On Optimal Differentially Private Mechanisms for Count-Range Queries[*]

Chen Zeng
University of Wisconsin-Madison
zeng@cs.wisc.edu

Jin-Yi Cai
University of Wisconsin-Madison
jyc@cs.wisc.edu

Pinyan Lu
Microsoft Research Asia
pinyanl@microsoft.com

Jeffrey F. Naughton
University of Wisconsin-Madison
naughton@cs.wisc.edu

ABSTRACT

While there is a large and growing body of literature on differentially private mechanisms for answering various classes of queries, to the best of our knowledge "count-range" queries have not been studied. These are a natural class of queries that ask "is the number of rows in a relation satisfying a given predicate between two integers θ_1 and θ_2?" Such queries can be viewed as a simple form of SQL "having" queries. We begin by developing a provably optimal differentially private mechansim for count-range queries for a single consumer. For count queries (in contrast to count-range queries), Ghosh et al. [9] have provided a differentially private mechanism that simultaneously maximizes utility for multiple consumers. This raises the question of whether such a mechanism exists for count-range queries. We prove that the answer is no — for count range queries, no such mechanism exists. However, perhaps surprisingly, we prove that such a mechanism does exist for "threshold" queries, which are simply count-range queries for which either $\theta_1 = 0$ or $\theta_2 = +\infty$. Furthermore, we prove that this mechanism is a two-approximation for general count-range queries.

Categories and Subject Descriptors

H.2.0 [**Database Management**]: General—*Security*

General Terms

Algorithms, Security

1. INTRODUCTION

Recently, concomitant with the increasing ability to collect personal data, privacy has become a major concern. In response to this concern, the research community has devoted considerable attention to differentially private mechanisms for various classes of queries, including averages, sums, and counts. However, to the best of our knowledge, there is no published work on count-range queries

A count-range query tests if the number of rows satisfying a given predicate is within a specified range — that is, they ask "is the count within this range?" Count-range queries are a natural generalization of count queries, and correspond to a simple form of SQL "having" queries. An obvious differentially private mechanism for evaluating count-range queries is to count the number of rows satisfying the predicate, then to add Laplacian noise or geometric noise to the count, and to return *yes* if the noisy result is within the range and *no* otherwise. Our question in this paper is whether it is possible to do better. We answer in the affirmative, and give a different, optimal mechanism; but before doing so, we must first specify what "better" means.

Of course, privacy is just one aspect of the problem; utility also matters, as adding noise decreases accuracy. We measure the utility of a differentially private mechanism for count-range queries in terms of weighted errors. We adopt a model in which each information consumer provides an *error penalty function* that describes that consumer's perceived utility loss for a given error. For example, for a typical information consumer, errors that are in some sense "close" to the true answer may be less harmful than errors that are "farther" away.

However, different consumers may assign a different importance (weight) to the same error. In addition, each consumer may also have arbitrary side information about the data being queried. In an approach similar to that presented in [9] for count queries, we model a consumer's side information as a *prior distribution* over the number of rows satisfying the predicate of a count-range query. We combine a consumer's error penalty function and prior distribution to give a weighted error function. Therefore, returning to the issue of whether or not it is possible to do better than the naïve approach, the question becomes: given a privacy parameter, a consumer's weighted error penalty function and prior distribution, does adding geometric noise to the count and then checking the range minimize the consumer's weighted error? We show that the answer is "no", and propose a different, optimal differentially private mechanism for count-

[*]This work was supported by National Science Foundation grants IIS-0524671, CCF-0914969 and NIGMS grant R01LM011028-02.

ICDT '13, March 18 - 22 2013, Genoa, Italy,
Copyright 2013 ACM 978-1-4503-1598-2/13/03... $ 15.00..

range queries.

With this result, we turn to consider a generalization in which the differentially private mechanism must serve multiple information consumers, each asking the same count-range query and each with their own weighted error function. A natural question is how to guarantee optimal utility for all such consumers. A naïve solution is to apply the single consumer mechanism separately for each consumer. However, this would result in the release of multiple randomizations of the query result, which would allow consumers to collude and reduce the effective noise in the answers. We seek a better alternative.

In the context of count queries (not count-range queries), [9, 10] showed that there is a more sophisticated differentially private mechanism that is both *collusion-resistant* and *simultaneously* optimal for every consumer. Their approach works by first perturbing the result of the count query, and then individually transforming that noisy result for each consumer. They proved that when the mechanism is the range-restricted geometric mechanism [9], the transformation guarantees optimal utility for every consumer.

Since count-range queries are a natural generalization of count queries, it is natural to ask if a similar approach works for count-range queries. We prove that the answer is no — in fact, we show that there is no differentially private mechanism that simultaneously maximizes every consumer's utility for count-range queries. On a more positive note, we prove the range-restricted geometric mechanism is an *approximate* universally utility maximizing mechanism for count-range queries, and that the weighted error for any consumer is at most twice that consumer's optimal weighted error.

Next, we consider threshold queries, a natural special case of count-range queries. Here, by "threshold queries" we mean queries that test if the number of rows satisfying a predicate is less/greater than a constant. That is, if a count-range query asks if the count is between two constants θ_1 and θ_2, threshold queries are just count-range queries for which either $\theta_1 = 0$ or $\theta_2 = +\infty$. Perhaps surprisingly, we show that with this apparently small change (requiring that one of the endpoints of the range in count-range query be zero or infinity), the range-restricted geometric mechanism guarantees optimal utility for every consumer while satisfying differential privacy.

The rest of this paper is organized as follows: Section 2 formulates the problem of guaranteeing differential privacy for count-range queries, and defines our utility model. Section 3 presents our results for count-range queries while Section 4 considers threshold queries. Section 5 discusses related work, while we conclude and discuss future directions in Section 6. Proofs not found in this submission are presented in the long version of our paper [1].

2. PRELIMINARIES

2.1 Count-Range Queries and Diff. Privacy

A *database* is a collection of rows. The domain of each row is a finite set D. The domain of a database of n rows is thus represented as D^n. In the rest of this paper, we shall use n to denote the number of rows in a database.

