm 1 runs in time $O(N \log N)$.

Before proving Theorem 9, we show that it implies that Algorithm 1 is almost optimal for any given \mathbf{h} .

Theorem 10. For any \mathbf{h} , Algorithm 1 satisfies (ε, δ) -differential privacy and achieves expected mean squared error $O\left(\text{specLB}(\mathbf{h}) \frac{\log^2 N \log^2 |I| \ln(1/\delta)}{\varepsilon^2}\right)$.

Proof. Assume that $|\hat{h}_0| > |\hat{h}_1| > \dots > |\hat{h}_{N-1}|$. Then, by definition of $I = \{0 \leq i \leq N-1 : |\hat{h}_i| > 0\}$, we have $|\hat{h}_j| = 0$, for all $j > |I| - 1$. Thus,

$$\begin{aligned}
\|\hat{\mathbf{h}}\|_1 &= \sum_{i=0}^{|I|-1} |\hat{h}_i| = \sum_{i=1}^{|I|} \frac{1}{i} |\hat{h}_{i-1}| \\
&\leq \left(\sum_{i=1}^{|I|} \frac{1}{i} \right) \sqrt{N} \log N \sqrt{\text{specLB}(\mathbf{h})} \\
&= H_{|I|} \sqrt{N} \log N \sqrt{\text{specLB}(\mathbf{h})}, \tag{6}
\end{aligned}$$

where $H_m = \sum_{i=1}^m \frac{1}{i}$ denotes the m -th harmonic number. Recalling that $H_m = O(\log m)$, and combining the bound (6) with the expression of the MSE (10) yields the desired bound. \square

of Theorem 9. For running time, we note that our algorithm is no more expensive than computing a Fast Fourier Transform, which can be done in $O(N \log N)$ arithmetic operations using the classical Cooley-Tukey algorithm, for example.

Denote the set $I = \{0 \leq i \leq N-1 : |\hat{h}_i| > 0\}$. We formulate the problem of finding the algorithm $\mathcal{A}(\mathbf{b})$ which minimizes MSE subject to privacy constraints

as the following optimization problem:

$$\min_{\{b_i\}_{i \in I}} \sum_{i \in I} b_i^2 |\hat{h}_i|^2 \quad (7)$$

$$\text{s.t. } \sum_{i \in I} \frac{1}{Nb_i^2} = \frac{\varepsilon^2}{2 \ln(1/\delta)} \quad (8)$$

$$b_i > 0, \forall i \in I. \quad (9)$$

Next we justify this formulation.

Privacy Constraint. We first show that the output $\tilde{\mathbf{y}}$ of an algorithm $\mathcal{A}(\mathbf{b})$ is an (ε, δ) -differentially private function of \mathbf{x} , if the constraint (8) is satisfied. Denote $\bar{\mathbf{y}} = \hat{\mathbf{H}}\hat{\mathbf{x}}$. If $\bar{\mathbf{y}}$ is an (ε, δ) -differentially private function of \mathbf{x} , then by Theorem 3, $\tilde{\mathbf{y}}$ is also (ε, δ) -differentially private, since the computation of $\tilde{\mathbf{y}}$ depends only on \mathbf{F}_N^H and $\bar{\mathbf{y}}$ and not on \mathbf{x} directly. Thus we can focus on the requirements on \mathbf{b} for which $\bar{\mathbf{y}}$ is (ε, δ) private.

If $i \notin I$, then $\bar{y}_i = 0$ and does not affect privacy regardless of b_i . Thus, we can set $b_i = 0$ for all $i \notin I$. If $i \in I$, we first characterize the ℓ_1 -sensitivity of \hat{x}_i as a function of \mathbf{x} . Recall that $\hat{x}_i = \mathbf{f}_i^H \mathbf{x}$ is the inner product of \mathbf{x} with the Fourier basis vector \mathbf{f}_i . The sensitivity of \hat{x}_i is therefore $\|f_i\|_\infty = \frac{1}{\sqrt{N}}$, $\forall i$. Then, by Theorem 2, $\tilde{x}_i = \hat{x}_i + \text{Lap}(0, b_i)$ is ε_i -differentially private in \mathbf{x} , with $\varepsilon_i = \frac{1}{\sqrt{Nb_i}}$. The computation of \bar{y}_i depends only on \hat{h}_i and \tilde{x}_i , thus, by Theorem 3, \bar{y}_i is $\frac{1}{\sqrt{Nb_i}}$ -differentially private in \mathbf{x} .

