Universally Utility-Maximizing Privacy Mechanisms*

Arpita Ghosh† Tim Roughgarden‡ Mukund Sundararajan§

August 15, 2009

Abstract

A mechanism for releasing information about a statistical database with sensitive data must resolve a trade-off between utility and privacy. Publishing fully accurate information maximizes utility while minimizing privacy, while publishing random noise accomplishes the opposite. Privacy can be rigorously quantified using the framework of *differential privacy*, which requires that a mechanism's output distribution is nearly the same whether or not a given database row is included or excluded. The goal of this paper is strong and general utility guarantees, subject to differential privacy.

We pursue mechanisms that guarantee near-optimal utility to every potential user, independent of its side information (modeled as a prior distribution over query results) and preferences (modeled via a loss function). Our main result is: for each fixed count query and differential privacy level, there is a *geometric mechanism* M^* — a discrete variant of the simple and well-studied mechanism that adds random noise from a Laplace distribution — that is *simultaneously expected loss-minimizing* for every possible user, subject to the differential privacy constraint. This is an extremely strong utility guarantee: *every* potential user u, no matter what its side information and preferences, derives as much utility from M^* as from interacting with a differentially private mechanism M_u that is optimally tailored to u. More precisely, for every user u there is an optimal mechanism M_u for it that factors into a user-independent part (the geometric mechanism M^*) and a user-specific post-processing step that depends only on the output of the geometric mechanism and not on the underlying database.

The first part of our proof of this result characterizes the optimal differentially private mechanism for a user as a certain basic feasible solution to a linear program with a user-specific objective function and user-independent constraints that encode differential privacy. The second part shows that all of the relevant vertices of the feasible region (ranging over all possible users) are derivable from the geometric mechanism via suitable remappings of its range.

1 Introduction

Organizations including the census bureau, medical establishments, and Internet companies collect and publish statistical information [6, 19]. The census bureau may, for instance, publish the result

*A preliminary version of this paper appeared in the Proceedings of the 40th Annual Symposium on Theory of Computing, June 2008.

†Yahoo! Research, 2821 Mission College Boulevard, Santa Clara, CA. Email: arpita@yahoo-inc.com.

‡Department of Computer Science, Stanford University, 462 Gates Building, 353 Serra Mall, Stanford, CA 94305. Supported in part by NSF CAREER Award CCF-0448664, an ONR Young Investigator Award, an AFOSR MURI grant, and an Alfred P. Sloan Fellowship. Email: tim@cs.stanford.edu.

§Google, Inc., Mountain View, CA. This work was done at Yahoo! Research and Stanford University, and was supported in part by NSF award CCF-0448664 and a Stanford Graduate Fellowship. Email: mukunds@google.com

of a query such as: "How many individuals have incomes that exceed $100,000?". An implicit hope in this approach is that aggregate information is sufficiently anonymous so as not to breach the privacy of any individual. Unfortunately, publication schemes initially thought to be "private" have succumbed to privacy attacks [1, 17, 19], highlighting the urgent need for mechanisms that are *provably* private. The differential privacy literature [4, 5, 7, 8, 10, 12, 16, 18] has proposed a rigorous and quantifiable definition of privacy, as well as provably privacy-preserving mechanisms for diverse applications including statistical queries, machine learning, and pricing. Essentially, for a parameter $\alpha \in [0, 1]$, a randomized mechanism is *α-differentially private* if changing a row of the underlying database—the data of a single individual—changes the probability of each mechanism output by at most an α factor. Larger values of α correspond to greater levels of privacy. Differential privacy is typically achieved by adding "noise" to a query result that scales with α.

It is trivial to achieve any level of differential privacy, for instance by always returning (data-independent) random noise. This "solution" obviously completely defeats the original purpose of providing useful information. On the other hand, returning fully accurate results yields privacy violations [8]. *The goal of this paper is to identify, for each $\alpha \in [0, 1]$, the optimal (i.e., utility-maximizing) α-differentially private mechanism.*

2 The Model: Privacy, Utility, and Rational Users

2.1 Differential Privacy

We consider databases with n rows drawn from a finite set D. Every row corresponds to an individual. Two databases are *neighbors* if they coincide in $n - 1$ rows. A *count query* f takes a database $d \in D^n$ as input and returns the result $f(d)$ that is the number of rows that satisfy a fixed, non-trivial predicate on D. Such queries are also called predicate or subset-sum queries; they have been extensively studied in their own right [4, 5, 7, 12], and form a basic primitive from which more complex queries can be constructed [4].

A mechanism with a range R is a probabilistic function from D^n to R. Typical ranges include the real numbers, the integers, and the set $N = \{0, 1, 2, \ldots, n\}$ of the possible true answers to a count query. For a mechanism X with a countable range, we use x_{dr} to denote the probability that the mechanism outputs the response $r \in R$ when the underlying database is $d \in D^n$. For such a mechanism X and a parameter $\alpha \in [0, 1]$, the mechanism is *α-differentially private* if and only if the ratio $x_{d_1 r}/x_{d_2 r}$ lies in the interval $[\alpha, 1/\alpha]$ for every possible output $r \in R$ and pair d_1, d_2 of neighboring databases.[1] (We interpret 0/0 as 1.) Intuitively, the probability of every response of the privacy mechanism — and hence the probability of a successful privacy attack following an interaction with the mechanism — is, up to a controllable α factor, independent of whether a given user "opts in" or "opts out" of the database [10, 14].

For a given query f, a mechanism is *oblivious* if, for all $r \in R$, $x_{d_1 r} = x_{d_2 r}$ whenever $f(d_1) = f(d_2)$ — that is, if the output distribution depends only on the query result. Most of this paper considers only oblivious mechanisms; this assumption is natural and, for optimal privacy mechanism design, it is without loss of generality in a precise sense (see Section 6). The notation and definitions above simplify for oblivious mechanisms and count queries. We can specify an oblivious mechanism with a countable range via the probabilities x_{ir} of outputting a response $r \in R$ for each query result

[1] For general measurable ranges, differential privacy requires these bounds on the corresponding ratio of every subset of the range. When the range is countable, as in most of this paper, the two definitions are equivalent.

$i \in N$; α-differential privacy is then equivalent to the constraint that the ratios $x_{ir}/x_{(i+1)r}$ lie in the interval $[\alpha, 1/\alpha]$ for every possible output $r \in R$ and query result $i \in N \setminus \{n\}$.

Example 2.1 (Geometric Mechanism) For a count query f and parameter value $\alpha \in (0,1)$, the *α-geometric mechanism* is an oblivious mechanism with range $\mathbb{Z}$, defined as follows. When the true query result is $f(d)$, the mechanism outputs $f(d) + \Delta$, where Δ is a random variable with a two-sided geometric distribution:

$$\mathbf{Pr}[\Delta = \delta] = \frac{1-\alpha}{1+\alpha}\alpha^{|\delta|} \tag{1}$$

for every integer δ. See also Figure 2.1(a). For convenience, we also define the 0-geometric mechanism as that which always returns the true query result $f(d)$, and the 1-geometric mechanism as that which always results the answer 0 (say), independent of the input database.

The α-geometric mechanism is α-differentially private because the result of a count query differs by at most one on neighboring databases, and because, for each δ, the probabilities $\mathbf{Pr}[\Delta = \delta + 1]$ and $\mathbf{Pr}[\Delta = \delta - 1]$ lie between $\alpha\mathbf{Pr}[\Delta = \delta]$ and $\mathbf{Pr}[\Delta = \delta]/\alpha$. This mechanism is a discretized version of a well-known mechanism that adds random noise from a Laplace distribution (with density $\epsilon/2 \cdot e^{-\epsilon|t|}$ on $\mathbb{R}$, where $\epsilon = \ln \frac{1}{\alpha}$); see e.g. [10].

Example 2.2 (Truncated Geometric Mechanism) The α-geometric mechanism outputs an "obviously wrong" output $f(d)+\Delta$ — one less than 0 or greater than n – with non-zero probability. The *truncated α-geometric mechanism* has range $N = \{0, 1, \ldots, n\}$ and addresses this drawback in the obvious way, by "remapping" all negative outputs to 0 and all outputs greater than n to n. In other words, the mechanism uses the following distribution of noise Δ when the query result is $f(d)$: $\mathbf{Pr}[\Delta < -f(d)] = \mathbf{Pr}[\Delta > n - f(d)] = 0$; $\mathbf{Pr}[\Delta = -f(d)] = \alpha^{f(d)}/(1+\alpha)$; $\mathbf{Pr}[\Delta = n - f(d)] = \alpha^{n-f(d)}/(1+\alpha)$; and all other probabilities are as in Example 2.1. See also Figure 2.1(b). This mechanism is again α-differentially private.

