Open Problems On Rigorous Notions of Privacy from the DIMACS Workshop of October 2012

November 15, 2012

Contents

1 Introduction

To be written (probably by Adam).

2 Technical Questions About Differential Privacy

2.1 Efficient Algorithms for Releasing Conjunctions

Contributed by: Adam Smith, Aaron Roth, Jon Ullman

Consider the universe $\mathcal{X} = \{0,1\}^d$. A function $f : \mathcal{X} \to \{0,1\}$ is a *conjunction* if it is the logical AND of literals, each of which consists of a bit of the input or its negation. Let C be the set of all 3^d nontrivial conjunctions, and C_k be the set of conjunctions that involve only k literals.

For simplicity, assume the size of the database n is public, and the answer to query f on data set X is the average $\frac{1}{n}\sum_{x\in X} f(x)$. For asymptotic notation, we take ϵ constant and $\log(1/\delta) = O(\log n)$.

Open Question 1. *Is there a (ϵ,δ)-differentially private query release mechanism that can answer all queries in C_k to additive error $\mathrm{poly}(d,k)/n$ that runs in time $\mathrm{poly}(d^k)$ (or even simply subexponential in d)?*

Background The Laplace mechanism applied to this problem directly runs in time $O(\binom{d}{k})$ and adds expected noise $\tilde{O}(2^k\binom{d}{k}/n)$ per entry (ignoring log factors, this is an upper bound on the error for all answers with high probability), or $\sqrt{2^k\binom{d}{k}}/n$ with (ϵ,δ)-differential privacy. Various optimizations improve the dependency on k [BCD+07].

Blum et al. [BLR08] and subsequent work [DNR+09, RR10, HR10b, GRU12a, HLM12a] give algorithms with additive error only $\tilde{O}\left(\frac{\mathrm{poly}(k,d)}{\sqrt{n}}\right)$. Unfortunately, even the most efficient of these algorithms [HR10b] has running time that is linear in $|\mathcal{X}| = 2^d$.

An even more recent line of work [GHRU11, HRS11, TUV12] gave algorithms for releasing conjunctions with subexponential error in subexponential time. [TUV12], for example, give algorithms for releasing k-way marginals with running time $\mathrm{poly}(n)$ and error $o(1)$ provided $n = d^{\omega(\sqrt{k})}/\epsilon$. See [TUV12] for a succinct summary of the state of the art.

The lower bounds are far from matching the upper bounds. [KRSU10] show that the error of (ϵ,δ)-d.p. algorithms for releasing all k-way marginals must be $\tilde{\Omega}(\min\{\frac{1}{\sqrt{n}}, \frac{d^{k/2}}{n}\})$ (regardless of running time). That bound only rules out $o(1)$ error when $n \gg d^{k/2}$. [UV11] show that *efficient, synthetic data* releases which preserve all 2-way marginals up to $o(1)$ additive error must have running time superpolynomial in d, but this does not rule out other types of synopses.

2.2 Efficient Algorithms for Answering Convex Queries?

Contributed by: Zhiyi Huang and Aaron Roth

Let $\mathcal{X} = [0,1]^\ell$ be the ℓ-dimensional unit rectangle endowed with the Euclidean norm, and let $S \subseteq \{\phi : [0,1]^\ell \to [0,1]\}$ be the collection of predicates such that for each $\phi \in S$:

1. ϕ is 1-Lipschitz: for all $x, y \in [0,1]^\ell$, $|\phi(x) - \phi(y)| \le ||x - y||_2$
2. ϕ is convex: for all $x, y \in [0,1]^\ell$ and for all $t \in [0,1]$, $\phi(tx + (1-t)y) \le t\phi(x) + (1-t)\phi(y)$

For each $\phi \in S$, define the query $f_\phi(D) = \frac{1}{n}\sum_{x\in D}\phi(x)$. Then:

Open Question 2. *Let $C = \{f_\phi : \phi \in S\}$ denote the set of 1-Lipschitz, convex linear queries defined over the universe $\mathcal{X} = [0,1]^\ell$. Is there a differentially private query release mechanism operating in the interactive setting, that can answer any subset of k queries from C to additive error $\tilde{O}(\mathrm{poly}(\ell, \log(k))/\sqrt{n})$ with per-query update time poly(ℓ, n)?*

Background The Laplace mechanism gives a computationally efficient algorithm for answering any collection of k linear queries on a database of size n defined over a universe $\mathcal{X}$ to additive error $\tilde{O}\left(\frac{\sqrt{k}}{n}\right)$ [DMNS06, DRV10], and the running time of this mechanism is only $O(n)$ per query (i.e. the time it takes to evaluate the query itself). On the other hand, a series of results [BLR08, DNR+09, RR10, HR10b, GHRU11, GRU12a, HLM12a] has shown that it is in principle possible to answer any k such queries with error that scales only like $\tilde{O}\left(\frac{\text{poly}(\log k, \log |\mathcal{X}|)}{\sqrt{n}}\right)$, which allows a mechanism to privately answer nearly exponentially many queries in the size of the database to non-trivial accuracy. Unfortunately, even the most efficient of these algorithms [HR10b] has running time that is linear in $|\mathcal{X}|$ (i.e. *exponential* in the dimension of the space), which can be prohibitive. Moreover, recent results have shown that modulo mild hardness assumptions, this running time is the best possible for any mechanism that can answer more than $O(n^2)$ general linear queries to non-trivial accuracy [DNR+09, Ull12]. In other words: the Laplace mechanism has essentially optimal accuracy among polynomial time algorithms that can answer queries at a comparable level of generality.