Finally, by Theorem 4, $\bar{\mathbf{y}}$ is (ε, δ) differentially private for any $\delta > 0$, as long as constraint (8) holds.

Accuracy Objective. We show that finding the algorithm $\mathcal{A}(\mathbf{b})$ which minimizes the MSE is equivalent to finding the parameters $b_i \geq 0$, $i \in I$, which minimize the objective function (7). Note that $\tilde{\mathbf{y}} = \mathbf{F}_N^H \hat{\mathbf{H}}\hat{\mathbf{x}} = \mathbf{F}_N^H \hat{\mathbf{H}}(\mathbf{F}_N \mathbf{x} + \mathbf{z}) = \mathbf{y} + \mathbf{F}_N^H \hat{\mathbf{H}}\mathbf{z}$. Thus, the output $\tilde{\mathbf{y}}$ is unbiased: $\mathbb{E}[\tilde{\mathbf{y}}] = \mathbf{y}$. The mean squared error is given by:

$$\begin{aligned} \text{MSE} &= \frac{1}{N} \mathbb{E}[\|\mathbf{F}_N^H \hat{\mathbf{H}}\mathbf{z}\|_2^2] \\ &= \frac{1}{N} \mathbb{E}[\text{tr}(\mathbf{F}_N^H \hat{\mathbf{H}}\mathbf{z}\mathbf{z}^H \hat{\mathbf{H}}^H \mathbf{F}_N)] \\ &= \frac{1}{N} \text{tr}(\hat{\mathbf{H}}^2 \mathbb{E}[\mathbf{z}\mathbf{z}^H]) = 2 \sum_{i \in I} |\hat{h}_i|^2 b_i^2, \end{aligned}$$

which yields the desired objective function (7).

Closed Form Solution. The program (7)–(9) is convex in $1/b_i^2$. Using the KKT conditions of this program, we can derive a closed form optimal solution: $b_i^* = \sqrt{(2 \ln(1/\delta) \|\hat{\mathbf{h}}\|_1) / (N \varepsilon^2 |\hat{h}_i|)}$ when $i \in I$ and $b_i^* = 0$ otherwise. Substituting these values back into the objective finishes the proof. Full details of the analysis of the convex program can be found in Appendix B. \square

5 Generalizations and Applications

In this section we describe some generalizations and applications of our lower bounds and algorithms for private convolution.

5.1 Compressible Convolutions

A case of special interest is convolutions $h * x$ where h is a compressible sequence. Such cases appear in practice in signal processing. For compressible h we can show that Algorithm 1 outperforms input and output perturbation. First we present a definition of compressible sequences and then we give the improved upper bounds. A specific example of private compressible convolutions is developed in Section 5.4 in the context of computing marginal queries.

Definition 6. A vector $\mathbf{h} \in \mathbb{R}^N$ is (c, p) -compressible (in the Fourier basis) if it satisfies:

$$\forall 0 \leq i \leq N-1 : |\hat{h}_i|^2 \leq c \frac{1}{(i+1)^p}.$$

Theorem 11. Let \mathbf{h} be a (c, p) -compressible vector for some constant $p > 2$. Then Algorithm 1 satisfies (ϵ, δ) -differential privacy and achieves expected mean squared error $O\left(\frac{c^2 \log^2 N \log(1/\delta)}{N \epsilon^2}\right)$ for $p = 2$ and for $p \neq 2$ achieves $O\left(\left(\frac{cp}{p-2}\right)^2 \frac{\log(1/\delta)}{N \epsilon^2}\right)$.

Notice that the bound on squared error improves on input and output perturbation by a factor $\tilde{O}(\frac{1}{N})$.

The proof of Theorem 11 follows from Theorem 9 and the following lemma.