2.2 Utility Model

A key contribution of this paper is a utility model that enables strong and general utility guarantees for privacy mechanisms. Just as differential privacy guarantees protection against every potential attacker, independent of its side information, we seek mechanisms that guarantee optimal utility to *every* potential user, independent of its side information and preferences.

We now formally define preferences and side information. We model the preferences of a user via a *loss function* l defined on $N \times N$, where $l(i,j)$ denotes the user's loss when the true query result is i and the user believes it to be j. We would obviously prefer to assume as little as possible about a user's preferences, so we permit quite general loss functions, requiring only one symmetry and one monotonicity property. Precisely, we call a loss function *legal* if the loss $l(i,j)$ depends only on i and $|j-i|$, and if the loss is nondecreasing in $|j-i|$ for each fixed $i \in N$. For example, the loss function $l(i,j) = |j-i|$ measures mean error, the implicit measure of (dis)utility in most previous literature on differential privacy. Two among the many other natural possibilities are the squared error $(j-i)^2$ and the binary loss function $l_{bin}(i,j)$, defined as 0 if $i = j$ and 1 otherwise. Most natural loss functions depend only on $|j-i|$ and not directly on i, but we do not require this additional property for our results.

We model the side information of a user as a prior probability distribution p over the query results $i \in N$. This prior represents the beliefs of the user, which might stem from other information

Input/Output	...	-1	0	1	2	3	4	5	6	...
0	...	1/6	1/3	1/6	1/12	1/24	1/48	1/96	1/192	...
1	...	1/12	1/6	1/3	1/6	1/12	1/24	1/48	1/96	...
2	...	1/24	1/12	1/6	1/3	1/6	1/12	1/24	1/48	...
3	...	1/48	1/24	1/12	1/6	1/3	1/6	1/12	1/24	...
4	...	1/96	1/48	1/24	1/12	1/6	1/3	1/6	1/12	...
5	...	1/192	1/96	1/48	1/24	1/12	1/6	1/3	1/6	...

(a) $\frac{1}{2}$-geometric mechanism with $n = 5$

Input/Output	0	1	2	3	4	5
0	2/3	1/6	1/12	1/24	1/48	1/48
1	1/3	1/3	1/6	1/12	1/24	1/24
2	1/6	1/6	1/3	1/6	1/12	1/12
3	1/12	1/12	1/6	1/3	1/6	1/6
4	1/24	1/24	1/12	1/6	1/3	1/3
5	1/48	1/48	1/24	1/12	1/6	2/3

(b) Truncated $\frac{1}{2}$-geometric mechanism with $n = 5$

Figure 1: The defining probabilities of the geometric and truncated geometric mechanisms, for $\alpha = \frac{1}{2}$ and $n = 5$. Rows correspond to query results $i \in N$, and columns to mechanism outputs $r \in R$.

sources, previous interactions with the mechanism, introspection, or common sense. We emphasize that we are *not* introducing priors to weaken the definition of differential privacy; we use the standard definition of differential privacy (which makes no assumptions about the side information of an attacker) and use a prior only to discuss the *utility* of a (differentially private) mechanism to a potential user.

Consider a user with a prior p and loss function l that interacts with an oblivious mechanism X with range R. We assume that the implementation of X is publicly known. Since the range R of X need not coincide with the set N of legitimate query results, a user generally must reinterpret an output $r \in R$ of the mechanism as some query result $j \in N$. For example, a user that observes the output "-2" from the α-geometric mechanism (Example 2.1) might guess that the actual query result is most likely to be 0.

Our utility model thus motivates the concept of a *remap* of a mechanism X with range R, which is a probabilistic function Y from R to N, with y_{rj} denoting the probability that a user reinterprets the mechanism's response $r \in R$ as the query result $j \in N$. A mechanism X and a remap Y together induce a new mechanism $Z = Y \circ X$ with $z_{ij} = (Y \circ X)_{ij} = \sum_{r \in R} x_{ir} \cdot y_{rj}$. For example, the truncated α-geometric mechanism (Example 2.2) can be written as $Y \circ X$, where X is the (untruncated) α-geometric mechanism; and $y_{rj} = 1$ whenever $r \leq j = 0$, $r \geq j = n$, or $r = j \in \{1, 2, \ldots, n-1\}$; and $y_{rj} = 0$ otherwise. When the range R is finite, a mechanism X and remap Y naturally correspond to matrices, and composition $Y \circ X$ is simply matrix multiplication. The next section details our assumptions about the remaps employed by a "rational" user, which arises from expected loss minimization following a Bayesian update.

We can now define the expected loss of a user u with respect to a mechanism X and a remap Y. For a given input d with query result $i = f(d)$, let $z_{ij} = (Y \circ X)_{ij}$ denote the probability that the user reinterprets the mechanism's (random) output as the query result j. The user's expected loss

for an input d with $f(d) = i$ is

$$\sum_{j \in N} z_{ij} \cdot l(i,j),$$

where the expectation is over the coin flips internal to the mechanism and the remap. The user's prior provides a way to aggregate expected losses for different inputs, thereby yielding a measure of the overall (dis)utility to the user under mechanism X with remap Y:

$$\sum_{i \in N} p_i \sum_{j \in N} z_{ij} \cdot l(i,j). \tag{2}$$

The quantity in (2) is simply the expected loss over the coin tosses of the mechanism X, the remap Y, and the prior p.[2]

2.3 User Post-Processing and Optimal Privacy Mechanisms

Simultaneous optimality of a single mechanism for every possible user would be an extremely strong guarantee. To achieve it, it is necessary (and, as we show, sufficient) to delegate post-processing work to a user in the form a suitable remap. To motivate this, consider a user with prior p and loss function l that interacts with a mechanism X. We have already seen the need for remaps (from R to N) when the mechanism's range does not correspond directly to legitimate query results. The next example shows that, even for a mechanism with range $R = N$, an expected loss-minimizing user might be motivated to reinterpret the mechanism's outputs.

Example 2.3 (Post-Processing Decreases Expected Loss) Fix a database size n that is odd. Consider a user with the binary loss function l_{bin}, and prior $p_0 = p_n = 1/2$ and $p_j = 0$ for $j \in \{1, 2, \ldots, n-1\}$. Suppose this user interacts with the α-geometric mechanism X (Example 2.1). If the user accepts the mechanism's outputs "at face value", in the sense that it uses the identity remap, then its expected loss (2) is $\mathbf{Pr}[\Delta \neq 0] = 2\alpha/(1+\alpha)$, where Δ denotes the random noise in the geometric mechanism (1). If the user instead remaps outputs of the geometric mechanism that are at least $(n+1)/2$ to n and all other outputs to 0 — reflecting its certainty that the true query result must be 0 or n — it effectively induces a new mechanism with the much smaller expected loss of $\mathbf{Pr}[\Delta \geq (n+1)/2] = \alpha^{(n+1)/2}/(1+\alpha)$.

We assume that a (rational) user with prior p and loss function l, interacting with a publicly known mechanism X, employs a remap Y that induces the mechanism $Z = Y \circ X$ that minimizes the user's expected loss (2) over all such remaps. It is well known (e.g. [15, Chapter 9]) and easy to prove that, among all possible (randomized) remappings, the optimal one follows from applying Bayes rule and then minimizing expected loss. For example, given an output r of a mechanism X with a countable range, one computes the induced posterior distribution q over query results: for each $i \in N$, $q_i = p_i \cdot x_{ir} / (\sum_{i' \in N} p_{i'} \cdot x_{i'r})$. Then, for a query result $j^* \in N$ that minimizes expected loss (over all $j \in N$) with respect to this posterior q, one sets $y_{rj^*} = 1$ and $y_{rj} = 0$ for $j \neq j^*$. This remap Y is deterministic and simple to compute.

When we speak of the expected loss of a user u with respect to a mechanism X, we assume that the user employs an optimal remap Y of X for u. We can then define an *optimal* α-differentially

[2]The central theorem of choice theory (e.g. [15, Chapter 6]) states that every preference relation over mechanisms that satisfies reasonable axioms (encoding "rationality") can be modeled via expected utility, just as we propose. This theorem justifies the use of priors for expressing a rational user's trade-off over possible inputs.

private oblivious mechanism for a user as one that minimizes the user's expected loss (2) over all such mechanisms.

3 Main Result and Discussion

Our main results concern mechanisms that are simultaneously optimal for all users in the following sense.