We will focus on a class of queries called *count-range queries*, where a count-range query consists of three parameters: a predicate p and two non-negative integers θ_1, θ_2 where $(\theta_1 < \theta_2)$[1]. Thus, we can characterize a count-range query by a triple $\langle p, \theta_1, \theta_2 \rangle$. Given a count-range query $\langle p, \theta_1, \theta_2 \rangle$ with predicate $p : D \to \{true, false\}$, the result of that count-range query is *yes* if the number of rows in a database that satisfy the predicate p is within the range $[\theta_1, \theta_2]$, and *no* otherwise. For ease of presentation, we define the *count* of a count-range query to be the number of rows satisfying the predicate of that query. When $\theta_1 = 0$ or $\theta_2 = +\infty$, the count-range query is a *threshold query.*

Our first goal is to propose a *differentially private* mechanism for count-range queries. In our context, a *mechanism* is a probabilistic function from D^n to some range R. Typical ranges include the real numbers, the integers, subranges of integers, and $\{yes, no\}$. For a mechanism X with a countable range R, we use $x_{\tau,r}$ to denote the probability of outputting $r \in R$ when the underlying database is τ. A mechanism X is called *α-differentially private* $(\alpha > 1)$ if and only if for any pair of databases τ, τ' that differ by one row, $\forall r \in R$, $x_{\tau',r}/\alpha \le x_{\tau,r} \le \alpha x_{\tau',r}$. Two such databases τ and τ' are called *neighboring databases.*

We want to guarantee differential privacy for a count-range query. We formalize that problem next.

2.2 Diff. Private Mechanisms for Count-Range Queries

Because the result of a count-range query is either *yes* or *no*, given a differentially private mechanism X for a count-range query, let $x_{\tau,1}$ $(x_{\tau,0})$ be the probability of outputting *yes* (*no*) when the underlying database is $\tau \in D^n$. Because $x_{\tau,0} = 1 - x_{\tau,1}$, we can characterize a differentially private mechanism for a count-range query by $x_{\tau,1}$.

We assume that the probability that the result of a count-range query is *yes* is related to the count, the range, and the privacy parameter. More precisely, let μ_τ and $\mu_{\tau'}$ be the count of a count-range query over the databases τ and τ', respectively. Fixing the range and the privacy parameter, if $\mu_\tau = \mu_{\tau'}$, then we assume that $x_{\tau,1} = x_{\tau',1}$[2]. In the rest of this paper, unless otherwise specified, we assume that the range $[\theta_1, \theta_2]$ and the privacy parameter α are fixed. We call mechanisms satisfying our assumption *count-oriented* mechanisms.

Definition 1. *(Count-oriented mechanism): A differentially private mechanism is* count-oriented *if and only if the output distributions produced by that mechanism on any pair of databases that have the same counts for a count-range query are identical.*

[1] We do not consider the case for $\theta_1 = \theta_2$ since a count-range query is equivalent to a count query in that case, which was considered in [9].

[2] The rationale of this assumption is discussed in the long version of our paper [1].

We introduce a function ϕ to characterize a *count-oriented* differentially private mechanism for a count-range query. Let $x_{\tau,1} = \phi(\mu)$ where μ is the count of that count-range query when the underlying database is τ. Of course, not every function ϕ can be used to define that probability. The following two basic properties on ϕ capture the requirement:

Definition 2. *(Legal function): A function ϕ is a* legal function *if and only if for any integer μ,*

1. *for $0 \leq \mu \leq n$, $0 \leq \phi(\mu) \leq 1$;*
2. *for $0 \leq \mu < n$, $\phi(\mu)/\alpha \leq \phi(\mu+1) \leq \alpha\phi(\mu)$ and $(1-\phi(\mu))/\alpha \leq 1-\phi(\mu+1) \leq \alpha(1-\phi(\mu))$.*

The second property comes from the requirement of differential privacy that the ratio of the probabilities of outputting the same result (either *yes* or *no*) for any pair of neighboring databases must be bounded by the privacy parameter α. Thus, a legal function naturally corresponds to a count-oriented differentially private mechanism for a count-range query. Of course, there are many functions satisfying those two properties. Let Ω be the set of all legal functions: $\Omega = \{\phi \mid \phi \text{ is a legal function}\}$. Next, we propose a utility model to quantify the quality of a legal function. Our first goal in this paper is to find an optimal legal function that maximizes the utility of a consumer.

2.3 Utility Model

Because differentially private mechanisms are probabilistic, they commit errors. Thus, informally, the best differentially private mechanism should be the least likely to commit errors. Specifically, there are two types of errors in answering a count-range query:

1. False negative: the output is *no* but the correct answer is *yes*. The probability of a legal function ϕ to commit a false negative error when the count is μ is:
$$F^-_\phi(\mu) = 1 - \phi(\mu),\ \theta_1 \leq \mu \leq \theta_2$$
2. False positive: the output is *yes* but the correct answer is *no*. The probability of a legal function ϕ to commit a false positive error when the count is μ is:
$$F^+_\phi(\mu) = \phi(\mu),\ 0 \leq \mu < \theta_1 \text{ or } \theta_2 < \mu \leq n$$

It is possible that different errors incur different utility losses for different consumers. Consider the following example: when the count is equal to θ_1, and the output is *no*, then that error is close to being correct in the sense that the correct answer will change from *yes* to *no* upon deleting even a single row that satisfies the predicate. Thus, that error may not severely impact utility. On the other hand, if the count is much larger than θ_2 and the output is *no*, the error might incur a large utility loss because it is far from being correct. Furthermore, each consumer may have a different tolerance on this type of errors. Therefore, we introduce an *error penalty function* ω for a consumer where $\omega(i)$ is the penalty to the error of a legal function when the count is i. The idea of error penalty function was proposed in [9, 10] for count queries, and we extend that idea to count-range queries. In both [9, 10], the error penalty functions are assumed to be monotone such that the error penalty function must be non-decreasing in the difference between the correct result of a count query and the output. In our work we do not require such property for ω, which provides greater flexibility in modeling a consumer's perceived utility loss for different errors.