Lemma 3. Let \mathbf{h} be a (c, p) -compressible vector for some $p > 1$. Then, we have

$$\|\hat{\mathbf{h}}\|_1 = \sum_{i=0}^{N-1} |\hat{h}_i| \leq \begin{cases} c(1 + \ln N), & \text{if } p = 2 \\ \frac{cp}{p-2}, & \text{if } p > 2 \end{cases}$$

Proof. Approximating a sum by an integral in the usual way, for $0 \leq a \leq b$ and $p \geq 2$, we have

$$\begin{aligned} \sum_{i=a}^b \frac{1}{(i+1)^{p/2}} &= \sum_{i=a+1}^{b+1} \frac{1}{i^{p/2}} \\ &\leq \frac{1}{(a+1)^{p/2}} + \int_{a+1}^{b+1} \frac{dx}{x^{p/2}} \end{aligned}$$

Bounding the integral on the right hand side, we get

$$\sum_{i=a}^b \frac{1}{(i+1)^{p/2}} \leq \begin{cases} 1 + \ln \frac{b+1}{a+1}, & \text{if } p = 2 \\ 1 + \frac{1}{(p/2-1)(a+1)^{p/2-1}}, & \text{if } p > 2 \end{cases}$$

The lemma then follows from the definition of (c, p) -compressibility. \square

5.2 Running Sum

Running sums can be defined as the circular convolution $x' * h$ of the sequences $h = (1, \dots, 1, 0, \dots, 0)$, where there are N ones and N zeros, and $x' = (x, 0, \dots, 0)$, where the private input x is padded with N zeros. An elementary computation reveals that $\hat{h}_1 = \sqrt{N}$ and $\hat{h}_i = O(N^{-1/2})$ for all $i > 1$. By Theorem 9, Algorithm 1 computes running sums with mean squared error $O(1)$ (ignoring dependence on ϵ and δ), improving on the bounds of [4, 9, 26] in the mean squared error regime.

5.3 Linear Filters in Time Series Analysis

Linear filtering is a fundamental tool in analysis of time-series data. A time series is modeled as a sequence $x = (x_t)_{t=-\infty}^{\infty}$, supported on a finite set of time steps. A filter converts the time series into another time series. A linear filter does so by computing the convolution of x with a series of *filter coefficients* w , i.e. computing $y_t = \sum_{i=-\infty}^{\infty} w_i x_{t-i}$. For a finitely supported x , y can be computed using circular convolution by restricting x to its support set and padding with zeros on both sides.

We consider the case where x is a time series of sensitive *events*. Each element x_i is a count of events or sum of values of individual transactions that have occurred at time step i . When we deal with values of transactions, we assume that individual transactions have much smaller value than the total. We emphasize that the definition of differential privacy with respect to x defined this way corresponds to *event-level privacy*. Semantically, this guarantee implies that even an adversary who has arbitrary information about all but a single event of interest cannot find out with certainty whether the event of interest has occurred. This guarantee is weaker than the user-level guarantee, which implies that knowing all events related to all but a single user of interest provides little information about the user. The user-level guarantee would unfortunately require excessive noise for filtering time series data, as the sensitivity of the convolution query becomes unbounded. On the other hand, the event-level guarantee is often sufficient, specifically in settings when sensitive events occur only infrequently.

We consider applications to financial analysis, but our methods are applicable to other instances of time series data, e.g. we may also consider network traffic logs or a time series of movie ratings on an online movie streaming service. We can perform almost optimal differentially private linear filtering by casting the filter as a circular convolution. Next we briefly describe a couple of applications of private linear filtering to financial analysis. For more references and detailed description, we refer the reader the book of Gençan, Selçuk, and Whitcher [11].

Volatility Estimation. The value at risk measure is used to estimate the potential change in the value of a good or financial instrument. Assume, for example, that in an online advertising system we would like to estimate potential changes in the number of clicks per day for a set of display ad campaigns, and

denote by x_i the number of clicks on day i from the start of the campaigns. The sensitive event is assumed to be a single ad click, for example a click on an ad for a type of medical treatment. In order to estimate volatility, we need to estimate a measure of the deviation of the x_i for a given time period $[t - W + 1, t]$. It is appropriate to take older fluctuations with less significance. One way to do this is by using linear filtering of the time series of absolute deviations in the click counts:

$$\sigma_t^e = \frac{1}{\sum_{i=1}^{W-1} \lambda^i} \sum_{i=0}^{W-1} \lambda^i |x_{t-i} - \bar{x}_{t-i}|,$$

where λ is a decay parameter and \bar{x}_t is the average count over $[t - W + 1, t]$. The quantity \bar{x}_t is itself given by the convolution $\frac{1}{W} \sum_{i=0}^{W-1} x_{t-i}$ and can be computed nearly optimally using Algorithm 1. Given the sequence \bar{x} , we can construct the time series $(y_i)_i = (|x_i - \bar{x}_i|)_i$. Using the triangle inequality, one can verify that for a fixed value of \bar{x} , $\|y - y'\|_1 \leq \|x - x'\|_1$, and therefore an algorithm which is differentially private with respect to y is also differentially private with respect to x . Therefore, we can use Algorithm 1 to estimate σ^e with nearly optimal mean squared error.