Definition 3.1 (Universally Utility-Maximizing Mechanism) Fix arbitrary values for $n \geq 1$ and $\alpha \in [0, 1]$, and a count query. An oblivious α-differentially private mechanism X is *universally utility-maximizing* if and only if: for every user u with a prior over $N = \{0, 1, 2, \ldots, n\}$ and a legal loss function on $N \times N$, the mechanism X is optimal for u.

A mechanism that satisfies Definition 3.1 — assuming, for the moment, that one exists — provides an extremely strong utility-maximization guarantee. *Every* potential user u, no matter what its side information and preferences, derives as much utility from such a mechanism as it does from interacting with a differentially private mechanism that is optimally tailored to u. We reiterate that the prior from the utility model plays no role in the definition of privacy, which is the standard, worst-case (over adversaries with arbitrary side information and intent) guarantee provided by differential privacy.

Our main result is a characterization of the universally utility-maximizing mechanisms.

Theorem 3.2 (Main Characterization) *Fix arbitrary values for $n \geq 1$ and $\alpha \in [0, 1]$, and a count query. A mechanism X is universally utility-maximizing if and only if there is a remap Y of X such that $Y \circ X$ is the truncated α-geometric mechanism.*

Of course, the most significant implication of Theorem 3.2 is that universally utility-maximizing mechanisms exist. Recall from Section 2.3 that the truncated α-geometric mechanism is a remap of the α-geometric mechanism (and also a trivial remap of itself).

Corollary 3.3 (Implications for Geometric Mechanisms) *For every $n \geq 1$, $\alpha \in [0, 1]$, and every count query, the corresponding α-geometric and truncated α-geometric mechanisms are universally utility-maximizing.*

Our arguments also imply that the truncated α-geometric mechanism is the *unique* universally utility-maximizing mechanism with range N, up to renaming.

Corollary 3.4 (Uniqueness of the Truncated Geometric Mechanism) *Fix arbitrary values for $n \geq 1$ and $\alpha \in [0, 1]$, and a count query. A mechanism X with range N is universally utility-maximizing if and only if it has the form $\pi \circ T$, where π is a permutation of N and T is the truncated α-geometric mechanism.*

A universally utility-maximizing mechanism with range larger than N need not have the form $\pi \circ T$, as shown by the α-geometric mechanism. We prove Theorem 3.2 and Corollary 3.4 in Section 5.

We emphasize that while the geometric mechanism is user-independent — all users see the same distribution over responses — different users remap its responses in different ways, as informed by their individual prior distributions and loss functions. Rephrasing Corollary 3.3, for every user

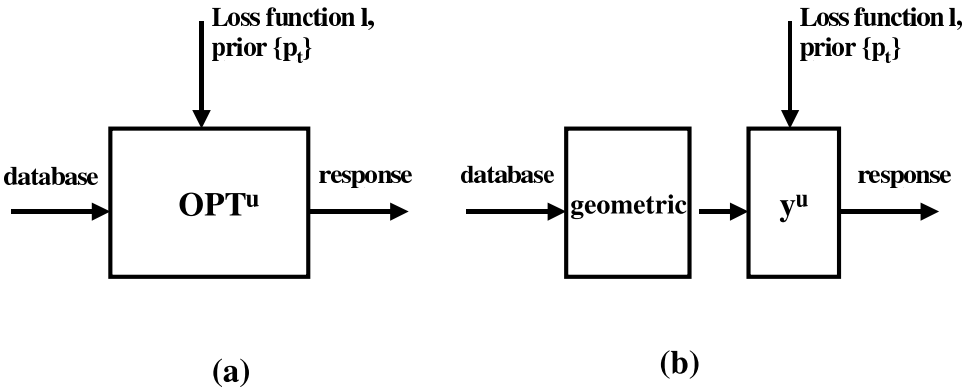

Figure 2: Theorem 3.2. For every rational user u, the utility-maximizing mechanism for u (shown in (a)) can be factored into a user-independent part (the α-geometric mechanism) followed by a user-dependent post-processing step (the optimal remap Y^u).

there is an optimal mechanism for it that factors into a user-independent part — the α-geometric mechanism — and a user-specific post-processing step that depends only on the output of the geometric mechanism *and not on the underlying database or query result*. See Figure 2 for an illustration of this interpretation of Theorem 3.2 and Corollary 3.3.

Our results thus show how to achieve the same utility as a user-specific optimal mechanism without directly implementing one. Direct user-specific optimization would clearly involve several challenges. First, it would require advance knowledge or elicitation of user preferences, which we expect is impractical in most applications. And even if a mechanism was privy to the various preferences of its users, it would effectively need to answer the same query in different ways for different users, which in turn degrades its differential privacy guarantee (to α^a, where a is the number of different answers).

Finally, from a design perspective, our results strongly advocate using a random perturbation drawn from a two-sided geometric distribution as *the* way to implement a differentially private count query. By contrast, adding random noise from a Laplace distribution, as done in several previous works, does not yield a universally utility-maximizing mechanism.[3]

4 Related Work

Differential privacy is motivated in part by the provable impossibility of absolute privacy against attackers with arbitrary side information [8]. One interpretation of differential privacy is: no matter what prior distribution over databases a potential attacker has, its posterior after interacting with a differentially private mechanism is almost independent of whether a given user "opted in" or "opted out" of the database [10, 14]. Below we discuss the papers in the differential privacy literature closest to the present work; see [9] for a recent survey of the state of the field.

Dinur and Nissim [7] showed that for a database with n rows, answering $O(n \log^2 n)$ randomly chosen subset count queries with $o(\sqrt{n})$ error allows an adversary to reconstruct most of the rows

[3]For example, take $\alpha = 1/2$ and $n = 1$. Consider a user with uniform prior on $\{0, 1\}$ and the binary loss function (see Section 2.2). The truncated α-geometric mechanism leads to an expected loss of $1/3$ for this user. Using random noise from a Laplace distribution with density $\epsilon/2 \cdot e^{-\epsilon|t|}$, where $\epsilon = \ln \frac{1}{\alpha}$, leads to an expected loss of $1/2\sqrt{2} \approx .35$.

of the database (a blatant privacy breach); see Dwork et al. [11] for a more robust impossibility result of the same type. Most of the differential privacy literature circumvents these impossibility results by focusing on interactive models where a mechanism supplies answers to only a sub-linear (in n) number of queries. Count queries (e.g. [7, 12]) and more general queries (e.g. [10, 18]) have been studied from this perspective.

Blum et al. [5] take a different approach by restricting attention to count queries that lie in a restricted class; they obtain non-interactive mechanisms that provide simultaneous good accuracy (in terms of worst-case error) for all count queries from a class with small VC dimension. Kasiviswanathan et al. [13] give further results for privately learning hypotheses from a given class.

The use of abstract "utility functions" in McSherry and Talwar [16] has a similar flavor to our use of loss functions, though the motivations and goals of their work and ours are unrelated. Motivated by pricing problems, McSherry and Talwar [16] design differentially private mechanisms for queries that can have very different values on neighboring databases (unlike count queries); they do not consider users with side information (i.e., priors) and do not formulate a notion of mechanism optimality (simultaneous or otherwise).

Finally, in recent and independent work, McSherry and Talwar (personal communication, October 2008) apply linear programming theory in the analysis of privacy mechanisms (as we do here). Again, their goal is different: they do not consider a general utility model, but instead ask how expected error must scale with the number of queries answered by a differentially private mechanism.

5 Proof of Theorem 3.2

5.1 Overview and Interpretations

This section proves Theorem 3.2. The "only if" direction is relatively straightforward and we prove it in Section 5.7. The proof of the "if" direction has three high-level steps.

1. For a given user u, we formulate the problem of minimizing expected loss over the set of oblivious and differentially private mechanisms as a linear program (LP). The objective function of this LP is user-specific, but the feasible region is not.

2. We identify stringent necessary conditions met by every privacy mechanism that is optimal for some user and is also a vertex solution of the aforementioned linear program.

3. For every privacy mechanism X meeting these necessary conditions, we construct a remap Y of the truncated α-geometric mechanism T such that $Y \circ T = X$. Since for every user there is an optimal mechanism that is a vertex solution, and since users employ optimal remaps, it follows that T is optimal for every user.

From a geometric perspective, the second step identifies the subset of vertices of the feasible region that can arise as optimal privacy mechanisms (for some user), and the third step shows that all of these also arise as remaps of a single vertex (the truncated α-geometric mechanism).[4] From a

[4]Vertices that do not correspond to the truncated α-geometric mechanism can be optimal for certain users; see e.g. Figure 3. Also, some vertices of the feasible region do not arise as remaps of the truncated α-geometric mechanism (Appendix A). Fortunately, the corresponding mechanisms are not uniquely optimal for any user with a legal loss function.

linear algebraic perspective, our proof shows that, for every possible optimal mechanism X that is also a vertex of the LP, the matrix product XT^{-1} has an extremely simple form that corresponds to a remap.