Following the model presented in [9] in their study of count queries, we also assume that each consumer has side information about the underlying database. We model that side information as a prior probability distribution ρ over the count, where $\rho(i)$ represents the probability that a consumer believes the count of the given threshold query to be i. That prior distribution represents the beliefs of that consumer, which might stem from other information sources, previous interactions with the database, 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 adversary, and use a prior only to discuss the utility of a legal function to a potential consumer. We model the utility of a legal function for a consumer in terms of her weighted error:

$$err(\phi) = \sum_{i=0}^{\theta_1-1}\omega(i)\rho(i)\phi(i) + \sum_{i=\theta_1}^{\theta_2}\omega(i)\rho(i)(1-\phi(i)) + \sum_{i=\theta_2+1}^{n}\omega(i)\rho(i)\phi(i) \tag{1}$$

3. OPTIMAL PRIVATE MECHANISMS FOR COUNT-RANGE QUERIES

In this section, we first propose an optimal differentially private mechanism for count-range queries, then consider the problem of serving multiple consumers. Ghosh et al. [9] showed that there is a mechanism for count queries that simultaneously maximizes every consumer's utility while guaranteeing differential privacy. Although count-range queries are a simple generalization of count queries, surprisingly, our results indicate that there is no such mechanism for count-range queries.

3.1 An Optimal Diff. Private Mechanism

First, we define the notion of an *optimal legal function* for a count-range query by a single consumer.

Definition 3. *Given a consumer with error penalty function ω and a prior distribution ρ, a legal function ϕ^* is an* optimal legal function *for a count-range query by that consumer if and only if $\forall \phi \in \Omega$,*

$$err(\phi^*) \leq err(\phi)$$

where $err()$ is the weighted error function defined in (1).

A straightforward way to find an optimal legal function for a consumer is to treat each $\phi(i)$, $0 \leq i \leq n$, as a variable, and to solve the linear programming problem that minimizes her weighted error subject to the requirements of a legal function. This amounts to solving an optimization problem of $n+1$ variables. However, we will prove that for the design of an optimal legal function, it suffices to solve an optimization problem with two variables. First, we prove a theorem about the existence of an optimal legal function of a particular form. To better understand our results, we define two recurrence relations:

$$\psi_1(\mu+1) = \min\{\alpha\psi_1(\mu), \frac{\alpha - 1 + \psi_1(\mu)}{\alpha}\} \qquad (2)$$

and

$$\psi_2(\mu+1) = \max\{\frac{1}{\alpha}\psi_2(\mu), 1 - \alpha + \alpha\psi_2(\mu)\} \qquad (3)$$

We will prove that an optimal legal function for a count-range query $\langle p, \theta_1, \theta_2\rangle$ can be characterized by the minimum of the two recurrence relations defined in (2) and (3).

THEOREM 1. *An optimal legal function ϕ^* for the count-range query $\langle p, \theta_1, \theta_2\rangle$ is of the following form:*

$$\phi^*(\mu) = \min\{\psi_1(\mu), \psi_2(\mu)\} \qquad (4)$$

where ψ_1 and ψ_2 satisfy (2) *and* (3)*, respectively, and*

$$\begin{aligned} \psi_1(\theta_1) &\le \psi_2(\theta_1) \\ \psi_1(\theta_2) &\ge \psi_2(\theta_2) \end{aligned} \qquad (5)$$

As we will explain in more detail later, (2) actually characterizes a family of optimal legal functions for the threshold query $\langle p, \theta_1, +\infty\rangle$ while (3) does so for the threshold query $\langle p, 0, \theta_2\rangle$. Since a count-range query $\langle p, \theta_1, \theta_2\rangle$ can be expressed as the “and” of the two threshold queries $\langle p, \theta_1, +\infty\rangle$ and $\langle p, 0, \theta_2\rangle$, we expect that an optimal legal function for that count-range query should be closely related to those two recurrence relations, and Theorem 1 confirms this.

By Theorem 1, when searching for an optimal legal function, it suffices to consider legal functions satisfying (4), which is an optimization problem consisting of two variables. That is, if we fix $\psi_1(0) = \beta_1$, then ψ_1 is well-defined because (2) is a first-order linear recurrence relation. More precisely, we can rewrite (2) as:

$$\psi_1(\mu+1) = \begin{cases} \alpha\psi_1(\mu) & \text{if } \psi_1(\mu) \le 1/(\alpha+1) \\ (\alpha - 1 + \psi_1(\mu))/\alpha & \text{otherwise.} \end{cases}$$

For any integer μ ($0 \le \mu \le n$), if $\beta_1 \in [0, 1/(\alpha^{n-1}(\alpha+1)))$, then

$$\psi_1(\mu) = \alpha^\mu \beta_1$$

and if $\beta_1 \in [1/(\alpha+1), 1]$, then

$$\psi_1(\mu) = 1 - \frac{1-\beta_1}{\alpha^\mu}$$

and if $\beta_1 \in [1/(\alpha^{n-1}(\alpha+1)), 1/(\alpha+1))$, let k be the unique integer between 1 and $n-1$, such that if $\beta_1 \in [1/(\alpha^k(\alpha+1)), \alpha/(\alpha^k(\alpha+1)))$, then

$$\psi_1(\mu) = \begin{cases} \alpha^\mu \beta_1 & \text{if } 0 \le \mu \le k \\ 1 - \frac{1-\beta_1\alpha^k}{\alpha^{\mu-k}} & \text{otherwise.} \end{cases}$$

Therefore, we divide the range $[0,1]$ into $n+1$ subintervals, where $\psi_1(\mu)$ is linear in β_1 when β_1 is in a subinterval. Similarly, we can show ψ_2 is also well-defined by fixing $\psi_2(0) = \beta_2$.