Computing \bar{x} was treated in [3] as the window sums problem, together with other decayed sum problems. The quantity σ^e is an exponentially decayed sum computed over a window and can be approximated under ε -differential privacy using the methods of [3]. However, as noted above, Algorithm 1 gives improved mean squared error guarantees for window sums, as well as a near-optimality guarantee.

Business Cycle Analysis. The goal of business cycle analysis is to extract cyclic components in the time series and smooth-out spurious fluctuation. Two classical methods for business-cycle analysis are the Hodrick-Prescott filter and the Baxter-King filter. Here we briefly sketch the form of the Hodrick-Prescott (HP) filter. Let us take the example of time series x of ad clicks again, with a single component x_i giving number of clicks on a set of ads per day or per hour. We can use the HP filter to detect cyclical trends in ad clicking activity. The filtered-out cyclical (smooth) component of the data extracted by the HP filter can be written as a convolution of the following form:

$$y_t^s = \frac{\theta_1 \theta_2}{\lambda} \left(\sum_{j=0}^{\infty} (A_1 \theta_1^j + A_2 \theta_2^j) (x_{t-j} + x_{t+j}) \right).$$

Above, λ is a smoothing parameter: the larger λ is, the more the data is smoothed by the filter; θ_i and A_i are functions of λ . In principle, this is a convolution of infinite time series, but in practice we truncate the series to a finite length.

5.4 Generalized Marginal Queries

Marginal queries are a class of queries posed to d -attribute binary databases, i.e. databases where each row of the database is associated with a d -bit binary

vector, corresponding to the values of d binary attributes. A marginal query is specified by a setting $\mathbf{a} \in \{0, 1\}^d$ of the d attributes and a subset $S \subseteq [d]$ of k attributes; the exact answer to the query is the number of rows in the database consistent with \mathbf{a} on S . In this subsection we address the error required to privately answer a natural generalization of marginal queries. A generalized marginal query is specified by a setting $\mathbf{a} \in \{0, 1\}^d$ of the d attributes and a w -DNF h and the exact answer is the number of rows $\mathbf{b} \in \{0, 1\}^d$ in the private database for which $h(\mathbf{a} \oplus \mathbf{b})$ is satisfied (here \oplus is componentwise XOR). In the case of traditional marginal queries the DNF h is a single disjunction of k unnegated variables. Generalized marginals however allow more complex queries such as, for example, “show all users who agree with \mathbf{a} on a_1 and at least one other attribute”.

More formally, we encode a binary d -attribute database in histogram representation as a function $x : \{0, 1\}^d \rightarrow [n]$. The value of $x(\mathbf{a})$ for $\mathbf{a} \in \{0, 1\}^d$ corresponds to the number of rows in the database with attribute setting \mathbf{a} , and n is the database size.

Definition 7. Let $h(\mathbf{c})$ be a w -DNF given by $h(\mathbf{c}) = (\ell_{1,1} \wedge \dots \wedge \ell_{1,w}) \vee \dots \vee (\ell_{s,1} \wedge \dots \wedge \ell_{s,w})$, where $\ell_{i,j}$ is a literal, i.e. either c_p or \bar{c}_p for some $p \in [d]$. The generalized marginal function for h and a database $x : \{0, 1\}^d \rightarrow [n]$ is a function $(x * h) : \{0, 1\}^d \rightarrow [n]$ defined by