We now fix, for the rest of this section, a value of $n \geq 1$, a value of $\alpha \in [0, 1]$, and a count query f over databases with n rows. Theorem 3.2 is true but trivial if α is 0 or 1, so we assume henceforth that $\alpha \in (0, 1)$.

5.2 Linear Programming Formulation of a User-Optimal Mechanism

This section implements the first step of the proof outline of Section 5.1. We begin with a definition.

Definition 5.1 (Direct Privacy Mechanism) A mechanism X with range N is *direct for user u* if the identity remap is an optimal remap of X for u.

In other words, direct mechanisms are those whose outputs the user u might as well as accept "at face value".

We aim to characterize the optimal direct mechanisms for a given user — those minimizing expected loss (2) — as the optimal solutions to a linear program. We first show that restricting attention to direct mechanisms is without loss of generality for optimal privacy mechanism design.

Lemma 5.2 (Direct Mechanisms Suffice) *For every mechanism X and user u, there is a direct mechanism Z for u such that u's expected loss is the same with X and with Z.*

Proof: Fix a user u and let X denote an α-differentially private mechanism for u. Let Y denote an optimal remap of X for u. By linearity, the induced mechanism $Z = Y \circ X$ is an α-differentially private mechanism with range N. Since Y is an optimal remap of X for u, the identity remap of Z is optimal for u. Thus Z is a direct mechanism for u, and its expected loss (for u) is the same as that of X. ■

Encouraged by Lemma 5.2, we now consider only direct mechanisms for a given user. In the notation of Sections 2.2 and 2.3, we are assuming that $X = Z$. Fix a user u with prior p over N and legal loss function l. Optimizing this user's expected loss (2) over all direct differentially private mechanisms can be expressed as the following linear program.

User-specific LP:

$$\text{minimize} \quad \sum_{i \in N} p_i \sum_{j \in N} x_{ij} \cdot l(i, j) \tag{3}$$

$$x_{ij} - \alpha \cdot x_{(i+1)j} \geq 0 \qquad \forall i \in N \setminus \{n\}, \forall j \in N \tag{4}$$

$$\alpha \cdot x_{ij} - x_{(i+1)j} \leq 0 \qquad \forall i \in N \setminus \{n\}, \forall j \in N \tag{5}$$

$$\sum_{j \in N} x_{ij} = 1 \qquad \forall i \in N \tag{6}$$

$$x_{ij} \geq 0 \qquad \forall i \in N, \forall j \in N. \tag{7}$$

Constraints (6) and (7) ensure that every feasible solution to this linear program can be interpreted as a probabilistic function from N to N — i.e., as a mechanism with range N. Constraints (4) and (5) enforce α-differential privacy for every feasible solution. The objective function is precisely the expected loss (2) incurred by user u, assuming that it employs the identity remap. Since this

linear program has a bounded and non-empty feasible region, it has at least one optimal solution. Lemma 5.2 implies that every such optimal solution corresponds to an optimal mechanism for u (that is also direct for u). Figure 3(a) displays an example of such an optimal mechanism for a particular user.

5.3 Properties of User-Optimal Mechanisms

A fundamental difficulty that our proof of Theorem 3.2 must resolve is to certify whether or not a remapped version $Y \circ T$ of the truncated α-geometric mechanism T is an optimal direct mechanism for a given user u. We approach this problem by focusing on the constraints of the linear program (4)–(7) that are binding for different mechanisms. If we can show that the same constraints are binding for a mechanism $Y \circ T$ and for a direct optimal mechanism for a given user u, and also there are enough such constraints that there is a unique point at which all of them bind, then we can conclude that $Y \circ T$ is an optimal direct mechanism for u (and hence T is optimal for u).

We phrase our arguments about binding constraints in terms of the *signature* of a feasible solution X to the user-specific linear program, which is a matrix that encodes conveniently which of the linear programming constraints are binding at X and which are slack. Formally, it is an $(N \setminus \{n\}) \times N$ matrix Σ in which each entry σ_{ij} lies in the set $\{\downarrow, \uparrow, S, 0\}$. Each row i of the signature corresponds to rows i and $i+1$ of the corresponding mechanism, and the meaning of these symbols is as follows.

1. $\sigma_{ij} = \downarrow$ if and only if x_{ij} and $x_{(i+1)j}$ are both positive and $x_{(i+1)j} = \alpha x_{ij}$ — if the probability of output j is decreasing as rapidly as possible (when moving from input i to input $i+1$), subject to α-differential privacy.

2. $\sigma_{ij} = \uparrow$ if and only if x_{ij} and $x_{(i+1)j}$ are both positive and $x_{(i+1)j} = x_{ij}/\alpha$ — if the probability of output j is increasing as rapidly as possible (when moving from input i to input $i+1$), subject to α-differential privacy.

3. $\sigma_{ij} = S$ if and only if x_{ij} and $x_{(i+1)j}$ are both positive and $\alpha x_{ij} < x_{(i+1)j} < x_{ij}/\alpha$, so that both of the corresponding privacy constraints are slack.

4. $\sigma_{ij} = 0$ if both x_{ij} and $x_{(i+1)j}$ are zero.

Under our standing assumption that $0 < \alpha < 1$, every feasible solution X to the user-specific linear program has a well-defined signature Σ. Since $\alpha < 1$, no entry meets more than one of the four conditions. Since $\alpha > 0$, a variable x_{ij} can only be zero if $x_{(i+1)j}$ is also zero (by α-differential privacy), so each entry meets at least one of the conditions. For example, Figure 3(b) shows the signature of the mechanism listed in Figure 3(a).

The next lemma states two obvious properties of signatures.

Lemma 5.3 *For every feasible solution X to (4)–(7) and corresponding signature Σ:*

(a) no column of Σ has both a zero entry and a non-zero entry;

(b) no row of Σ consists entirely of $\downarrow$ and 0 entries or entirely of $\uparrow$ and 0 entries.

input/output	0	1	2	3	4	5
0	2/3	0	1/4	1/24	1/48	1/48
1	1/3	0	1/2	1/12	1/24	1/24
2	1/6	0	1/2	1/6	1/12	1/12
3	1/12	0	1/4	1/3	1/6	1/6
4	1/24	0	1/8	1/6	1/3	1/3
5	1/48	0	1/16	1/12	1/6	2/3

(a) The mechanism X

input/output	0	1	2	3	4	5
0	↓	0	↑	↑	↑	↑
1	↓	0	S	↑	↑	↑
2	↓	0	↓	↑	↑	↑
3	↓	0	↓	↓	↑	↑
4	↓	0	↓	↓	↓	↑

(b) The corresponding signature Σ

Figure 3: In (a), an optimal 1/2-differentially private mechanism when $n = 5$ for a user with prior $(1/4, 0, 1/4, 0, 1/4, 1/4)$ and loss function $l(i, j) = |i - j|^{1.5}$. In (b), the signature of this mechanism.

Proof: Part (a) is immediate from α-differential privacy with $\alpha > 0$: if $x_{ij} = 0$, then $x_{(i-1)j} = x_{(i+1)j} = 0$ as well. Part (b) is equally simple: if row i consists entirely of $\downarrow$ and 0 entries (respectively, $\uparrow$ and 0 entries), then the total probability mass in row $i + 1$ of the mechanism X is exactly α times (respectively, $1/\alpha$ times) that in row i of the mechanism. But $\alpha \neq 1$ and every row of mechanism X must have probability mass 1 (as in constraint (6)), a contradiction. ■

By design, a signature encodes which privacy constraints (4) and (5) are tight and which are slack. The signature also implicitly encodes the state of the nonnegativity constraints (7): $x_{ij} > 0$ if and only if the jth column of the corresponding signature has no 0 entries. We summarize this observation as a lemma.

Lemma 5.4 *Two feasible solutions X and X' to (4)–(7) have the same signature Σ if and only if the same set of constraints are binding at X and at X'.*

At this point in the proof of Theorem 3.2, we temporarily adopt two additional assumptions about the preferences of a user; both will be discharged via a limiting argument in Section 5.6. First, we consider only users with a prior p that has *full support*, meaning that $p_i > 0$ for every $i \in N$. Second, we assume that the user's loss function l is *strictly legal*, meaning that the loss $l(i, j)$ is a strictly increasing function of $|j - i|$ for each fixed $i \in N$.