These results do not rule out mechanisms which can answer *special, structured* subclasses of linear queries to accuracy that approaches the best information-theoretic results, while maintaining computational efficiency. Convex, Lipschitz queries seem like a natural candidate. A natural example of a convex query is an ℓ_2 distance query: such a query is specified by a point in the metric space, and asks for the *average distance* from the query point to the database points. It was recently shown that it is indeed possible to give computationally efficient algorithms for answering such queries (indeed, distance queries defined over *arbitrary* metrics) [HR12d]. This result crucially uses the metric structure of such queries, however. Is it possible to get a similar result for the set of all convex, Lipschitz queries?

We remark that the question is also interesting without the Lipschitz constraint: do there exist efficient algorithms for the set of *convex* queries? However, we note that without the Lipschitz constraint, the set of convex queries includes boolean conjunctions, because the convex function:

$$f_S(x) = \left(\frac{1}{|S|}\sum_{i \in S} x_i\right)^t$$

approaches the value of the boolean conjunction on variables S on the vertices of the boolean hypercube, as the exponent t approaches infinity. Boolean conjunctions are themselves an important challenge problem in differential privacy.

2.3 Efficient, Synthetic Data for Graph Cut Queries?

Contributed by: Aaron Roth

Consider a d vertex weighted graph with a publicly known vertex set and a private edge set. Such a graph can be described by a matrix $A \in [0,1]^{d \times d}$, where $A(i,j) = A(j,i) = w(i,j)$ is defined to be the weight of the edge between vertex i and j. Write m to denote the total weight of the edge set: $m = \sum_{i=1}^{d}\sum_{j=1}^{d} A(i,j)$. Two such graphs A and A' are defined to be *edge neighbors* if there exist indices i^*, j^* such that for all $(i,j) \neq (i^*, j^*)$, $A(i,j) = A'(i,j)$ and $A(j,i) = A'(j,i)$: i.e. they are identical except in a single edge. We can ask for differentially private algorithms with respect to the edge neighbor relation on graphs.

A cut query between two subsets of vertices $S, T \subseteq V$ is defined to be the sum weight of the edges crossing between S and T:

$$f_{S,T}(A) = \sum_{i \in S} \sum_{j \in T} A(i,j)$$

The set of all cut queries is denoted $Q_{\text{cut}} = \{f_{S,T} : S, T \subseteq V\}$. Note that cut queries are 1-sensitive linear queries on the private database, and that there are $|Q_{\text{cut}}| = O(2^{2d})$ such queries in total.

Do there exist efficient, privacy preserving algorithms for non-interactively answering all cut queries to optimal accuracy?

Open Question 3. *Is there a 1-differentially private algorithm running in time polynomial in d, that for any graph A can produce a synthetic graph $\hat{A}$ such that for all $S, T \subseteq V$:*

$$f_{S,T}(A) - \tilde{O}(\sqrt{m \cdot d}) \leq f_{S,T}(\hat{A}) \leq f_{S,T}(A) + \tilde{O}(\sqrt{m \cdot d})$$

Open Question 4. *Is there a 1-differentially private algorithm running in time polynomial in d, that for any graph A can produce a synthetic graph $\hat{A}$ such that for all $S, T \subseteq V$:*

$$0.99 \cdot f_{S,T}(A) - \tilde{O}(d) \leq f_{S,T}(\hat{A}) \leq 1.01 \cdot f_{S,T}(A) + \tilde{O}(d)$$

Background The problem of privately answering graph cut queries was first considered by [GHRU11, GRU12a]. These are an attractive class of linear queries, because they seem to avoid one of the main obstacles for developing efficient, non-interactive query release mechanisms: namely, they are defined over a universe (the edge set of the complete graph) which is polynomial, not exponential, in the size of the database (the edge set of the private graph). As such, *interactive* mechanisms such as multiplicative weights can be efficiently run on such queries [HR10b]. Nevertheless, efficient algorithms for synthetic data are lacking. It was observed in [GRU12a] that simple input perturbation (i.e. adding noise $\text{Lap}(1/\epsilon)$ to each entry of A) can be used to produce synthetic data which answers every cut query to additive error $O(d^{3/2})$. However, this is suboptimal when $m \ll d^2$. For example, private multiplicative weights produces synthetic data which achieves the error from Question 3 when all $k = 2^{2d}$ cut queries are asked – however, this takes time proportional to the number of queries asked [HR10b, GRU12a]. Moreover, the net mechanism paired with a net of Benczur Karger cut sparsifiers produces synthetic data which achieves the error from Question 4, but again, this mechanism is not computationally efficient [BLR08, BBDS12]. Blocki et al. [BBDS12] give a mechanism which produces a data structure which can answer $(S, \bar{S})$ cut queries with constant multiplicative error and additive error $O(|S|\sqrt{d})$. This improves over the input perturbation bound of $O(d^{3/2})$ for cuts of the form $(S, \bar{S})$ when $|S| \ll d$. It does not however improve over input perturbation for large cuts.

One way to efficiently achieve the bounds of 3 would be to give a private algorithm for approximating the cut-norm of a graph, which corresponds to the *private distinguishing* problem of finding the cut query which most differs on two different graphs A, A' [GRU12a]. This would allow an algorithm to efficiently find the update queries necessary for multiplicative weights, which gives an efficient offline algorithm [GHRU11]. Note that good non-private algorithms are known for efficiently approximating the cut-norm [AN06].

Note that if the analyst is only interested in the answers to a polynomial number of k cut queries, rather than all cut queries, then the input perturbation bound improves to additive error

$O(d\sqrt{\log k})$, and private multiplicative weights can be run in time polynomial in d and k, and gives synthetic data with additive error $\tilde{O}(\sqrt{m \log k})$. Note also that the problem of approximating a normalized variant of the cut-norm is related to the problem of computing the top singular vector of the matrix A. Near optimal, efficient algorithms are known for this task: [KT13, CSS12, HR12a] but these algorithms all require some assumption on the spectral gap of the matrix, which precludes their black-box use with graphs produced as intermediate steps using algorithms like private multiplicative weights.