$$(x * h)(\mathbf{a}) = \sum_{b \in \{0, 1\}^d} x(b) h(\mathbf{a} \oplus b).$$

The overload of notation for $x * h$ here is on purpose as generalized marginals can be interpreted as an instance of a generalization of circular convolutions. In particular, circular convolutions are associated naturally with the group of addition modulo N , while generalized marginals are an instance of convolutions associated with the group of addition modulo 2 of d -dimensional binary vectors (formally $(\mathbb{Z}/2\mathbb{Z})^d$). Moreover, there is a Fourier transform that diagonalizes convolutions over $(\mathbb{Z}/2\mathbb{Z})^d$ and that shares all properties with the transform defined in Section 2 which are necessary for our lower and upper bound arguments. In particular, we need that any component of any Fourier basis vector has norm $1/\sqrt{N}$, which is true for the Fourier transform diagonalizing convolutions over $(\mathbb{Z}/2\mathbb{Z})^d$. Therefore, we can privately approximate generalized marginal queries using Algorithm 1, and, furthermore, our analysis of the privacy and accuracy guarantees for the algorithm still holds. Using results from learning theory on the spectral concentration of bounded width DNFs and the bound from Section 5.1, we can show that Algorithm 1 gives non-trivial error for generalized marginal queries.

Theorem 12. Let h be a w -DNF and $x : \{0, 1\}^d \rightarrow [n]$ be a private database. Algorithm 1 satisfies (ε, δ) -differential privacy and computes the generalized marginal $x * h$ for h and x with mean squared error bounded by $O(\frac{\log(1/\delta)}{\varepsilon^2} 2^{d(1-1/O(w \log w))})$.

In addition to this explicit bound, we also know (by Theorem 14) that up to a factor of d^4 , Algorithm 1 is optimal for computing generalized marginal

functions. Notice that error bound we proved improves on randomized response by a factor of $2^{-\Omega(d/(w \log w))}$; interestingly this factor is independent of the size of the w -DNF formula.

In related work, Hardt et al. [14] considered database queries that can be computed by an AC0 circuit. Generalized marginal queries can be computed by a two-layer AC0 circuit. However, our results are incomparable to theirs, as they consider the setting where the database is of bounded size $\|\mathbf{x}\|_1 \leq n$ and our error bounds are independent of $\|x\|_1$. Our error bounds improve on the bounds of [14] when the database is large enough so that our error bound is sublinear in database size.

The proof of Theorem 12 follows from Lemma 4 and the following concentration result for the spectrum of w -DNF formulas, originally proved by Mansour [22] in the context of learning under the uniform distribution.

Theorem 13 ([22]). *Let $h : \{0, 1\}^d \rightarrow \{0, 1\}$ be a w -DNF. Let $\mathcal{F} \subseteq 2^{[d]}$ be the index set of the top 2^{d-k} Fourier coefficients of h . Then,*

$$\sum_{S \notin \mathcal{F}} |\hat{h}(S)|^2 \leq 2^{d + \frac{k-d}{O(w \log w)}}.$$

6 Conclusion

We derive nearly tight upper and lower bounds on the error of (ε, δ) -differentially private for computing convolutions. Our lower bounds rely on recent general lower bounds based on discrepancy theory and elementary linear algebra; our upper bound is a simple computationally efficient algorithm. We also sketch several applications of private convolutions, in time series analysis and in computing generalizes marginal queries on a d -attribute database.

Our results are nearly optimal for any h when the database size is large enough with respect to the number of queries. In some settings it is reasonable to assume however that database size is much smaller, and our algorithms give suboptimal error for such sparse databases. Nearly optimal algorithms for computing a workload of M linear queries posed to a database of size at most n were given in [24], but their algorithm has running time at least $O(M^2 N n)$. Since our dense case algorithm for computing convolutions has running time $O(N \log N)$, an interesting open problem is to give an algorithm with running time $O(N n \text{polylog}(N, n))$ for computing convolutions with optimal error when the database size is at most n .

References

- [1] BARAK, B., CHAUDHURI, K., DWORK, C., KALE, S., MCSHERRY, F., AND TALWAR, K. 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* (2007), ACM, pp. 273–282.