The next lemma makes use of our restriction to (strictly) legal loss functions and is central in our proof of Theorem 3.2. To state it, we need a definition: a row of a signature is *unimodal* if its non-zero entries form a sequence that begins with zero or more $\downarrow$ entries, followed by zero or one S entries, followed by zero or more $\uparrow$ entries. A signature is *unimodal* if each of its rows is unimodal. For example, the signature in Figure 3(b) is unimodal.

Lemma 5.5 *For every user u with a full-support prior and a strictly legal loss function, every optimal direct mechanism for u has a unimodal signature.*

Proof: Fix a user u with full-support prior p and strictly legal loss function l. Let X be an optimal direct mechanism for u with signature Σ. We only need to show that there is no row h and columns $k < m$ such that σ_{hk} is either S or $\uparrow$ and also σ_{hm} is either S or $\downarrow$.

To obtain a contradiction, suppose such a row h and columns $k < m$ exist. There are two cases. First suppose that h is at most the average value $(k + m)/2$ of k and m. We claim that we can obtain a superior mechanism X' from X by reassigning a small γ fraction of the probability mass in rows 0 through h of column m to the same rows of column k. Formally, for a parameter γ that

is positive but close to zero, set $x'_{im} = (1-\gamma)x_{im}$ and $x'_{ik} = x_{ik} + \gamma x_{im}$ for all $i \in \{0, 1, \ldots, h\}$; and $x'_{ij} = x_{ij}$ in all other cases.

We first note that, if feasible for the linear program (3)–(7), then X' is superior to the purported optimal solution X (the desired contradiction). Indeed, since $h \leq (k+m)/2$, $|i-k| \leq |i-m|$ for all $i \leq h$. Also, this inequality is strict for $i = 0$. Since the strictly legal loss function $l(i,j)$ depends only on $|j-i|$ for each i, and is strictly increasing in this quantity, the expected loss under each input $i \in \{0, 1, 2, \ldots, h\}$ — the inner sum in (3) — is no larger with X' than with X, and is strictly smaller for the input $i = 0$. Since the prior p has full support, the objective function value of mechanism X' is strictly smaller than that of X, for arbitrarily small positive values of γ.

We now verify the feasibility of X' for the linear constraints (4)–(7), for sufficiently small positive values of γ. The nonnegativity constraints (7) remain satisfied for small enough γ since, by the assumption that $\sigma_{hm} \neq 0$ and Lemma 5.3(a), $x_{im} > 0$ for every row i. The probability constraints (6) continue to hold because we only shifted some mass from one column to another. The privacy constraints (4) and (5) involving the unmodified rows $\{h+1, \ldots, n\}$ or the unmodified columns clearly still hold. The privacy constraints involving only the rows $\{0, 1, \ldots, h\}$ in columns k and m continue to hold by linearity. Finally, two of the four privacy constraints involving rows $h, h+1$ and columns k, m are obviously still satisfied, while the other two (that $x'_{hk} \leq x'_{(h+1)k}/\alpha$ and $x'_{hm} \geq \alpha x'_{(h+1)m}$) hold for small enough γ because of our assumption that $\sigma_{hk} \neq \downarrow, 0$ and $\sigma_{hm} \neq \uparrow, 0$.

The second case, in which the row index h exceeds the average index $(k+m)/2$ of the columns, is symmetric. Here, we obtain a superior mechanism X' from X by reassigning a small enough fraction of the probability mass in rows $h+1$ through n of column k to the same rows of column m. The formal argument is identical to that in the first case. ■

Define an ordering on the unimodal rows of a signature according to the number of $\downarrow$ entries, so that "larger" rows have more such entries. The next lemma refines Lemma 5.5 by identifying, under the same conditions, monotone structure in the (unimodal) rows of a signature. Recall that a unimodal row has at most one S entry; we call such a row *strictly unimodal* if it has no S entry.

Lemma 5.6 *For every user u with a full-support prior and a strictly legal loss function, every optimal direct mechanism for u has a signature Σ that satisfies:*

(a) the rows $i = 0, 1, 2, \ldots, n-1$ of Σ are nondecreasing; and

(b) every strictly unimodal row $i < n-1$ is followed by a row that is either strictly larger or that has an S entry.

Proof: We prove both parts of the lemma at once. Lemma 5.5 implies that every optimal direct mechanism X for u has a unimodal signature Σ. Also, recall from Lemma 5.3(a) that every column of Σ is either devoid of zero entries or comprises only zero entries.

Consider a row $i < n-1$ of Σ, corresponding to rows i and $i+1$ of the mechanism X. Let j^* denote the smallest index such that σ_{ij} is either S or $\uparrow$; such an index exists by Lemma 5.3(b). Define $a = \sum_{j<j^*} x_{ij}$, $b = x_{ij^*}$, and $c = \sum_{j>j^*} x_{ij}$. By the probability constraint (6), $a+b+c$ — the total probability mass in row i of X — equals 1. The total mass in row $i+1$ of X also equals 1. The entries of row i of Σ indicate that the values in columns $j < j^*$ "contract" while those in columns $j > j^*$ "expand" in row $i+1$ relative to row i, so the latter mass equals $\alpha a + b' + c/\alpha$, where $b' = x_{(i+1)j^*}$ is a number between αb and b/α. Express this mass as $\alpha a + \gamma(b+c)$ for an appropriate $\gamma \leq 1/\alpha$. Note that $\gamma \geq 1$ because $\alpha < 1$ and $a+b+c = 1 = \alpha a + \gamma(b+c)$.

For a contradiction, assume that row $i+1$ of the signature Σ is not larger than row i and has no S entry. Intuitively, our goal is to show that the overall rate of expansion moving from row $i+1$ to row $i+2$ of the mechanism X is then strictly larger than that from i to $i+1$, which violates the property that all rows have equal probability mass. Formally, this assumption and the unimodality of the rows of Σ imply that $\sigma_{(i+1)j} = \uparrow$ or 0 for all $j \geq j^*$. The total mass in row $i+2$ of X is therefore at least $\alpha^2 a + \frac{\gamma}{\alpha}(b+c)$. We can derive

$$\alpha a - \alpha^2 a < a - \alpha a = \gamma(b+c) - (b+c) \leq \gamma^2(b+c) - \gamma(b+c) \leq \frac{\gamma}{\alpha}(b+c) - \gamma(b+c)$$

using the facts that $\alpha < 1$, $\alpha a + \gamma(b+c) = a+b+c$, $\gamma \geq 1$, and $\gamma \leq 1/\alpha$. This inequality implies that the probability mass in row $i+2$ of X is at least

$$\alpha^2 a + \frac{\gamma}{\alpha}(b+c) > \alpha a + \gamma(b+c) = 1,$$

which completes the contradiction. ■

5.4 Properties of User-Optimal Vertex Mechanisms

To identify more detailed properties of user-optimal privacy mechanisms, we restrict attention to optimal mechanisms of a particular form. Recall that given a collection of linear inequalities, every subset of them defines a *face* of the corresponding feasible region — the feasible solutions that meet each of the distinguished inequalities with equality. A *vertex* is a face that consists of a single point. We call a direct mechanism a *vertex mechanism* if it corresponds to a vertex of the feasible region of the linear program (4)–(7). By convexity and linearity, every linear program with a non-empty and bounded feasible region admits an optimal solution that is a vertex (e.g. [3]). The next lemma translates this fact and Lemma 5.2 into the present context.

Lemma 5.7 *For every user u, there is an optimal (direct) mechanism for u that is a vertex mechanism.*

For the next lemma, a *zero column* of a signature is a column comprising only 0 entries (recall Lemma 5.3(a)).

Lemma 5.8 *For every user u with a full-support prior and a strictly legal loss function, every optimal vertex mechanism for u has a signature Σ in which the number of S entries is at most the number of zero columns.*

Proof: Consider an optimal vertex mechanism X for the user u, with signature Σ. There are $(n+1)^2$ variables (or dimensions) in the user-specific linear program (4)–(7). A vertex of the feasible region must satisfy at least $(n+1)^2$ constraints with equality.

We now account for them. The $n+1$ constraints (6) are always met with equality. Let ζ denote the number of zero columns of Σ. By Lemma 5.3(a), the mechanism X meets precisely $\zeta(n+1)$ non-negativity constraints with equality. Thus, at least $(n+1)(n-\zeta)$ privacy constraints must bind at X. By the definition of a signature, every such binding constraint corresponds to a unique $\downarrow$ or $\uparrow$ entry of Σ. As there must be at least $(n+1)(n-\zeta)$ such entries, and exactly ζn entries of Σ are 0, at most $n(n+1) - (n+1)(n-\zeta) - \zeta n = \zeta$ entries are S. ■

The next lemma uses the row structure of a unimodal signature and Lemma 5.8 to deduce structure about the columns. Call a column of a signature *single-peaked* if it consists of zero or more $\uparrow$ entries (the *incline*), followed by zero or more S entries (the *peak*), followed by zero or more $\downarrow$ entries (the *decline*), as one traverses the rows in order $i = 0, 1, 2, \ldots, n-1$. For example, in Figure 3(b), the incline, peak, and decline of column 2 are the row index sets $\{0\}$, $\{1\}$, and $\{2, 3, 4\}$, respectively.