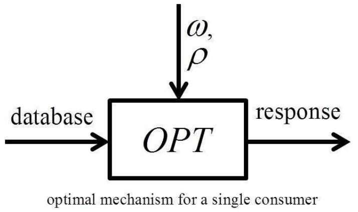

Figure 1: Consumer Dependent Optimal Mechanism

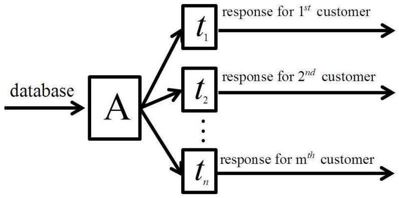
database
A
t_1
t_2
t_n
response for 1st customer
response for 2nd customer
response for mth customer

Figure 2: Serving Multiple Consumers with a Single Mechanism

Hence, the weighed error of ϕ can be written as $err(\phi) = err(\beta_1, \beta_2)$, where $err(\beta_1, \beta_2)$ is a piecewise multilinear function in β_1, β_2. To compute the minimum of $err(\beta_1, \beta_2)$, we can compute the minimum of $err(\beta_1, \beta_2)$ on each 2-dimensional subinterval, which is trivial, and then compare those local minima to get the global minimum. Let that global minimum be $err(\beta_1^*, \beta_2^*)$. Then, an optimal legal function ϕ^* is of the following form:

$$\phi^*(\mu) = \min\{\psi_1^*(\mu), \psi_2^*(\mu)\}$$

where $\psi_1^*(0) = \beta_1^*, \psi_2^*(0) = \beta_2^*$, and ψ_1^* and ψ_2^* satisfy the (2) and (3), respectively.

3.2 Multiple Consumers

While we have given an optimal solution for the single consumer case, the situation where there are multiple consumers, each with their own error penalty functions and prior distributions, is more complex. Our main question is whether there is a single optimal function that works for multiple consumers. In this section, we consider a scenario in which multiple consumers ask the same count-range query. Our goal is to enforce differential privacy for multiple consumers while simultaneously guaranteeing optimal utility for every consumer.

A naïve application of our single consumer mechanism to

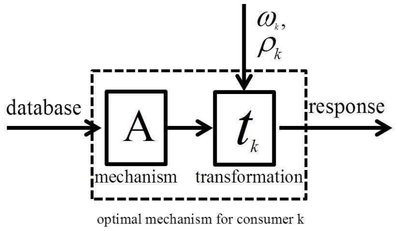

Figure 3: Decomposition of the Optimal Mechanism for a Consumer

multiple consumers is to invoke the optimal α-differentially private mechanism for each consumer separately. However, that naïve application allows colluding consumers to combine their noisy results and reduce the noise, and thus infer the real result more accurately. It is well-known that in such a situation the database has to operate under a more stringent privacy parameter to satisfy the utility requirements of the consumers. More precisely, suppose that there are m consumers, and the database guarantees α_i-differential privacy for the i^{th} consumer. By the composition property of differential privacy [7], we can only guarantee α-differential privacy for those m consumers provided $\prod_{i=1}^{m} \alpha_i \leq \alpha$, and thus, $\alpha_i \ll \alpha$. As a result, the weighted error of each consumer is actually larger than that of the optimal α-differentially private mechanism for each consumer.

The problem with collusion arises because the true answer is randomized and released multiple times. To avoid that problem, we observe that if the true query result is randomized only once, and every consumer receives the same noisy result, then the problem goes away. However, if the true query result is only randomized once, the differentially private mechanism cannot be optimal for every consumer unless they have the same error penalty function and prior distribution. That apparent paradox is resolved by assuming that the database can individually further transform the intermediate noisy output (which is the same for every consumer), and that transformation is deliberately calibrated to the consumer's parameters — her error penalty function and prior distribution — such that the combination of the differentially private mechanism and that transformation maximizes that consumer's utility.

To illustrate that idea, for a single consumer, the optimal mechanism depends on her error penalty function and prior distribution as shown in Figure 1. When serving multiple consumers, instead of invoking each consumer's optimal mechanism, the database employs a common mechanism A for every consumer to produce an intermediate noisy result, and then for each consumer, individually transforms that intermediate noisy result for each consumer to produce the output (*yes,no*) for that consumer. This is shown in Figure 2. Therefore, that approach actually decomposes a consumer's optimal mechanism into two parts, a consumer independent mechanism A, and a consumer dependent transformation as shown in Figure 3. For each consumer, if that decomposition is "lossless", then the mechanism A indirectly guarantees optimal utility for that consumer. We inquire whether there exists such a common mechanism A. In the rest of this paper, we shall refer to the consumer independent mechanism A as the "deployed mechanism." In our context, a transformation is a probabilistic reinterpretation of the intermediate noisy output produced by A. This is defined in Definition 4.

Definition 4. *(Transformation): For a deployed differentially private mechanism $A : D^n \rightarrow R$, a* transformation $\mathbf{t}$ *for a count-range query is a probabilistic function from R to $\{yes\}$. For a countable range R, t_r denotes the probability that the database reinterprets the outcome $r \in R$ of the mechanism A to yes.*

It suffices to only consider the probability of mapping an intermediate noisy result r to *yes* since $1 - t_r$ naturally corresponds to the probability of reinterpretting the outcome r to *no*. To output a noisy result for a count-range query by a consumer, let $\mathbf{t}$ be a transformation for a particular consumer, and r be the intermediate noisy result produced by the deployed mechanism A. Then the database flips a biased coin with probability t_r to output *yes*, and $1 - t_r$ to output *no*. Note that the output range of the deployed mechanism A does not necessarily correspond to $\{yes,no\}$ as the transformation will eventually remap the noisy output of A to that range. Furthermore, only the deployed mechanism A needs to be differentially private since the transformation receives the noisy output from A, which has already been differentially private.

Given a deployed differentially private mechanism A, and a transformation $\mathbf{t}$, the combination of A and $\mathbf{t}$ induces a new mechanism X for count-range queries, where the probability of returning *yes* for a database τ is: $x_\tau = \sum_{r \in R} a_{\tau,r} t_r$. Since A is differentially private, X is also differentially private by linearity. In accordance with the literature [3, 13], we say a mechanism X can be *derived* from the deployed mechanism A if there is a transformation $\mathbf{t}$ such that $X = A \circ \mathbf{t}$. Since X is actually a vector, we shall denote it by $\mathbf{x}$.

Since $\mathbf{x}$ is a differentially private mechanism for count-range queries, for the design of an optimal differentially private mechanism for count-range queries, we shall assume that $\mathbf{x}$ is count-oriented. To guarantee that, we require that the deployed differentially private mechanism is also count-oriented, and thus, we can characterize the domain of that mechanism by $\{0, \ldots, n\}$ instead of D^n. After that restriction, the induced mechanism $\mathbf{x}$ naturally corresponds to a legal function. We say the induced mechanism $\mathbf{x}$ is optimal for a consumer if and only if $\mathbf{x}$ minimizes that consumer's weighted error.