2.4 Reduction Hypothesis Under the ϵ-Matrix Mechanism

Contributed by: Chao Li, Gerome Miklau

Designing strategy matrix is core to the matrix mechanism[LHR$^+$10]. Given a workload matrix $\mathbf{W}$ and a strategy matrix $\mathbf{A}$, the expected squared error of answering $\mathbf{W}$ using $\mathbf{A}$ under the ϵ-matrix mechanism is $||\mathbf{A}||_1^2 trace(\mathbf{W}^T\mathbf{W}(\mathbf{A}^T\mathbf{A})^+)/\epsilon^2$. Here $||\mathbf{A}||_1$ is the maximum L_1 norm of columns of $\mathbf{A}$, ^{T} denotes the transpose of a matrix and $^+$ denotes the pseudo inverse of a matrix. Given two strategy matrices $\mathbf{A}$ and $\mathbf{B}$, if $\mathbf{A}^T\mathbf{A} = \mathbf{B}^T\mathbf{B}$ and $||\mathbf{B}||_1 \leq ||\mathbf{A}||_1$, we say strategy $\mathbf{A}$ is dominated by strategy $\mathbf{B}$ since $\mathbf{B}$ can answer any query that can be answered by $\mathbf{A}$ with smaller or the same error. Since $\mathbf{A}^T\mathbf{A} = \mathbf{B}^T\mathbf{B}$, the number of queries in $\mathbf{B}$ is at least the rank of $\mathbf{A}$. A natural question is that given a strategy $\mathbf{A}$ with rank r, whether can we find another $r \times n$ strategy that dominates $\mathbf{A}$.

Open Question 5. *Given an $m \times n$ matrix $\mathbf{A}$ with rank r, whether there exists an $r \times n$ matrix $\mathbf{B}$ such that $\mathbf{A}^T\mathbf{A} = \mathbf{B}^T\mathbf{B}$ and $||\mathbf{B}||_1 \leq ||\mathbf{A}||_1$.*

Background When searching for a good strategy in the matrix mechanism, the size of the matrix is always fixed as the size of the domain[LHR$^+$10] or proportional to the rank of the workload[YZW$^+$12, LM12]. However it is not clear whether there exist better strategy matrices with larger size. If the answer to Question 5 is yes, it is guaranteed that there always exists an optimal strategy under the ϵ-matrix mechanism with the least number of queries.

2.5 Are there subexponential-time local Lipschitz reconstructors?

Contributed by: Madhav Jha and Sofya Raskhodnikova

A function $f : \{1, \ldots, n\}^d \to \mathbb{R}$ is c-Lipschitz (with respect to the ℓ_1 metric on $\{1, \ldots, n\}^d$) if $|f(x) - f(y)| \leq c \cdot \|x - y\|_1$ for all points x, y in the domain $\{1, \ldots, n\}^d$. We say f is *Lipschitz* (or satisfies the Lipschitz property) if f is 1-Lipschitz. This problem is concerned with *local reconstruction* of the Lipschitz property. As explained in [JR11], *local reconstructors* (which are also called *local filters*) can be used in a *filter mechanism*. This mechanism is designed for the privacy settings where the client specifies a query by sending an arbitrary program f to the database curator asking to evaluate it on the database x; moreover, the client claims that the function f specified by the program has low global sensitivity. A local filter for the Lipschitz property can be used in a mechanism based on global sensitivity (such as the Laplace mechanism) to obtain algorithms which are (i) differentially private even if the client is misreporting the global sensitivity of her program and (ii) as accurate as the Laplace mechanism for honest clients.

Beginning with [ACCL07], the definition of property reconstructors has been strengthened and relaxed in several ways [SS10, BGJ$^+$12, JR11, AJMR12]. We give the most relevant definition from

the privacy point of view, based on [AJMR12]. Intuitively, a local filter can be used to replace oracle access to an arbitrary function with oracle access to a related Lipschitz function.

Definition 1. *A* local filter *for the Lipschitz property is a randomized algorithm A which gets oracle access to an input function $f : D \to \mathbb{R}$ and an input $x \in D$. The randomness of algorithm A is specified by a string ρ (the "random seed"), so that for fixed f and ρ, algorithm A runs deterministically on input x and produces output $A_{f,\rho}(x) \in \mathbb{R}$. (In particular, a local filter has no internal state to store previously made queries – this property is the reason for calling it "local".) The function $g(x) = A_{f,\rho}(x)$ output by the filter must obey the following conditions:*

1. *For each f and ρ, the function g must be Lipschitz.*
2. *If f is Lipschitz then with high probability (over the choice of ρ) g should be close to f: $\forall x \in D$, $|f(x) - g(x)|$ should be small.*

Open Question 6. *What is the time complexity of local Lipschitz reconstruction of functions $f : \{1, \ldots, n\}^d \to \mathbb{R}$?*

Best known upper bound: [JR11] gave a deterministic local Lipschitz filter for functions $f : \{1, \ldots, n\}^d \to \mathbb{R}$ which runs in time $O((\log n+1)^d)$. Their filter guarantees that the output function g is identical to f (and not just close to f in ℓ_∞) if f is Lipschitz.

Best known lower bound: [AJMR12] showed that every (possibly randomized and adaptive) local Lipschitz filter for functions $f : \{0,1\}^d \to \mathbb{R}$ must make $2^{\Omega(d)}$ look ups to its oracle f even if the filter is allowed to make an $\Omega(d)$ additive error on every output.