Lemma 5.9 *For every user u with a full-support prior and a strictly legal loss function, every optimal vertex mechanism for u has a signature Σ that satisfies:*

(P1) every non-zero column is single-peaked;

(P2) the incline of the lowest-indexed non-zero column and the decline of the highest-indexed non-zero column are $\emptyset$;

(P3) the incline of every non-zero column after the first is the union of the incline of the previous non-zero column, the peak of the previous non-zero column, and one additional row.

Proof: Consider an optimal vertex mechanism X for the user u with signature Σ with ζ zero columns. No two strictly unimodal rows of Σ are identical (by Lemma 5.6), and every such row has at least one $\downarrow$ entry and one $\uparrow$ entry (by Lemma 5.3(b)). Thus Σ contains at most $n - \zeta$ strictly unimodal rows. Since Σ has n rows and at most ζ S entries (Lemma 5.8), it has exactly $n - \zeta$ (distinct) strictly unimodal rows: for each $k \in \{1, 2, \ldots, n-\zeta\}$, there is a row r_k with exactly k $\downarrow$ entries and $(n+1-\zeta-k)$ $\uparrow$ entries (and ζ 0 entries). Lemma 5.6 implies that these rows $r_1, \ldots, r_{n-\zeta}$ appear in increasing order in Σ, with r_1 as the first row. That lemma also implies that, for every $k \in \{1, 2, \ldots, n-\zeta\}$, all of the $c_k \geq 0$ (unimodal) rows that appear after row r_k and before row r_{k+1} in Σ are identical copies of an "interpolation" of them — the sequence s_k that agrees with r_k, r_{k+1} in all but the $(k+1)$th non-zero column, which by definition is an S entry in s_k.

Thus, for every $m \in \{1, 2, \ldots, n-\zeta+1\}$, the sequence of entries in the mth non-zero column is the following: $\uparrow$ entries in the first $(m-1)+\sum_{j<m-1} c_j$ rows, corresponding to rows $r_1, \ldots, r_{m-1}$ and all copies of $s_1, \ldots, s_{m-2}$; S entries in the next c_{m-1} rows, corresponding to the copies of of s_{m-1}; and $\downarrow$ entries in the remaining rows, corresponding to $r_m, \ldots, r_{n-\zeta}$ and all copies of $s_m, \ldots, s_{n-\zeta}$. All three properties (P1)–(P3) asserted by the lemma now follow immediately. ■

5.5 The Remap

This section completes the proof of the "if" direction of Theorem 3.2 for a user with a full-support prior and a strictly legal loss function.

Lemma 5.10 *Let X be an α-differentially private mechanism and Y a remap such that $Y \circ X$ is the truncated α-geometric mechanism. For every user u with a full-support prior and a strictly legal loss function, X is optimal for u.*

Proof: We prove the lemma for the case where X is the truncated α-geometric mechanism T (and Y is the identity); the general statement follows easily by composing remaps. Let u be a user with a full-support prior and a strictly legal loss function, and let X^u be an optimal vertex mechanism for u with signature Σ with ζ zero columns.

We next use Σ to construct a deterministic remap Y of T. Let $j_1, \ldots, j_{n-\zeta+1}$ denote the indices of the non-zero columns of Σ. For $k \in \{1, 2, \ldots, n-\zeta+1\}$, define a_k as the index (in $\{0, 1, \ldots, n-1\}$) of the first row of Σ in which column j_k has an S or $\downarrow$ entry, or as n if there is no such row; and b_k as the index of the first row of Σ in which column j_k has a $\downarrow$ entry, or as n if there is no such row. Let I_k denote the contiguous set $\{a_k, \ldots, b_k\}$ of integers, which corresponds to the peak of column j_k and the extra index b_k. Properties (P1)–(P3) of Lemma 5.9 imply that the I_k's form a partition of N: (P3) guarantees that the I_k's are disjoint and that there is no integer strictly in between two consecutive I_k's; and (P2) ensures that the union of the I_k's is all of N. Define Y by mapping all inputs between a_k and b_k, inclusive, to the output j_k.

Consider the induced mechanism $Z = Y \circ T$, and let Σ^* denote its signature. We claim that $\Sigma^* = \Sigma$. Certainly, the definition of Y ensures that the non-zero columns of Σ^* are precisely $j_1, \ldots, j_{n-\zeta+1}$, as in Σ. As for a non-zero column j_k, in Z this column is the sum of columns $a_k, \ldots, b_k$ of T. Recall from Example 2.2 that the jth column of T is a positive scalar multiple of the column vector $(\alpha^j, \alpha^{j-1}, \ldots, \alpha^0, \ldots, \alpha^{n-j-1}, \alpha^{n-j})$. Summing such vectors for $j = a_k, \ldots, b_k$, we see that the ratio between the ith and $(i+1)$th entry of column j_k of Z is precisely α whenever $i \in \{0, 1, \ldots, a_k - 1\}$; precisely $1/\alpha$ whenever $i \in \{b_k, \ldots, n-1\}$; and strictly in between α and $1/\alpha$ whenever $i \in \{a_k, \ldots, b_k - 1\}$. Thus, column j_k of the corresponding signature Σ^* consists of a_k entries that are $\uparrow$, followed by $b_k - a_k$ entries that are S, followed by $(n - b_k)$ entries that are $\downarrow$. This is identical to column j_k of Σ, so $\Sigma^* = \Sigma$.

Finally, Lemma 5.4 implies that the same constraints of the user-specific linear program (4)–(7) are binding at Z and at X^u. Since X^u is a vertex, these binding constraints identify a unique α-differentially private direct mechanism, so we must have $Z = X^u$. Since X^u is an optimal direct mechanism for u, T is an optimal mechanism for u. ■

Our proof of Lemma 5.10 immediately yields the following corollary, which will be useful in proving the "only if" direction of Theorem 3.2.

Corollary 5.11 *For every user u with a full-support prior and a strictly legal loss function, every optimal vertex mechanism for u is a deterministic remap of the truncated α-geometric mechanism.*

5.6 Generalization to Arbitrary Priors and Legal Loss Functions

We now prove the "if" direction of Theorem 3.2 in its full generality, by extending Lemma 5.10 via a straightforward limiting argument.

Lemma 5.12 *Let X be a mechanism and Y a remap such that $Y \circ X$ is the truncated α-geometric mechanism. Then the mechanism X is universally utility-maximizing.*

Proof: As in Lemma 5.10, we only need to consider the case where X is the truncated α-geometric mechanism T. For a user u, let $\lambda(u)$ denote the optimal objective function value of the user-specific linear program (3)–(7). Let $\mu(u)$ denote the minimum expected loss that user u can obtain via an (optimal) remap of the truncated α-geometric mechanism.

Now consider a user u with prior p and legal loss function l. Let $\{p^k\}$ denote a sequence of full-support priors that converges to p, and $\{l^k\}$ a sequence of strictly legal loss functions that converges to l. (For example, one can use convex combinations of p and the uniform prior, and of l and the mean error loss function.) Hypothesize a user u^k with prior p^k and loss function l^k. Lemma 5.10 implies that $\lambda(u^k) = \mu(u^k)$ for each k. As both λ and μ are continuous functions of

the user's prior and loss function (e.g. [3]), this equality also holds in limit: $\lambda(u) = \mu(u)$. Since user u was arbitrary, the proof is complete. ■

5.7 The "Only If" Direction of Theorem 3.2

We finally complete the proof of Theorem 3.2 by proving the easier "only if" direction. This will follow quickly from the next lemma.

Lemma 5.13 *There exists a user u for which the truncated α-geometric mechanism is the unique optimal direct mechanism.*

Proof: Let T be the truncated α-geometric mechanism and let $\gamma > 0$ be a small positive parameter. Consider a user u with uniform prior p on N and the strictly legal loss function l with $l(i, j) = 0$ if $i = j$ and $l(i, j) = 1 + \gamma \cdot |j - i|$ otherwise. Provided γ is very small, the expected loss of a direct mechanism for u is essentially the probability that the mechanism output differs from the true query result. A simple calculation shows that the unique optimal remap of T for u is the identity map.