For each consumer, if the decomposition of her optimal mechanism is "lossless" in the sense that there is a transformation such that the induced mechanism of the deployed mechanism and that transformation is also optimal for her, then the deployed mechanism indirectly guarantees optimal utility for that consumer. Such a mechanism is called a *universally utility maximizing mechanism.*

Definition 5. *(Universally utility maximizing differentially private mechanism): A differentially private mechanism A is* universally utility maximizing *if and only if for* *each consumer k, there is a transformation $\mathbf{t}_k$ such that the induced differentially private mechanism $\mathbf{x}_k$ is optimal for that consumer k.*

If we can find such a mechanism, then the database can utilize that mechanism to randomize the count only once to produce an intermediate noisy result, and then store that intermediate noisy result. For every consumer, the database uses a transformation tailored for that consumer to randomize the stored intermediate noisy output. That "double-randomization" approach rules out the privacy threat of colluding consumers as even if they successfully cancel out the noise, the result is the intermediate noisy result, which is still differentially private. Furthermore, by carefully selecting a transformation for each consumer, the induced mechanism of the universally utility maximizing mechanism and that transformation guarantees optimal utility for that consumer, which provides a strong utility guarantee. Ghosh et al. [9] showed that the range-restricted mechanism is universally utility maximizing for count queries when the error penalty function is monotone, and is of the following form:

Definition 6.*(Range-restricted geometric mechanism[9]): For a given privacy parameter α and a count query Q, $\forall \tau \in D^n$, let μ be the correct result of Q over τ. The range-restricted geometric mechanism outputs $Z(\mu)$ where $Z(\mu)$ is a random variable with the following distribution for each integer z:*

$$\Pr[Z(\mu) = z] = \begin{cases} \frac{\alpha}{\alpha+1}\alpha^{-|z-\mu|} & \textit{if } z \in \{0, n\} \\ \frac{\alpha-1}{\alpha+1}\alpha^{-|z-\mu|} & \textit{if } 0 < z < n \\ 0 & \textit{otherwise.} \end{cases} \tag{6}$$

Since count-range queries are a natural generalization of count queries, one may expect that the range-restricted geometric mechanism is also universally utility maximizing for count-range queries. However, we will prove that this is not so. More surprisingly, our results indicate that there is no differentially private mechanism that is universally utility maximizing for count range queries, as stated in Theorem 2.

Theorem 2. *There is no universally utility maximizing differentially private mechanism for count-range queries.*

Note that Theorem 2 not only deals with count-oriented mechanisms, but also considers other differentially private mechanisms. To give an intuition for why Theorem 2 holds, let us first examine why the range-restricted geometric mechanism is not universally utility maximizing for count-range queries.

3.2.1 Discussion on Range-restricted Geometric Mechanism

Let us consider a special case where the size of the database $n = 3$, and the bounds for the count-range query are $\theta_1 = 1$ and $\theta_2 = 2$. In this case we can represent the optimal differentially private mechanism for the count-range query as a vector $(z_0, z_1, z_2, z_3)^t$, where z_i $(0 \le i \le 3)$ is the probability of outputting *yes* when the count is i. We can prove that when a consumer posing this query has a uniform error penalty function and a uniform prior distribution, the *only* optimal differentially private mechanism for that consumer is

$$\hat{\mathbf{x}} = (\frac{1}{\alpha+1}, \frac{\alpha}{\alpha+1}, \frac{\alpha}{\alpha+1}, \frac{1}{\alpha+1})^t$$

We can prove that the range-restricted geometric mechanism can not derive $\hat{\mathbf{x}}$, and thus is not universally utility maximizing. We can also understand why the range-restricted geometric mechanism is not universally utility maximizing for count-range queries from a more intuitive perspective: by Theorem 1, an optimal legal function for count-range queries can be rewritten into the following form:

$$\phi^*(\mu) = \begin{cases} \psi_1(\mu) & \text{if } 0 \le \mu \le \gamma \\ \psi_2(\mu) & \text{if } \gamma < \mu \le n \end{cases}$$

where γ is an integer between θ_1 and θ_2.

Let $\mathbf{r}_1 = (\psi_1(0), \ldots, \psi_1(n))^t$ and $\mathbf{r}_2 = (\psi_2(0), \ldots, \psi_2(n))^t$. As proved later in Section 4, there exists a transformation $\mathbf{t}_1$ such that the induced mechanism of the range-restricted geometric mechanism M and that transformation is $\mathbf{r}_1$, where $M \circ \mathbf{t}_1 = \mathbf{r}_1$. There also exists a transformation $\mathbf{t}_2$ such that $M \circ \mathbf{t}_2 = \mathbf{r}_2$. Therefore, intuitively, the database should pick $\mathbf{t}_1$ to transform the noisy count produced by the range-restricted geometric mechanism if the count μ does not exceed γ, and $\mathbf{t}_2$ otherwise. However, the decision of which transformation to employ depends on the correct count μ. If that decision is deterministic, then it is a violation of differential privacy since no deterministic algorithm satisfies differential privacy [5]. Therefore, that decision has to be randomized to accommodate the privacy requirement. As a result, the database will inevitably commit errors in picking the correct transformation because of the randomized nature, and thus, the combination of the range-restricted geometric mechanism and the transformation can not yield an optimal differentially private mechanism for a consumer.