Possible relaxations of local reconstructors. In the light of the exponential lower bound above, the second open question is about relaxations of Definition 1. Specifically, item 1 of Definition 1 may be relaxed to only require that the filter output κ-Lipschitz function for some constant $\kappa > 1$. It may be further relaxed to require that g is κ-Lipschitz *with high probability*. We call such filters *κ-relaxed*. Another relaxation, suggested by Or Sheffet, is to require item 2 to hold only when function f belongs to a particular class $\mathcal{H}$. We say such filters are *accurate on $\mathcal{H}$*. It is not hard to see that these relaxations still yield useful filter mechanisms. (The first relaxation will cause additional noise proportional to κ/ϵ, where ϵ is the parameter of ϵ-differential privacy. Requiring item 1 to hold only with high probability will yield (ϵ, δ)-differential privacy instead of "pure" ϵ-differential privacy. Finally, the relaxation to class $\mathcal{H}$ will cause the mechanism to be accurate only on programs which specify functions from $\mathcal{H}$.)

Open Question 7. *What is the time complexity of the relaxed local filters, defined above? Specifically, are there* κ-relaxed local Lipschitz filters *for functions $f : \{0,1\}^d \to \mathbb{R}$ which run in time $2^{o(d)}$? Also, is there an interesting class $\mathcal{H}$ of functions $f : \{0,1\}^d \to \mathbb{R}$ such that there are local Lipschitz filters which are* accurate on $\mathcal{H}$ *and run in time $2^{o(d)}$?*

2.6 Competitive online mechanisms

Contributed by: Aleksandar Nikolov

Let us represent a database $D \in U^n$ drawn from a universe U as a histogram $x \in \mathbb{N}^{|U|}$ ($\|x\|_1 \leq n$ is the database size). Let us further represent a set of d counting queries as a matrix $A \in \{0,1\}^{d \times |U|}$ in the natural way. The mean squared error of a mechanism $\mathcal{M}$ on the query set A is

$$\mathrm{err}_{\mathcal{M}}(A) = \max_x \mathbb{E}\frac{1}{d}\|\mathcal{M}(x) - Ax\|_2^2,$$

where the expectation is taken over the randomness of $\mathcal{M}$. The optimal error for A and n is $\mathrm{opt}_{\epsilon,\delta}(A,n) = \min_{\mathcal{M}}(\mathrm{err}_{\mathcal{M}}(A))$, where the minimum is over (ϵ,δ)-differentially private mechanisms.

An online mechanism is initially given x. At time step i, the mechanism is given the i-th row of A and must approximate $(Ax)_i$. Let $A_{\leq i}$ be the matrix consisting of the first i rows of A. An (ϵ,δ)-differentially private online mechanism $\mathcal{M}$ is C-competitive if at any time step i,

$$\mathrm{err}_{\mathcal{M}}(A_{\leq i}) \leq C \cdot \mathrm{opt}_{\epsilon,\delta}(A_{\leq i})$$

Open Question 8. *Does there exist an online mechanism which is C-competitive for all A and for some non-trivial C? What about restricted classes of A? What is the optimal value of C?*

Background There exist (ϵ,δ)-differentially private online mechanisms [HR10a, GRU12b] which answer any set of d counting queries on databases of size at most n with mean squared error $O(n)$ (ignoring dependence on ϵ, δ and $|U|$). Since any non-trivial set of linear queries requires error $\Omega(1)$, these mechanisms are $O(n)$-competitive for counting queries on databases of size at most n. Furthermore, simple randomized response is $O(N)$-competitive for counting queries by the same reasoning. Finally, the Gaussian noise mechanism [BDMN05] is $O(d)$-competitive for d counting queries.

2.7 Dependence of error on universe size

Contributed by: Aleksandar Nikolov

Let us represent a database $D \in U^n$ drawn from a universe U as a histogram $x \in \mathbb{N}^{|U|}$, where the database size is $\|x\|_1 \leq n$. Let us further represent a set of d counting queries as a matrix $A \in \{0,1\}^{d \times |U|}$ in the natural way. The mean squared error of a mechanism $\mathcal{M}$ on the query set A and databases of size at most n is

$$\mathrm{err}_{\mathcal{M}}(A,n) = \max_{x:\|x\|_1 \leq n} \mathbb{E}\frac{1}{d}\|\mathcal{M}(x) - Ax\|_2^2,$$

where the expectation is taken over the randomness of $\mathcal{M}$. The optimal error for A and n is $\mathrm{opt}_{\epsilon,\delta}(A,n) = \min_{\mathcal{M}}(\mathrm{err}_{\mathcal{M}}(A,n))$, where the minimum is over (ϵ,δ)-differentially private mechanisms

Open Question 9. *Does there exist a family of* 0-1 *matrices $\{A_i\}_{i=1}^{\infty}$, A_i of dimensions $d \times |U_i|$, for which $\mathrm{opt}_{\epsilon,\delta}(A_i,n)$ grows to infinity with $|U_i|$ for fixed n, ϵ, and $\delta > 0$? What is the optimal worst-case dependence of $\mathrm{opt}_{\epsilon,\delta}(A,n)$ on $|U|$?*

Background There exist (ϵ,δ)-differentially private mechanisms [HR10a, GRU12b, HLM12b, NTZ12] which can answer any set of d counting queries with error

$$\mathrm{err}_{\mathcal{M}}(A,n) = O\left(\frac{n\sqrt{\log(1/\delta)\log|U|}}{\epsilon}\right).$$

On the other hand, the best known lower bounds on $\mathrm{opt}_{\epsilon,\delta}(A,n)$ for databases of size at most n are given by matrices A of dimensions $O(n) \times n$, i.e. the universe size is n. For example, for a random $O(n) \times n$ 0-1 matrix A, $\mathrm{opt}(A,n) = \Omega\left(\frac{n}{\epsilon}\right)$ with constant probability [DN03a]. A lower bound technique which uses queries over a universe of size n cannot show that the optimal error for counting queries must grow with $|U|$ for fixed n.