- [2] BHASKARA, A., DADUSH, D., KRISHNASWAMY, R., AND TALWAR, K. Unconditional differentially private mechanisms for linear queries. In *Proceedings of the 44th symposium on Theory of Computing* (New York, NY, USA, 2012), STOC '12, ACM, pp. 1269–1284.
- [3] BOLOT, J., FAWAZ, N., MUTHUKRISHNAN, S., NIKOLOV, A., AND TAFT, N. Private decayed sum estimation under continual observation. *Arxiv preprint arXiv:1108.6123* (2011).
- [4] CHAN, T., SHI, E., AND SONG, D. Private and continual release of statistics. In *ICALP* (2010).
- [5] CHERAGHCHI, M., KLIVANS, A., KOTHARI, P., AND LEE, H. Submodular functions are noise stable. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms* (2012), SIAM, pp. 1586–1592.
- [6] CORMODE, G., PROCOPIUC, C. M., SRIVASTAVA, D., AND YAROSLAVTSEV, G. Accurate and efficient private release of datacubes and contingency tables.
- [7] DINUR, I., AND NISSIM, K. Revealing information while preserving privacy. In *Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems* (2003), ACM, pp. 202–210.
- [8] DWORK, C., MCSHERRY, F., NISSIM, K., AND SMITH, A. Calibrating noise to sensitivity in private data analysis. In *TCC* (2006).
- [9] DWORK, C., PITASSI, T., NAOR, M., AND ROTHBLUM, G. Differential privacy under continual observation. In *STOC* (2010).
- [10] DWORK, C., ROTHBLUM, G., AND VADHAN, S. Boosting and differential privacy. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on* (2010), IEEE, pp. 51–60.
- [11] GENÇAY, R., SELÇUK, F., AND WHITCHER, B. *An Introduction to Wavelets and Other Filtering Methods in Finance and Economics*. Elsevier Academic Press, 2002.
- [12] GRAY, R. M. Toeplitz and circulant matrices: a review. *Foundations and Trends in Communications and Information Theory* 2, 3 (2006), 155–239.
- [13] GUPTA, A., HARDT, M., ROTH, A., AND ULLMAN, J. Privately releasing conjunctions and the statistical query barrier. In *Proceedings of the 43rd annual ACM symposium on Theory of computing* (2011), ACM, pp. 803–812.
- [14] HARDT, M., ROTHBLUM, G., AND SERVEDIO, R. Private data release via learning thresholds. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms* (2012), SIAM, pp. 168–187.

- [15] HARDT, M., AND TALWAR, K. On the geometry of differential privacy. In *Proceedings of the 42nd ACM symposium on Theory of computing* (2010).
- [16] KASIVISWANATHAN, S., RUDELSON, M., SMITH, A., AND 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* (2010), ACM, pp. 775–784.
- [17] LI, C., HAY, M., RASTOGI, V., MIKLAU, G., AND MCGREGOR, A. Optimizing linear counting queries under differential privacy. In *Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems* (New York, NY, USA, 2010), PODS '10, ACM, pp. 123–134.
- [18] LI, C., AND MIKLAU, G. An adaptive mechanism for accurate query answering under differential privacy. *PVLDB* 5, 6 (2012), 514–525.
- [19] LI, C., AND MIKLAU, G. Measuring the achievable error of query sets under differential privacy. *CoRR abs/1202.3399* (2012).
- [20] LOVÁSZ, L., SPENCER, J., AND VESZTERGOMBI, K. Discrepancy of set-systems and matrices. *European Journal of Combinatorics* 7, 2 (1986), 151–160.
- [21] LOVÁSZ, L., AND VEMPALA, S. Fast algorithms for logconcave functions: Sampling, rounding, integration and optimization. In *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on* (2006), IEEE, pp. 57–68.
- [22] MANSOUR, Y. An $o(n \log \log n)$ learning algorithm for dnf under the uniform distribution. *Journal of Computer and System Sciences* 50, 3 (1995), 543–550.
- [23] MUTHUKRISHNAN, S., AND NIKOLOV, A. Optimal private halfspace counting via discrepancy. *Proceedings of the 44th ACM symposium on Theory of computing* (2012).
- [24] NIKOLOV, A., TALWAR, K., AND ZHANG, L. The geometry of differential privacy: the sparse and approximate cases.
- [25] THALER, J., ULLMAN, J., AND VADHAN, S. Faster algorithms for privately releasing marginals. *Automata, Languages, and Programming* (2012), 810–821.
- [26] XIAO, X., WANG, G., AND GEHRKE, J. Differential privacy via wavelet transforms.