3.2.2 Count-Oriented Mechanisms

Let us first assume that some universally utility maximizing mechanism exists that is a function of count. Again, we start with the special case where $n = 3$, $\theta_1 = 1$ and $\theta_2 = 2$. For a consumer with a uniform error penalty function and a uniform prior distribution, the only optimal differentially private mechanism for that consumer is:

$$\hat{\mathbf{x}} = (\frac{1}{\alpha+1}, \frac{\alpha}{\alpha+1}, \frac{\alpha}{\alpha+1}, \frac{1}{\alpha+1})^t.$$

Next, we reassign the probability mass of the uniform prior distribution such that the prior distribution ρ satisfies:

$$\rho(0)/\rho(1) = \alpha^2, \rho(2)/\rho(1) = \alpha, \rho(3) = \rho(2)$$

We can prove that for a consumer with a uniform error penalty function and such a prior distribution ρ, the *only* optimal differentially private mechanism is:

$$\hat{\mathbf{y}} = (\frac{1}{(\alpha+1)\alpha}, \frac{1}{\alpha+1}, \frac{\alpha}{\alpha+1}, \frac{1}{\alpha+1})^t$$

We characterize a differentially private mechanism A as a matrix of size $4\times m$, where the output range of A is $\{1,\ldots,m\}$ whose elements satisfy $1/\alpha \leq a_{i,j}/a_{i+1,j} \leq \alpha$. We prove that there is no universally utility maximizing mechanism by showing that no matrix satisfying differential privacy can derive both $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$. We start by investigating the properties of A.

Lemma 1. *Let* $B = \begin{bmatrix}\mathbf{p}^t \\ \mathbf{q}^t\end{bmatrix} = \begin{bmatrix} p_1 & p_2 & \ldots & p_m \\ q_1 & q_2 & \ldots & q_m\end{bmatrix}$ *be any* $2 \times m$ *matrix, where* $\mathbf{p}$ *and* $\mathbf{q}$ *are two probability vectors:* $0 \leq p_j, q_j \leq 1$ *and* $\sum_j p_j = \sum_j q_j = 1$. *Suppose they further satisfy the privacy constraints* $1/\alpha \leq p_j/q_j \leq \alpha$, *and for some* $\mathbf{t}$, *with* $0 \leq t_j \leq 1$, $B \circ \mathbf{t} = (\frac{1}{\alpha+1}, \frac{\alpha}{\alpha+1})^t$. *Then, all privacy constraints are tight: For all* $1 \leq j \leq m$, *either* $p_j/q_j = \alpha$, *or* $p_j/q_j = 1/\alpha$. *Furthermore,* $t_j = 0$ *in the former case, and* $t_j = 1$ *in the latter case.*

Proof. If for some j, $p_j/q_j > 1/\alpha$ and the corresponding $t_j \neq 0$, then $p_j t_j > q_j t_j/\alpha$. Since for every other j', $p_{j'}/q_{j'} \geq 1/\alpha$, we get $\mathbf{p}\circ\mathbf{t} > \mathbf{q}\circ\mathbf{t}/\alpha$, a contradiction. Hence for all $1 \leq j \leq m$, $p_j/q_j = 1/\alpha$ or $t_j = 0$.

Now consider $\mathbf{t}' = \mathbf{1} - \mathbf{t}$, where $\mathbf{1}$ is the vector of all ones. Note that $\mathbf{p}\circ\mathbf{t}' = 1 - \mathbf{p}\circ\mathbf{t} = \frac{\alpha}{\alpha+1}$, and $\mathbf{q}\circ\mathbf{t}' = 1 - \mathbf{q}\circ\mathbf{t} = \frac{1}{\alpha+1}$. Switching the roles of $\mathbf{p}$ and $\mathbf{q}$, we have for all $1 \leq j \leq m$, $p_j/q_j = \alpha$ or $t_j = 1$.

Since clearly $t_j = 0$ and $t_j = 1$ cannot hold simultaneously, we conclude that for all $1 \leq j \leq m$, either $p_j/q_j = \alpha$ or $1/\alpha$. In the former case, clearly $p_j/q_j \neq 1/\alpha$, and hence $t_j = 0$. Similarly when $p_j/q_j = 1/\alpha$, we have $t_j = 1$. □

Corollary 1. *If a* $4\times m$ *matrix* A *can derive both* $\hat{\mathbf{x}}$ *and* $\hat{\mathbf{y}}$, *then all the privacy constraints must be tight:* $\forall i, j$, *where* $0 \leq i < 3$, $1 \leq j \leq m$

$$a_{i,j}/a_{i+1,j} = \alpha \;\; or \;\; a_{i,j}/a_{i+1,j} = 1/\alpha$$

Next, we will prove that in fact no such matrix exists.

Lemma 2. *No matrix* A *can derive both* $\hat{\mathbf{x}}$ *and* $\hat{\mathbf{y}}$.

Proof. Let $A \circ \mathbf{t} = \hat{\mathbf{x}}$, and $A \circ \mathbf{t}' = \hat{\mathbf{y}}$. By Lemma 1 and Corollary 1, without loss of generality (by renaming the columns of A), we may assume that $\exists k(1 \leq k < m)$ such that $\forall j(1 \leq j \leq k)$, $a_{0,j}/a_{1,j} = 1/\alpha$ and $t_j = 1$, and $\forall j(k < j \leq m)$, $a_{0,j}/a_{1,j} = \alpha$ and $t_j = 0$. Note that since each row of A sums to 1, we must have $1 \leq k < m$.

Among $\{1,\ldots,k\}$, we may further assume without loss of generality that $\exists \ell(0 \leq \ell < k)$, such that $\forall j(1 \leq j \leq \ell)$, $a_{1,j}/a_{2,j} = 1/\alpha$, and $\forall j(\ell < j \leq k)$, $a_{1,j}/a_{2,j} = \alpha$. (Here either range is guaranteed to be non-empty.)

Then,

$$\frac{\alpha}{\alpha+1} = \mathbf{a}_1 \circ \mathbf{t} = \frac{1}{\alpha}\sum_{j=1}^{\ell} a_{2,j} + \alpha \sum_{j=\ell+1}^{k} a_{2,j}$$

$$\frac{\alpha}{\alpha+1} = \mathbf{a}_2 \circ \mathbf{t} = \sum_{j=1}^{\ell} a_{2,j} + \sum_{j=\ell+1}^{k} a_{2,j}$$

It follows that

$$\sum_{j=1}^{\ell} a_{2,j} = \frac{\alpha^2}{(\alpha+1)^2}$$

From $A \circ \mathbf{t}' = \hat{\mathbf{y}}$, and

$$\mathbf{a}_0 \circ \mathbf{t}' = \frac{1}{\alpha}\mathbf{a}_1 \circ \mathbf{t}'$$

we have $\forall j > k$, $t'_j = 0$. This is because for any $j > k$, $a_{0,j} = \alpha a_{1,j} > a_{1,j}/\alpha$. From

$$\mathbf{a}_1 \circ \mathbf{t}' = \frac{1}{\alpha}\mathbf{a}_2 \circ \mathbf{t}'$$

we have $\forall j(\ell < j \leq k)$, $t'_j = 0$, by the same reasoning. Therefore,

$$\frac{\alpha}{\alpha+1} = \mathbf{a}_2 \circ \mathbf{t}' \leq \sum_{j=0}^{\ell} a_{2,j} = \frac{\alpha^2}{(\alpha+1)^2}$$

a contradiction.