For pure differential privacy, i.e. when $\delta = 0$, we know packing-based lower bounds on $\mathrm{opt}_{\epsilon,0}(A,n)$ that depend logarithmically on $|U|$. Specifically, we know [HT10, Har11] that for $d \geq n^{1.1}$, and for a random $d \times |U|$ 0-1 matrix A, with constant probability

$$\mathrm{opt}_{\epsilon,0}(A,n) = \Omega\left(\frac{n \log(|U|/n)}{\epsilon}\right).$$

The tight dependence on $|U|$ for $(\epsilon, 0)$-differentially private mechanisms is also open. The best known upper bound [NTZ12] on $\mathrm{opt}_{\epsilon,0}(A,n)$ is $O(\frac{1}{\epsilon} n \,\mathrm{polylog}(d) \log^{3/2}|U|)$.

2.8 Removing the Square Root of Rank

Contributed by: Moritz Hardt and Aaron Roth

This problem is about privacy-preserving low-rank approximation in the spectral norm. The coherence of a symmetric $n \times n$ matrix A given in its singular value decomposition $A = U\Sigma U^t$ is defined as $\mu(A) = n\|U\|_\infty^2$.

Open Question 10. *Is there an (ϵ,δ)-differentially private algorithm which given an $n \times n$ matrix A and a number $1 \leq k \leq n$, returns a rank-k matrix B such that with probability $2/3$,*

$$\|A - B\|_2 \leq \|A - A_k\|_2 + O\big(\epsilon^{-1}\mathrm{poly}(k \log(1/\delta))\mu(A)\big)$$

Here, A_k is the best rank k approximation of A in the spectral norm which is denoted by $\|\cdot\|_2$.

Background. For motivation of coherence in the contect of privacy see [HR12b, HR12c]. The latter result showed the above bound up to a factor of $\sqrt{\mathrm{rank}(A)}$ in front of the coherence term. The question is to remove this dependence on the rank. The main motivation for this problem is that the above bound does not depend on the dimension n, but only on k and the coherence parameter $\mu(A)$ which can be significantly smaller than n.

3 Technical Questions Relating to Other Definitions

3.1 Necessary and Sufficient Conditions for Self-Composability

Contributed by: Ashwin Machanavajjhala

A very attractive property of differential privacy is that it linearly self-composes.

Definition 2. *Given two mechanisms $\mathfrak{M}_1$ and $\mathfrak{M}_2$, $\mathfrak{M}^*_{\mathfrak{M}_1,\mathfrak{M}_2}$ is the algorithm with $range(\mathfrak{M}^*_{\mathfrak{M}_1,\mathfrak{M}_2}) = range(\mathfrak{M}_1) \times range(\mathfrak{M}_2)$ such that for all databases D, $P\big[\mathfrak{M}^*_{\mathfrak{M}_1,\mathfrak{M}_2}(D) = (\omega_1,\omega_2)\big] = P[\mathfrak{M}_1(D) = \omega_1]P[\mathfrak{M}_2(D) = \omega_2]$. A privacy definition $\mathcal{P}$ with parameter ϵ* self-composes linearly *if for all ϵ_1, ϵ_2, and mechanisms $\mathfrak{M}_1$ and $\mathfrak{M}_2$ that satisfy $\mathcal{P}$ with parameters ϵ_1 and ϵ_2, respectively, $\mathfrak{M}^*_{\mathfrak{M}_1,\mathfrak{M}_2}$ satisfies $\mathcal{P}$ with parameter $\epsilon_1 + \epsilon_2$.*

Open Question 11. *What are necessary and sufficient conditions for a privacy definition to satisfy linear self-composability?*

Background We know a sufficient condition for when a general privacy definition from the Pufferfish framework [KM12] is linear self-composable. Namely, any privacy definition that compares the output distribution of a mechanism on some set $\mathcal{N}$ of pairs of neighboring datasets composes with itself. What is the full characterization of this class?

3.2 Characterization of general answerability

Contributed by: Daniel Li

Suppose that the privacy mechanism $\mathcal{K}$ is the Laplace mechanism. We use the interactive model and assume that the database does not remember the past queries. If one person asks one query Q for m times, and each time the variance of the answer is v. Then by averaging the answers of these m queries, she may refine the answer and get the answer with variance v/m.

Open Question 12. *Are there smarter methods rather than averaging, and what is the ultimate limit of these smarter methods? In other words, what is the least variance she can get by asking the same query m times?*

Background We know the answer for the case when the inference function is a linear function (averaging is an example). We call this linear answerability. And the answer is open for general inference functions (e.g. take a median), namely general answerability. Please refer to [LLMS12].

3.3 Blatant Non-Privacy Lower Bound for Releasing Non-differentiable Statistical Estimators

Contributed by: Shiva Kasiviswanathan

Background. Assume we have n samples $\mathbf{x}_1 \dots \mathbf{x}_n \in \mathbb{R}^{k+1}$, consider the following optimization problem:

$$\mathcal{L}(\theta; \mathbf{x}_1 \dots \mathbf{x}_n) = \frac{1}{n} \sum_{i=1}^{n} \ell(\theta; \mathbf{x}_i), \tag{1}$$

where $\theta \in \Theta \subset \mathbb{R}^{k+1}$, the *separable* loss function $\mathcal{L} : \Theta \times (\mathbb{R}^{k+1})^n \to \mathbb{R}$ measures the "fit" of $\theta \in \Theta$ to any given data $\mathbf{x}_1 \dots \mathbf{x}_n$, and $\ell : \Theta \times \mathbb{R}^{k+1} \to \mathbb{R}$ is the loss function associated with a single data point. The M-estimator $(\hat{\theta})$ associated with a given a function $\mathcal{L}(\theta; \mathbf{x}_1 \dots \mathbf{x}_n) \geq 0$ is

$$\hat{\theta} = \text{argmin}_{\theta \in \Theta} \, \mathcal{L}(\theta; \mathbf{x}_1 \dots \mathbf{x}_n) = \text{argmin}_{\theta \in \Theta} \, \frac{1}{n} \sum_{i=1}^{n} \ell(\theta; \mathbf{x}_i).$$

M-estimators are natural extensions of the Maximum Likelihood Estimators (MLE) and very commonly used in statistical studies. For a differentiable loss function ℓ, the estimator $\hat{\theta}$ could be found by setting $\partial \mathcal{L}(\theta; \mathbf{x}_1 \dots \mathbf{x}_n)$ to zero. For differentiable loss functions, Kasiviswanathan, Rudelson, and Smith [KRS13] showed that one needs to add at least $\Omega(1/\sqrt{n})$ noise while releasing these M-estimators to prevent a blatant non-privacy attack (this is an informal statement, for a formal statement see Theorem 4.2 of [KRS13]).