A Spectrum Partitioning Algorithm

We partition the spectrum of the convolution matrix \mathbf{H} into geometrically growing in size groups and adds different amounts of noise to each group. Noise is added in the Fourier domain, i.e. to the Fourier coefficients of the private input \mathbf{x} . The most noise is added to those Fourier coefficients which correspond to small (in absolute value) coefficients of \mathbf{h} , making sure that privacy is satisfied while the least amount of noise is added. In the analysis of optimality, we show that the noise added to each group can be charged to the lower bound $\text{specLB}(\mathbf{h})$. Because the number of groups is logarithmic in N , we get almost optimality. This analysis is inspired by the work of Hardt and Talwar [15]. However, our algorithm is simpler and significantly more efficient.

The (ε, δ) -differentially private algorithm we propose for approximating $h * x$ is shown as Algorithm 2. In the remainder of this section we assume for simplicity that N is a power of 2. We also assume, for ease of notation, that $|\hat{h}_0| \geq \dots \geq |\hat{h}_{N-1}|$. Our algorithm and analysis do not depend on i except as an index, so this comes without loss of generality.

Algorithm 2 SPECTRALPARTITION

```

Set  $\eta = \frac{\sqrt{2(1+\log N) \ln(1/\delta)}}{\varepsilon}$ 
Compute  $\hat{\mathbf{x}} = \mathbf{F}_N \mathbf{x}$  and  $\hat{\mathbf{h}} = \mathbf{F}_N \mathbf{x}$ .
 $\tilde{x}_0 = \hat{x}_0 + \text{Lap}(\eta)$ 
for all  $k \in [1, \log N]$  do
  for all  $i \in [N/2^k, N/2^{k-1} - 1]$  do
    Set  $\tilde{x}_i = \hat{x}_i + \text{Lap}(\eta 2^{-k/2})$ .
    Set  $\tilde{y}_i = \sqrt{N} \hat{h}_i \tilde{x}_i$ .
  end for
end for
Output  $\tilde{\mathbf{y}} = \mathbf{F}_N^H \tilde{\mathbf{y}}$ 

```

Lemma 4. *Algorithm 2 satisfies (ε, δ) -differential privacy. Also, there exists an absolute constant C such that Algorithm 2 achieves expected mean squared error*

$$\text{MSE} \leq C \frac{(1 + \log N) \log(1/\delta)}{\varepsilon^2} (|\hat{h}_0|^2 + \sum_{k=1}^{\log N} \frac{1}{2^k} \sum_{i=N/2^k}^{N/2^{k-1}-1} |\hat{h}_i|^2). \quad (10)$$

Proof. Privacy. We claim that $\tilde{\mathbf{x}}$ is an (ε, δ) -differentially private function of \mathbf{x} . The other computations depend only on \mathbf{h} and $\tilde{\mathbf{x}}$ and not on \mathbf{x} directly, so, by Theorem 3, incur no loss in privacy.

First we analyze the sensitivity of each Fourier coefficient \hat{x}_i . As a function of \mathbf{x} , \hat{x}_i is an inner product of \mathbf{x} with a Fourier basis vector. Let that vector be \mathbf{f} and let \mathbf{x}, \mathbf{x}' be two neighboring inputs, i.e. $\|\mathbf{x} - \mathbf{x}'\|_1 \leq 1$. Then we have

$$|\mathbf{f}^H (\mathbf{x} - \mathbf{x}')| \leq \|\mathbf{f}\|_\infty \|\mathbf{x} - \mathbf{x}'\|_1 \leq \frac{1}{\sqrt{N}}$$

Therefore, by Theorem 2, when $i \in [N/2^k, N/2^{k-1}-1]$, \tilde{x}_i is $(\frac{2^{k/2}}{\sqrt{N}\eta}, 0)$ -differentially private. By Theorem 4, $\tilde{\mathbf{x}}$ is (ε', δ) differentially private for any $\delta > 0$, where

$$\begin{aligned}\varepsilon'^2 &= 2 \ln(1/\delta) \left(\frac{1}{\eta^2} + \sum_{k=1}^{\log N} \frac{N}{2^k} \frac{2^k}{N\eta^2} \right) \\ &= 2 \ln(1/\delta) \frac{1 + \log N}{\eta^2} = \varepsilon^2\end{aligned}$$