Hence, there is no matrix A that can derive both $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ while guaranteeing differential privacy. □

Therefore, for the special case $n = 3$, $\theta_1 = 1$ and $\theta_2 = 2$, there is no universally utility maximizing mechanism that is count-oriented for count-range queries. For the general case, we can prove that there are consumers whose only optimal differentially private mechanisms, when expressed as vectors of length $n+1$, contain $\hat{\mathbf{x}}$, and $\hat{\mathbf{y}}$ respectively, as a subsequence. More precisely, let $k = (\theta_1 + \theta_2 - 1)/2$ if $\theta_1 + \theta_2$ is odd, and $k = (\theta_1 + \theta_2)/2$ otherwise.

Lemma 3. *There exists a consumer whose only optimal differentially private mechanism is:*

$$\mathbf{x} = (x_0, \ldots, x_n)^t$$

where for all i, $0 \leq i \leq k$, $x_i = 1/(\alpha^{k-1-i}(\alpha+1))$, *and for all* j, $k+1 \leq j \leq n$, $x_j = 1/(\alpha^{j-k-2}(\alpha+1))$, *and a consumer whose only optimal differentially private mechanism is:*

$$\mathbf{y} = (y_0, \ldots, y_n)^t$$

where for all i, $0 \le i \le k$, $y_i = 1/(\alpha^{k-i}(\alpha+1))$, and for all j, $k+1 \le j \le n$, $y_j = 1/(\alpha^{j-k-2}(\alpha+1))$.

By Lemma 3, in particular,

$$(x_{k-1}, x_k, x_{k+1}, x_{k+2})^t = \hat{\mathbf{x}}$$
$$(y_{k-1}, y_k, y_{k+1}, y_{k+2})^t = \hat{\mathbf{y}}$$

Therefore, if there is a matrix A that can derive both $\mathbf{x}$ and $\mathbf{y}$, then the submatrix $(\mathbf{a}_{k-1}, \mathbf{a}_k, \mathbf{a}_{k+1}, \mathbf{a}_{k+2})^t$ can derive both $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$, which is in contradiction to Lemma 2.

COROLLARY 2. *No matrix A satisfying differential privacy can derive both $\mathbf{x}$ and $\mathbf{y}$.*

3.2.3 Other Mechanisms

So far, we have only considered count-oriented mechanisms. For any arbitrary mechanism that may depend on the input database and not merely the count, we construct $n+1$ instances of databases, $\tau_0, \ldots, \tau_n$ where the count for a count-range query is i in the database τ_i, and τ_i, τ_{i+1} are neighboring databases. Therefore, we can still characterize the optimal mechanism, when restricted to these $n+1$ databases, for a consumer by a vector $(z_0, \ldots, z_n)^t$ where z_i is the probability of outputting *yes* when the underlying database is τ_i, and the mechanism as a matrix A of size $(n+1) \times m$ where the i^{th} row corresponds to the database τ_i, which is identical to that of count-oriented mechanisms. By differential privacy, $a_{i+1,j}/\alpha \le a_{i,j} \le \alpha a_{i+1,j}$. By Lemma 3, there are consumers whose only optimal mechanism is $\mathbf{x}$ and $\mathbf{y}$. By Corollary 2, no matrix can derive both $\mathbf{x}$ and $\mathbf{y}$. Therefore, there is no universally utility maximizing mechanism for count-range queries.

3.3 A Non-Existence Result

So far, we have proved that there is no universally utility maximizing mechanism for count-range queries. However, the requirements of a universally utility maximizing mechanism actually limit the behavior of a database: the database can only produce an intermediate noisy output, and then transform that intermediate noisy output for each consumer. Going beyond any such restriction, we can prove that no matter what mechanism is deployed by a database to produce the noisy results for multiple consumers, as long as that mechanism is differentially private, that mechanism can not maximize every consumer's utility.

THEOREM 3. *There is no differentially private mechanism that maximizes every consumer's utility for a count-range query.*

Proof Sketch: We start by considering a special case where $n = 3$, and $\theta_1 = 1$, $\theta_2 = 2$. We have already shown that there are two consumers whose only optimal differentially private mechanisms are:

$$\hat{\mathbf{x}} = (\frac{1}{\alpha+1}, \frac{\alpha}{\alpha+1}, \frac{\alpha}{\alpha+1}, \frac{1}{\alpha+1})^t$$
$$\hat{\mathbf{y}} = (\frac{1}{(\alpha+1)\alpha}, \frac{1}{\alpha+1}, \frac{\alpha}{\alpha+1}, \frac{1}{\alpha+1})^t$$

For ease of presentation, we shall define the consumer whose optimal mechanism is $\hat{\mathbf{x}}$ as the first consumer, and the other one as the second consumer. Then $\forall \tau \in D^n$, $i, j = 0, 1$, let $t_{\tau,i,j}$ be the probability of outputting i for the first consumer, and j for the second consumer. First, we prove the case where the mechanism is count-oriented. This will be generalized later. Therefore, we can characterize the output distribution for $\mu = 0, 1, 2$ by three 2×2 matrices:

	0	1	0	1	0	1
0	$t_{0,0,0}$	$t_{0,0,1}$	$t_{1,0,0}$	$t_{1,0,1}$	$t_{2,0,0}$	$t_{2,0,1}$
1	$t_{0,1,0}$	$t_{0,1,1}$	$t_{1,1,0}$	$t_{1,1,1}$	$t_{2,1,0}$	$t_{2,1,1}$