Question. The open question is to obtain blatant non-privacy lower bounds for non-differentiable loss functions. Note that non-differentiable ℓ's are very commonly use in practice, e.g., ℓ could be the L_1-loss function or the hinge loss appearing in Support Vector Machines. Since avoiding

blatant privacy in itself is a very weak privacy guarantee such lower bounds will "truly" reveal the cost of privacy.

4 Nontechnical Questions

4.1 Setting the privacy parameters

Contributed by: Arik Friedman, Andreas Haeberlen, Benjamin C. Pierce

Open Question 13. *What are reasonable values for ϵ (and δ) in differential privacy?*

Background Much of the research on differential privacy has focused on exploring theoretical upper and lower bounds on the relation between privacy and utility for different applications, usually exploring the asymptotic behavior of the system. However, to bridge the gap between theory and practice, researchers who apply differential privacy investigate privacy and utility trade-offs under specific settings and particular datasets, and evaluate based on the obtained outcomes whether a certain algorithm provides an acceptable performance within the given privacy constraints. To this end, understanding what are reasonable values of the privacy parameters is key in evaluation of practical systems. While ϵ is usually thought of as a small (< 1) number [Dwo11], in some cases larger values may be acceptable. For example, Dwork et al. [CDS11] suggested that for datasets such as a search query logs, in some scenarios even values as high as $\epsilon = 12$ may be acceptable.

Many experimental studies of differentially private mechanisms indicated significant loss of utility for low values of ϵ, and demonstrated that for many applications and datasets, achieving reasonable utility requires ϵ values well beyond 1 [CM08, MKA$^+$08, KKMN09, MM10, FS10, BLST10, MKS11]. Chen et. al. [CRFG12] have even proposed to avoid setting a pre-determined privacy budget altogether, and track privacy deficit instead.

The motivation underlying this open problem is to generate guidelines for practitioners and for researchers who work on applied differential privacy. Such guidelines would allow to assess whether a given solution achieves a reasonable privacy/utility tradeoff or not. Additionally, it would be useful to look at some concrete scenarios, e.g., how to design a medical study for which differential privacy guarantees will be provided, and to determine how large a privacy budget would be required to publish the expected results.

Motivating examples To motivate the discussion on reasonable values for ϵ and δ, consider two other problem areas: determining a key length for a cryptographic system, and evaluating the strength of a password. Recommendations for minimum key length to use in a cryptographic cipher may rely on the best known attacks on that cipher, the technology available at the time, assumptions about the adversary's power, the sensitivity of the protected data, and safety margins that capture expected technological advancements and improvements in attack methodologies[1]. In password cracking, considering a brute-force attack is useful for getting an estimate of what makes a good or a bad password, and websites can use such benchmarks to guide users and give them feedback whether their chosen password is weak, ok or strong, even though a particular user's password may be easy to guess given the right auxiliary information. In both cases, the choice of the security parameter (key length or password strength) can be translated to measures of the

[1] See, e.g., `http://www.keylength.com/`

expected effort the adversary would need to invest (in time or money), allowing decision makers to make security choices without needing to understand the underlying technology. Moreover, the resulting best practices change over time, to reflect technological advancements and knowledge of new attacks.

Similarly, while actual privacy guarantees depend on the data, the adversary's goal and auxiliary information, it may still be useful to consider attack models and typical scenarios that would allow to reason about privacy protection and the effort the adversary would need to invest to acquire certain data despite differential privacy protections.

Possible directions A possible way to address this problem is by considering several scenarios and attacks, and assessing what would be reasonable values for the privacy parameters in these contexts, or translate ϵ and δ to cost (e.g., in repeated queries – time or money) that reflects the effort that the adversary needs to invest to obtain its goals under the given privacy settings. For example, it is possible to discuss desirable privacy requirements under specific scenarios such as collaborative security [RAW+10], analysis of commuting patterns [MKA+08], of search logs [KKMN09], or network trace analysis [MM10], and assess the privacy guarantees under different kinds of attacks, such as database reconstruction attacks [DN03b], compositions attacks [GKS08], deFinetti attack [Kif09], classifier-based attacks [Cor11], and temporal inference attacks [CKN+11].

4.2 Defining Privacy for Social Networks

Contributed by: Ashwin Machanavajjhala

Open Question 14. *How can we formalize a privacy definition for social networks? Can correlations in a social network be mathematically described (in order to define a set of possible adversarial prior distributions)? Which correlations in a social network should be considered sensitive information, and which ones should we allow an adversary to learn about?*

Background Recent work [KM11, KM12] has shown that differential privacy does limit the ability of an adversary to accurately learn sensitive information about individuals when the data are correlated (especially in social networks). Privacy can be formally defined in the presence of certain kinds of correlations, namely those induced by publicly known constraints in the data [KM12]. Can this be done for social networks?

References

[ACCL07] Nir Ailon, Bernard Chazelle, Seshadhri Comandur, and Ding Liu. Estimating the distance to a monotone function. *Random Struct. Algorithms*, 31(3):371–383, 2007.