Accuracy. Observe $E[\tilde{x}_i] = \hat{x}_i$ since we add unbiased Laplace noise to each \hat{x}_i . Also, the variance of $\text{Lap}(\eta 2^{-k/2})$ is $2\eta^2 2^{-k}$. Therefore, $E[\bar{y}_i] = \sqrt{N} \hat{h}_i \hat{x}_i$ and the variance of \bar{y}_i when $i \in [N/2^k, N/2^{k-1}-1]$ is $O(N |\hat{h}_i|^2 \eta^2 2^{-k})$. By linearity of expectation, $E[\mathbf{F}_N^H \bar{\mathbf{y}}] = \mathbf{H}\mathbf{x}$. Adding variances for each k and dividing by N , we get the right hand side of (10). The proof is completed by observing that the inverse Fourier transform \mathbf{F}_N^H is an isometry for the ℓ_2 norm, so does not change mean squared error. \square

Theorem 14. For any \mathbf{h} , Algorithm 2 satisfies (ε, δ) -differential privacy and achieves expected mean squared error $O(\text{specLB}(\mathbf{h}) \frac{\log^4 N \ln(1/\delta)}{\varepsilon^2})$.

Proof. By Lemma 4, we know that

$$\text{MSE} \leq C \frac{\log N \log(1/\delta)}{\varepsilon^2} (|\hat{h}_0|^2 + \sum_{k=1}^{\log N} \frac{N}{2^{2k}} |\hat{h}_{N/2^{k-1}-1}|^2) = O(\text{specLB}(\mathbf{h}) \frac{\log^4 N \ln(1/\delta)}{\varepsilon^2}).$$

\square

B Closed Form Solution for the Optimal $\mathcal{A}(\mathbf{b})$

We derive a closed form solution of (7)–(9) using convex programming duality. Let us first rewrite the program by substituting $a_i = 1/b_i^2$:

$$\begin{aligned}\min_{\{a_i\}_{i \in I}} \quad & \sum_{i \in I} \frac{|\hat{h}_i|^2}{a_i} \\ \text{s.t.} \quad & \sum_{i \in I} a_i = \frac{N\varepsilon^2}{2 \ln(1/\delta)} \\ & a_i \geq 0, \quad \forall i \in I.\end{aligned}\tag{11}$$

The Lagrangian is

$$L(\mathbf{a}, \nu, \Lambda) = \sum_{i \in I} \frac{|\hat{h}_i|^2}{a_i} + \nu \left(\sum_{i \in I} a_i - \frac{N\varepsilon^2}{2 \ln(1/\delta)} \right) - \sum_{i \in I} \lambda_i a_i.\tag{12}$$

The KKT conditions are given by

$$\begin{aligned}
\forall i \in I, \quad & -\frac{|\hat{h}_i|^2}{a_i^2} + \nu - \lambda_i = 0 \\
\sum_{i \in I} a_i - \frac{N\varepsilon^2}{2\ln(1/\delta)} &= 0 \\
\lambda_i a_i &= 0 \\
a_i \geq 0, \lambda_i &\geq 0
\end{aligned} \tag{13}$$

The following solution $(\mathbf{a}^*, \nu^*, \Lambda^*)$ satisfies the KKT conditions, and is thus the optimal solution to (11)

$$\forall i \in I, \quad a_i^* = \frac{N\varepsilon^2}{2\ln(1/\delta)\|\hat{\mathbf{h}}\|_1} |\hat{h}_i|, \quad \lambda_i^* = 0, \quad \nu^* = \left(\frac{2\ln(1/\delta)\|\hat{\mathbf{h}}\|_1}{N\varepsilon^2} \right)^2. \tag{14}$$

Consequently, the optimal noise parameters \mathbf{b} for the original problem (7)–(9), and the associated MSE are

$$\begin{aligned}
b_i^* &= \begin{cases} \sqrt{\frac{2\ln(1/\delta)\|\hat{\mathbf{h}}\|_1}{N\varepsilon^2|\hat{h}_i|}} & \text{if } i \in I \\ 0 & \text{if } i \notin I \end{cases} \\
\text{MSE}^* &= 2 \sum_{i \in I} |\hat{h}_i|^2 b_i^2 = 4 \frac{\ln(1/\delta)}{\varepsilon^2 N} \|\hat{\mathbf{h}}\|_1^2,
\end{aligned} \tag{15}$$

which are the noise parameters and MSE of Algorithm 1.