Now, if there are multiple optimal direct mechanisms for a user, then there are also multiple optimal vertex mechanisms for it (by linearity and convexity). By Corollary 5.11, every optimal vertex mechanism of a user is a deterministic remap of T. Since every non-identity remap of T yields a sub-optimal direct mechanism for u, T is the unique optimal direct mechanism for u. ■

To finish the proof of Theorem 3.2, suppose that a mechanism X is universally utility-maximizing. Consider the user u of Lemma 5.13, and let Y^u denote an optimal remap of X for u. The mechanism $Z^u = Y^u \circ X$ is an optimal direct mechanism for u. Lemma 5.13 implies that Z^u must be the truncated α-geometric mechanism, and the proof of Theorem 3.2 is complete.

5.8 Proof of Corollary 3.4

We can also use Lemma 5.13 to give a quick proof of the "only if" direction of Corollary 3.4 — the "if" direction follows directly from Theorem 3.2. Let u be the user asserted by Lemma 5.13. If a mechanism X is universally utility-maximizing and has range N, then there is an optimal (deterministic) remap $Y^u : N \to N$ of X such that $Y^u \circ X$ is the truncated α-geometric mechanism, the unique optimal direct mechanism for u. This can only occur if Y^u is surjective — that is, is a permutation π of N. Inverting this permutation shows that X has the form $\pi^{-1} \circ T$, which proves the corollary.

6 Beyond Oblivious Mechanisms

The definition of a universally utility-maximizing mechanism (Definition 3.1) compares the expected loss of a mechanism only to that of other *oblivious* mechanisms. Recall that in an oblivious mechanism, with respect to a fixed query f, the output distribution depends only on the query result and not on more fine-grained information about the database itself. Natural mechanisms, including all those studied in the literature thus far, are oblivious. We next give a sense in which, for utility-maximizing privacy mechanism design, restricting to oblivious mechanisms is without loss of generality.

We need to define an optimal (non-oblivious) mechanism for a user. Recall that a mechanism is a probabilistic function from databases to some range R. We use x_{dr} to denote the probability that such a mechanism X outputs $r \in R$ when the input database is $d \in D^n$. The obvious way to proceed is to assume that a user has a prior over databases. Restricting attention to direct mechanisms X for a user — justified by a suitable analog of Lemma 5.2 — the natural objective is to minimize the user's expected loss

$$\sum_{d \in D^n} p_d \sum_{j \in N} x_{dj} \cdot l(f(d), j), \tag{8}$$

where the expectation is over both the user's prior p (over databases) and also the internal coin flips of the mechanism. Unfortunately, there is no single differentially private mechanism that can be remapped (in user-specific ways) to be simultaneously optimal for all users with different priors over databases (see Appendix B).

Instead, in the user-specific optimal mechanism design problem, we assume that a user only has a prior over query results (as in our original model), rather than a richer prior over databases. Under this assumption and with a non-oblivious mechanism, it is not clear how to formulate a posterior distribution (even just over query results). We therefore assume that, when interacting with a non-oblivious mechanism, a user chooses a remap to minimize its *worst-case* expected loss over all priors over databases that are consistent with its prior over query results. Precisely, consider a user with prior $\{p_i\}$ over query results, a legal loss function l, a mechanism X (from D^n to a range R), and a remap Y (from R to N). We write $\{p_d\} \to \{p_i\}$ to mean that the prior $\{p_d\}$ over databases is consistent with the prior $\{p_i\}$ over results, in the sense that $p_i = \sum_{d:\, f(d)=i} p_d$ for every $i \in N$. We define the expected loss of this user for the mechanism X and remap Y as

$$\max_{\{p_d\} \to \{p_i\}} \sum_{d \in D^n} p_d \sum_{j \in N} l(f(d), j) \sum_{r \in R} x_{dr} y_{rj}.$$

We assume that a rational user with a prior over query results and a loss function employs a remap Y that minimizes this quantity. As in Lemma 5.2, for optimal privacy mechanism design we can then confine our attention to direct mechanisms for the user (where the identity remap is optimal), leading to our final objective function of minimizing

$$\max_{\{p_d\} \to \{p_i\}} \sum_{d \in D^n} p_d \sum_{j \in N} x_{dj} \cdot l(f(d), j). \tag{9}$$

If X is an oblivious mechanism, then every extension $\{p_d\}$ of $\{p_i\}$ yields the same expected loss. Thus, this objective function coincides with our original one (2) for oblivious mechanisms.

We call a direct differentially private mechanism *optimal* for a user if it minimizes (9) over all such mechanisms. Conceptually, this benchmark allows a mechanism to take full advantage of all information in the input database, but precludes correlation of non-oblivious behavior with detailed knowledge of a prior over databases. We emphasize that the role of this benchmark is simply to provide further evidence that geometric mechanisms offer robust solutions to privacy-preserving utility-maximization, and not to model literally a privacy mechanism design problem or canonical user behavior.

For every user, the objective function in (9) is minimized by an oblivious mechanism.

Proposition 6.1 *For every database size, privacy level, count query, and user u with a prior over query results, there is an oblivious differentially private mechanism that is optimal for u under* (9)*.*

Proof: Fix a privacy level α, a database size n, a count query f, a prior p over query results, and a legal loss function l. Let X be an optimal (direct) mechanism for this user. We define a mechanism X' that is oblivious and α-differentially private, and with worst-case expected loss (under the identify remap) no larger than that of X.

We proceed by an averaging argument. For $i \in N$, let S_i denote the databases d with $f(d) = i$. For a database $d^* \in D^n$ and output $j \in N$, define x'_{d^*j} as the average value of x_{dr} over the databases d with the same query result: $x'_{d^*j} = \sum_{d \in S_{f(d^*)}} x_{dj}/|S_{f(d^*)}|$. This defines a mechanism X' — for each input, we have specified a valid probability distribution over N — and it is oblivious by construction.

We claim that X' is α-differentially private. To prove it, fix $j \in N$ and $i \in N \setminus \{n\}$ arbitrarily. Let $V = S_i$, $W = S_{i+1}$, and let E denote all pairs (d_1, d_2) of neighboring databases $d_1 \in V$ and $d_2 \in W$. The neighbors in W of a database $d_1 \in V$ are precisely the databases that can be obtained from d_1 by changing the value of a row that does not satisfy the given predicate to one that does. Thus, interpreting (V, W, E) as a bipartite graph, it is left-regular with degree $a = (n-i) \cdot t$, where t is the number of elements of D that satisfy the predicate of the counting query f. Similarly, (V, W, E) is right-regular with degree $b = (i+1) \cdot (|D| - t)$. By the α-differential privacy of X, $x_{d_1 j}$ lies between $\alpha x_{d_2 j}$ and $x_{d_2 j}/\alpha$ for each such pair. Summing over all such pairs yields

$$a \sum_{d_1 \in V} x_{d_1 j} = \sum_{(d_1,d_2) \in E} x_{d_1 j} \leq \sum_{(d_1,d_2) \in E} (x_{d_2 j}/\alpha) = \frac{b}{\alpha} \sum_{d_2 \in W} x_{d_2 j},$$

and dividing through by $|E| = a|V| = b|W|$ gives $x'_{ij} \leq x'_{(i+1)j}/\alpha$. Similarly, $x'_{ij} \geq \alpha x'_{(i+1)j}$, completing the proof of the claim.

We now show that the worst-case expected loss of X' with the identity remap (and hence also with an optimal remap) is no larger than that of X. Since X' is oblivious, the value of the sum in (9) is the same for every prior distribution over databases that induces the prior $\{p_i\}$, and is equal to

$$\sum_{i \in N} p_i \sum_{j \in N} x'_{ij} \cdot l(i,j) = \sum_{i \in N} p_i \sum_{d \in S_i} \frac{1}{|S_i|} \cdot \left(\sum_{j \in N} x_{dj} \cdot l(i,j) \right). \tag{10}$$

Now consider the prior distribution over databases in which, for each $i \in N$, there is p_i probability mass on the database of S_i that maximizes the quantity in parentheses in (10), and zero mass on the rest of S_i. The expected loss of the given mechanism X on this prior over databases is at least the worst-case expected loss (9) of the oblivious mechanism X' with the identity remap. ■

Combining Theorem 3.2 and Proposition 6.1, the truncated α-geometric mechanism, and everything that can be remapped to it, is optimal among all (not necessarily oblivious) mechanisms for every user with respect to the objective (9).