[AJMR12] Pranjal Awasthi, Madhav Jha, Marco Molinaro, and Sofya Raskhodnikova. Testing lipschitz functions on hypergrid domains. In Anupam Gupta, Klaus Jansen, José D. P. Rolim, and Rocco A. Servedio, editors, *APPROX-RANDOM*, volume 7408 of *Lecture Notes in Computer Science*, pages 387–398. Springer, 2012.

[AN06] N. Alon and A. Naor. Approximating the cut-norm via grothendieck's inequality. *SIAM Journal on Computing*, 35(4):787–803, 2006.

[BBDS12] Jeremiah Blocki, Avrim Blum, Anupam Datta, and Or Sheffet. The Johnson-Lindenstrauss transform itself preserves differential privacy. *STOC*, 2012.

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

[BDMN05] A. Blum, C. Dwork, F. McSherry, and K. Nissim. Practical privacy: the sulq framework. In *Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 128–138. ACM, 2005.

[BGJ+12] Arnab Bhattacharyya, Elena Grigorescu, Madhav Jha, Kyomin Jung, Sofya Raskhodnikova, and David P. Woodruff. Lower bounds for local monotonicity reconstruction from transitive-closure spanners. *SIAM Journal on Discrete Mathematics*, 26(2):618–646, 2012.

[BLR08] A. Blum, K. Ligett, and A. Roth. A learning theory approach to non-interactive database privacy. In *Proceedings of the 40th annual ACM Symposium on Theory of Computing*, pages 609–618. ACM, 2008.

[BLST10] Raghav Bhaskar, Srivatsan Laxman, Adam Smith, and Abhradeep Thakurta. Discovering frequent patterns in sensitive data. In *KDD*, pages 503–512, 2010.

[CDS11] Kobbi Nissim Cynthia Dwork, Frank McSherry and Adam Smith. Differential privacy - a primer for the perplexed. In *Conference of European Statisticians, Joint UNECE/Eurostat work session on statistical data confidentiality*, Tarragona, Spain, October 2011. `http://www.unece.org/fileadmin/DAM/stats/documents/ece/ces/ge.46/2011/26_Dwork-Smith.pdf`.

[CKN+11] Joseph A. Calandrino, Ann Kilzer, Arvind Narayanan, Edward W. Felten, and Vitaly Shmatikov. "you might also like: " privacy risks of collaborative filtering. In *IEEE Symposium on Security and Privacy*, pages 231–246, 2011.

[CM08] Kamalika Chaudhuri and Claire Monteleoni. Privacy-preserving logistic regression. In *NIPS*, pages 289–296, 2008.

[Cor11] Graham Cormode. Personal privacy vs population privacy: learning to attack anonymization. In *KDD*, pages 1253–1261, 2011.

[CRFG12] Ruichuan Chen, Alexey Reznichenko, Paul Francis, and Johannes Gehrke. Towards statistical queries over distributed private user data. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, NSDI'12, pages 13–13, Berkeley, CA, USA, 2012. USENIX Association.

[CSS12] Kamalika Chaudhuri, Anand Sarwate, and Kaushik Sinha. Near-optimal differentially private principal components. In *Proc. 26th Annual Conference on Neural Information Processing Systems (NIPS)*, 2012.

[DMNS06] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the 3rd Conference on Theory of Cryptography*, pages 265–284. Springer, 2006.

[DN03a] I. Dinur and K. Nissim. Revealing information while preserving privacy. In *Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 202–210. ACM, 2003.

[DN03b] Irit Dinur and Kobbi Nissim. Revealing information while preserving privacy. In *PODS*, pages 202–210, 2003.

[DNR+09] C. Dwork, M. Naor, O. Reingold, G.N. Rothblum, and S. Vadhan. On the complexity of differentially private data release: efficient algorithms and hardness results. In *Proceedings of the 41st annual ACM Symposium on Theory of Computing*, pages 381–390. ACM, 2009.

[DRV10] C. Dwork, G.N. Rothblum, and S. Vadhan. Boosting and differential privacy. In *Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science*, pages 51–60. IEEE, 2010.

[Dwo11] Cynthia Dwork. A firm foundation for private data analysis. *Commun. ACM*, 54(1):86–95, 2011.

[FS10] Arik Friedman and Assaf Schuster. Data mining with differential privacy. In *KDD*, pages 493–502, 2010.

[GHRU11] A. Gupta, M. Hardt, A. Roth, and J. Ullman. Privately releasing conjunctions and the statistical query barrier. In *Proceedings of the 43rd annual ACM Symposium on Theory of Computing*, pages 803–812. ACM, 2011.

[GKS08] Srivatsava Ranjit Ganta, Shiva Prasad Kasiviswanathan, and Adam Smith. Composition attacks and auxiliary information in data privacy. In *KDD*, pages 265–273, 2008.

[GRU12a] A. Gupta, A. Roth, and J. Ullman. Iterative constructions and private data release. In *Proceedings of the 9th Conference on Theory of Cryptography*, pages 339–356. Springer, 2012.

[GRU12b] A. Gupta, A. Roth, and J. Ullman. Iterative constructions and private data release. In *TCC*, pages 339–356, 2012.

[Har11] M. Hardt. *A Study of Privacy and Fairness in Sensitive Data Analysis*. PhD thesis, Princeton University, 2011.

[HLM12a] M. Hardt, K. Ligett, and F. McSherry. A simple and practical algorithm for differentially private data release. *NIPS*, 2012.

[HLM12b] Moritz Hardt, Katrina Ligett, and Frank McSherry. A simple and practical algorithm for differentially private data release. In *NIPS*, 2012. To appear.

[HR10a] M. Hardt and G. Rothblum. A multiplicative weights mechanism for privacy-preserving data analysis. *Proc. 51st Foundations of Computer Science (FOCS). IEEE*, 2010.