7 Future Directions

We proposed a general model of user utility in which a user is represented by a prior over query results (modeling side information) and a loss function (modeling preferences over possible perturbations of a query result). Our main result (Theorem 3.2) singles out the truncated α-geometric mechanism as universally utility-maximizing, in the sense that it is simultaneously optimal for all

users. This result strongly advocates using random perturbations drawn from a two-sided geometric distribution as the best way to implement a differentially private count query.

An obvious and important research agenda is to study utility-maximization for privacy mechanisms with multiple and more complex queries. A second research direction is to consider alternative notions of privacy — and more generally, abstract design constraints — such as the relaxation of differential privacy that accommodates both additive and relative changes in output probabilities (see [9]).

We do not necessarily expect a guarantee as strong as Theorem 3.2 in these more general contexts. The main point of further research on this topic should be to identify privacy mechanism design techniques that, in some rigorous sense, are robustly good for user utility. Additional assumptions should be adopted as needed in service of this overarching goal. For example, the class of allowable queries could be restricted in a way orthogonal to the present work; the set of potential users could be limited; approximate optimality could replace full optimality; and the "obvious" benchmark for (approximate) optimality could be weakened to enable positive results (cf., Proposition 6.1).

8 Acknowledgments

We thank Preston McAfee, John C. Mitchell, Rajeev Motwani, David Pennock and the anonymous STOC referees for many useful comments.

References

[1] L. Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In *Proceedings of the 16th International Conference on World Wide Web (WWW)*, pages 181–190, 2007.

[2] B. Barak, K. Chaudhuri, C. Dwork, S. Kale, F. McSherry, and K. Talwar. Privacy, accuracy, and consistency too: a holistic solution to contingency table release. In *Proceedings of the 26th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, pages 273–282, 2007.

[3] D. Bertsimas and J. N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, 1997.

[4] A. Blum, C. Dwork, F. McSherry, and K. Nissim. Practical privacy: The SuLQ framework. In *Proceedings of the 24th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, pages 128–138, 2005.

[5] 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 (STOC)*, pages 609–618, 2008.

[6] U. S. Census Bureau. The 2008 statistical abstract. http://www.census.gov/compendia/statab/.

[7] I. Dinur and Nissim K. Revealing information while preserving privacy. In *Proceedings of the 22nd ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, pages 202–210, 2003.

[8] C. Dwork. Differential privacy. In *Proceedings of the 33rd Annual International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 4051 of *Lecture Notes in Computer Science*, pages 1–12, 2006.

[9] C. Dwork. Differential privacy: A survey of results. In *5th International Conference on Theory and Applications of Models of Computation (TAMC)*, volume 4978 of *Lecture Notes in Computer Science*, pages 1–19, 2008.

[10] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Third Theory of Cryptography Conference (TCC)*, volume 3876 of *Lecture Notes in Computer Science*, pages 265–284, 2006.

[11] C. Dwork, F. McSherry, and K. Talwar. The price of privacy and the limits of LP decoding. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing (STOC)*, pages 85–94, 2007.

[12] C. Dwork and K. Nissim. Privacy-preserving datamining on vertically partitioned databases. In *24th Annual International Cryptology Conference (CRYPTO)*, volume 3152 of *Lecture Notes in Computer Science*, pages 528–544, 2004.

[13] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith. What can we learn privately? In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 531–540, 2008.

[14] S. P. Kasiviswanathan and A. Smith. A note on differential privacy: Defining resistance to arbitrary side information. http://arxiv.org/abs/0803.3946v1, 2008.

[15] A. Mas-Colell, M. D. Whinston, and J. R. Green. *Microeconomic Theory.* Oxford University Press, New York, 1995.

[16] F. McSherry and K. Talwar. Mechanism design via differential privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 94–103, 2007.

[17] A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy (SP)*, pages 111–125, 2008.

[18] K. Nissim, S. Raskhodnikova, and A. Smith. Smooth sensitivity and sampling in private data analysis. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing (STOC)*, pages 75–84, 2007.

[19] Wikipedia. AOL search data scandal. http://en.wikipedia.org/wiki/AOL_search_data_scandal.

A Necessity of Legal Loss Functions in Theorem 3.2

Figure 4 demonstrates that Theorem 3.2 does not hold if the restriction to legal loss functions is dropped. From a geometric perspective, this example gives a vertex of the feasible region of the user-specific linear program (4)–(7) that is not a remap of the truncated α-geometric mechanism.

Input/Output	0	1	2	3
0	1/3	1/3	1/3	0
1	2/3	1/6	1/6	0
2	1/3	1/3	1/3	0
3	1/6	1/6	2/3	0

(a) A vertex of the feasible region (4)–(7)

Input/Output	0	1	2	3
0	1	0	0	1
1	0	1	1	0
2	0	0	1	1
3	1	1	0	1

(b) An illegal loss function

Figure 4: A mechanism that is a vertex of the user-specific linear program with $n = 3$ and $\alpha = 1/2$. It is not derivable from the truncated α-geometric mechanism via a remap. The mechanism is optimal for a user with the (illegal) loss function shown in (b) and a uniform prior on $\{0, 1, 2, 3\}$

B Impossibility of Universal Utility-Maximization with Non-Oblivious Mechanisms

Recall from Section 6 that a universally utility-maximizing non-oblivious and differentially private mechanism X for users with priors over databases satisfies the following guarantee: for every user u with a prior p over databases and a legal loss function l, there is a (user-specific and optimal) remap Y of X such that the induced mechanism $Y \circ X$ minimizes (8) over all direct mechanisms for u. This section demonstrates that no such mechanism exists.

Take $D = \{0, 1\}$, $n = 3$, and $\alpha = \frac{1}{2}$. The query f counts the number of "1" rows. Consider two users, both with legal loss function $l(i, j) = |j - i|^{1-\delta}$ for small $\delta > 0$. The first has the following prior distribution over databases: the subset of the three rows that are "1" is $\{1\}$ with probability 1/4, $\{2\}$ with probability 1/4, $\{1, 3\}$ with probability ϵ, and $\{2, 3\}$ with probability $\frac{1}{2} - \epsilon$. (Here ϵ is sufficiently small.) The second user's prior is the same except with the probabilities of the subsets $\{1, 3\}$ and $\{2, 3\}$ exchanged. The unique optimal direct mechanisms for these users are shown in Figure 5.

Input DB/Output	1	2
{1}	11/12	1/12
{2}	2/3	1/3
{1, 3}	5/6	1/6
{2, 3}	1/3	2/3

(a) Optimal mechanism for user #1

Input DB/Output	1	2
{1}	2/3	1/3
{2}	11/12	1/12
{1, 3}	1/3	2/3
{2, 3}	5/6	1/6

(b) Optimal mechanism for user #2

Figure 5: The unique optimal direct mechanisms for the two users of the example. Unshown outputs occur with probability zero. Unshown inputs have prior probability zero and can be ignored.

Suppose that X is a universally utility-maximizing mechanism. Let Y_1 and Y_2 be remaps such that $Y_1 \circ X$ and $Y_2 \circ X$ are the mechanisms shown in Figure 5(a) and 5(b), respectively. We use the

notation $x_{d,jk}$ to denote the probability that X, with the input database d, outputs a result that is mapped to $j \in \{1,2\}$ by Y_1 and to $k \in \{1,2\}$ by Y_2. Thus, for example, if $\{1\}$ denotes the database that has a "1" only in the first row, these probabilities obey the equations $x_{\{1\},11} + x_{\{1\},12} = \frac{11}{12}$ and $x_{\{1\},11} + x_{\{1\},21} = \frac{2}{3}$.

Constraints of this type imply that $x_{\{1\},12} = \frac{1}{3} - x_{\{1\},22}$ and $x_{\{2\},12} = \frac{1}{12} - x_{\{2\},22}$. Since X is $\frac{1}{2}$-differentially private and the databases $\{1\}$ and $\{2\}$ differ in only two rows, $x_{\{1\},12} \leq 4x_{\{2\},12}$ and hence $4x_{\{2\},22} \leq x_{\{1\},22}$. The analogous argument using $x_{\{1\},21}, x_{\{2\},21}$ in place of $x_{\{1\},12}, x_{\{2\},12}$ implies that $4x_{\{1\},22} \leq x_{\{2\},22}$ and hence $x_{\{1\},22} = x_{\{2\},22} = 0$. Since X is $\frac{1}{2}$-differentially private, $x_{\{1,3\},22} = 0$ as well. Solving, we can obtain $x_{\{1\},11} = \frac{7}{12}$ and $x_{\{1,3\},11} = \frac{1}{6}$. Since databases $\{1\}$ and $\{1,3\}$ differ in only one row, this contradicts the assumption that X is $\frac{1}{2}$-differentially private.