[HR10b] M. Hardt and G.N. Rothblum. A multiplicative weights mechanism for privacy-preserving data analysis. In *Proceedings of the 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 61–70. IEEE Computer Society, 2010.

[HR12a] M. Hardt and A. Roth. Beyond Worst-Case Analysis in Private Singular Vector Computation. *arXiv preprint arXiv:1211.0975*, 2012.

[HR12b] Moritz Hardt and Aaron Roth. Beating randomized response on incoherent matrices. In *Proc. 44th Symposium on Theory of Computing (STOC)*, pages 1255–1268. ACM, 2012.

[HR12c] Moritz Hardt and Aaron Roth. Beyond worst-case analysis in private singular vector computation. *CoRR*, abs/1211.0975, 2012.

[HR12d] Z. Huang and A. Roth. Exploiting metric structure for efficient private query release. *Manuscript*, 2012.

[HRS11] Moritz Hardt, Guy N. Rothblum, and Rocco A. Servedio. Private data release via learning thresholds. *CoRR*, abs/1107.2444, 2011.

[HT10] Moritz Hardt and Kunal Talwar. On the geometry of differential privacy. In *Proceedings of the 42nd ACM symposium on Theory of computing*, STOC '10, pages 705–714, New York, NY, USA, 2010. ACM.

[JR11] Madhav Jha and Sofya Raskhodnikova. Testing and reconstruction of lipschitz functions with applications to data privacy. In Rafail Ostrovsky, editor, *FOCS*, pages 433–442. IEEE, 2011.

[Kif09] Daniel Kifer. Attacks on privacy and definetti's theorem. In *SIGMOD Conference*, pages 127–138, 2009.

[KKMN09] Aleksandra Korolova, Krishnaram Kenthapadi, Nina Mishra, and Alexandros Ntoulas. Releasing search queries and clicks privately. In *WWW*, pages 171–180, 2009.

[KM11] Daniel Kifer and Ashwin Machanavajjhala. No free lunch in data privacy. In *SIGMOD*, pages 193–204, 2011.

[KM12] Daniel Kifer and Ashwin Machanavajjhala. A rigorous and customizable framework for privacy. In *PODS*, 2012.

[KRS13] Shiva Prasad Kasiviswanathan, Mark Rudelson, and Adam Smith. The power of linear reconstruction attacks. In *Symposium on Discrete Algorithms (SODA)*, 2013.

[KRSU10] Shiva Prasad Kasiviswanathan, Mark Rudelson, Adam Smith, and Jonathan Ullman. The price of privately releasing contingency tables and the spectra of random matrices with correlated rows. In *STOC*, pages 775–784, 2010.

[KT13] Michael Kapralov and Kunal Talwar. On differentially private low rank approximation. In *Proc. 24rd Symposium on Discrete Algorithms (SODA)*. ACM-SIAM, 2013.

[LHR+10] Chao Li, Michael Hay, Vibhor Rastogi, Gerome Miklau, and Andrew McGregor. Optimizing linear counting queries under differential privacy. In Jan Paredaens and Dirk Van Gucht, editors, *PODS*, pages 123–134. ACM, 2010.

[LLMS12] Chao Li, Daniel Yang Li, Gerome Miklau, and Dan Suciu. A theory of pricing private data. *CoRR*, abs/1208.5258, 2012.

[LM12] Chao Li and Gerome Miklau. An adaptive mechanism for accurate query answering under differential privacy. *PVLDB*, 5(6):514–525, 2012.

[MKA+08] Ashwin Machanavajjhala, Daniel Kifer, John M. Abowd, Johannes Gehrke, and Lars Vilhuber. Privacy: Theory meets practice on the map. In *ICDE*, pages 277–286, 2008.

[MKS11] Ashwin Machanavajjhala, Aleksandra Korolova, and Atish Das Sarma. Personalized social recommendations - accurate or private? *PVLDB*, 4(7):440–450, 2011.

[MM10] Frank McSherry and Ratul Mahajan. Differentially-private network trace analysis. In *SIGCOMM*, pages 123–134, 2010.

[NTZ12] A. Nikolov, K. Talwar, and L. Zhang. The geometry of differential privacy: the approximate and sparse cases. unpublished manuscript, 2012.

[RAW+10] Jason Reed, Adam J. Aviv, Daniel Wagner, Andreas Haeberlen, Benjamin C. Pierce, and Jonathan M. Smith. Differential privacy for collaborative security. In *EUROSEC*, pages 1–7, 2010.

[RR10] A. Roth and T. Roughgarden. Interactive privacy via the median mechanism. In *Proceedings of the 42nd ACM Symposium on Theory of Computing*, pages 765–774. ACM, 2010.

[SS10] Michael E. Saks and C. Seshadhri. Local monotonicity reconstruction. *SIAM J. Comput.*, 39(7):2897–2926, 2010.

[TUV12] Justin Thaler, Jonathan Ullman, and Salil P. Vadhan. Faster algorithms for privately releasing marginals. *CoRR*, abs/1205.1758, 2012. Appeared in *ICALP* 2012.

[Ull12] J. Ullman. Answering $n^{2+o(1)}$ counting queries with differential privacy is hard. *arXiv preprint arXiv:1207.6945*, 2012.

[UV11] Jonathan Ullman and Salil P. Vadhan. Pcps and the hardness of generating private synthetic data. In Yuval Ishai, editor, *TCC*, volume 6597 of *Lecture Notes in Computer Science*, pages 400–416. Springer, 2011.

[YZW+12] Ganzhao Yuan, Zhenjie Zhang, Marianne Winslett, Xiaokui Xiao, Yin Yang, and Zhifeng Hao. Low-rank mechanism: Optimizing batch queries under differential privacy. *PVLDB*, 5(11):1352–1363, 2012.