By the requirement of differential privacy, $\forall \mu = 0, 1$, and $i, j = 0, 1$, $t_{\mu+1,i,j}/\alpha \le t_{\mu,i,j} \le \alpha t_{\mu+1,i,j}$. If that mechanism maximizes both consumers' utility, then the marginal distribution in each matrix for each consumer constitutes her optimal mechanism. Let $t_{\mu,1,1} = t_\mu$, and thus, the mechanism is:

	0	1	0	1	0	1
0	$t_0 + \frac{\alpha-1}{\alpha}$,	$\frac{1}{(\alpha+1)\alpha} - t_0$	t_1,	$\frac{1}{\alpha+1} - t_1$	$t_2 - \frac{\alpha-1}{\alpha+1}$,	$\frac{\alpha}{\alpha+1} - t_2$
1	$\frac{1}{\alpha+1} - t_0$,	t_0	$\frac{\alpha}{\alpha+1} - t_1$,	t_1	$\frac{\alpha}{\alpha+1} - t_2$,	t_2

By differential privacy, $t_1 \le \alpha t_0$, and

$$\frac{1}{\alpha+1} - t_1 \le \alpha(\frac{1}{\alpha(\alpha+1)} - t_0)$$

Thus, $\alpha t_0 \le t_1$. Therefore $t_1 = \alpha t_0$. Similarly, we can prove that $t_2 = \alpha t_1$. By differential privacy,

$$t_0 + \frac{\alpha - 1}{\alpha} \le \alpha t_1 = \alpha^2 t_0$$

Then, $t_0 \ge 1/(\alpha(\alpha+1))$. By differential privacy,

$$\frac{\alpha}{\alpha+1} - t_1 \le \alpha(\frac{\alpha}{\alpha+1} - t_2)$$

Since $t_2 = \alpha^2 t_0$ and $t_1 = \alpha t_0$, $t_0 \le 1/(\alpha+1)^2$. We have thus obtained a contradiction. Therefore, no such mechanism maximizes both consumers' utility when $n = 3$, $\theta_1 = 1$ and $\theta_2 = 2$.

For the more general case, the proof is similar to that of Corollary 2 and Theorem 2, and we omit the details here. □

As discussed in [3], the universally utility maximizing mechanism only exists for a limited class of queries. Brenner et al. characterized the necessary conditions on the queries that admit universally utility maximizing mechanism. The basic idea of their proof is to characterize a query by an undirected graph where each vertex corresponds to an output of the query, and an edge is drawn between two vertices if the addition/deletion of a tuple to/from a database can result in such a change in the outputs. Brenner et al. proved that if there is a cycle in the privacy constraint graph, then no universally utility maximizing mechanism exists. However, for count-range queries there is no cycle on the privacy constraint graph, hence Brenner's result cannot be used to prove our result.

3.4 An Approximate Mechanism

Given that there is no optimal mechanism, we turn to consider approximate mechanisms. First, we formulate the notion of *β-approximate universally utility maximizing*, which measures the approximation ratio in terms of the weighted error.

DEFINITION 7. *(β-approximate universally utility maximizing): A differentially private mechanism X is* β-approximate universally utility maximizing *if and only if for any consumer, there exists a differentially private mechanism that is derivable from X whose weighted error for that consumer is at most β times of the minimal weighted error of that consumer.*

We will prove that the range-restricted geometric mechanism is 2-approximate universally utility maximizing for count-range queries.

THEOREM 4. *The range-restricted geometric mechanism* (6) *is* 2*-approximate universally utility maximizing for count-range queries.*

We consider the range-restricted geometric mechanism as an approximation is because of the following observation: if there were no privacy concern, then the identity matrix I would be a trivial universally utility maximizing mechanism for count-range queries. An identity matrix I means that given a count i, the mechanism always outputs i. In other words, the output probability distribution is 1 at i and 0 everywhere. However, by the requirement of differential privacy, that probability distribution needs to be "flattenned" such that the probability mass of outputting i is assigned to other outputs. Intuitively, after that reassignment, given an output i, the most likely output should still be i, and the probability of outputting j decreases with the increasing in $|j - i|$. The range-restricted geometric mechanism exactly reflects that intuition.

4. OPTIMAL DIFF. PRIVATE MECHANISMS FOR THRESHOLD QUERIES

In this section, we consider threshold queries, a special case of count-range queries which test whether or not the number of rows in a database satisfying a predicate is less/greater than a threshold. More precisely, a threshold query can be characterized by either $\langle p, 0, \theta\rangle$ or $\langle p, \theta, +\infty\rangle$. We first revisit the problem of designing an optimal mechanism for a single consumer, now for the special case of threshold queries. It turns out that a simpler mechanism is possible for threshold queries than for count-range queries, and this simpler mechanism will be useful in our search for an optimal mechanism for multiple consumers for threshold queries. We show that, unlike the case for count-range queries, in the case when multiple consumers ask threshold queries with the same predicate (with possibly different thresholds), there exists a mechanism that simultaneously maximizes every consumer's utility while guaranteeing differential privacy.

4.1 An Optimal Diff. Private Mechanism

As was the case for count-range queries, a straightforward way to find an optimal legal function for a consumer asking a threshold query is to treat each $\phi(i)$, $0 \le i \le n$, as a variable, and to solve the linear programming problem that minimizes her weighted error subject to the requirements of a legal function. This amounts to solving an optimization problem of $n + 1$ variables. However, again, we will prove that for the design of an optimal legal function, it suffices to solve an optimization problem with a single variable. First, we prove a theorem about the existence of an optimal legal function of a particular form.

THEOREM 5. *There exists an optimal legal function ϕ^* for a threshold query $\langle p, \theta, +\infty\rangle$ that satisfies the recurrence relation in* (2)*.*

By Theorem 5, when searching for an optimal legal function, it suffices to consider legal functions satisfying (2). As discussed in Section 3, ϕ^* is well-defined if we fix $\phi^*(0) = \beta$. Hence, the weighed error of ϕ can be written as $err(\phi) = err(\beta)$, where $err(\beta)$ is a piecewise linear function in β. To compute the minimum of $err(\beta)$, we can compute the minimum of $err(\beta)$ on each subinterval, which is trivial, and then compare those local minima to get the global minimum. Let that global minimum be $err(\beta^*)$. Then, an optimal legal function ϕ^* is well-defined where $\phi^*(0) = \beta^*$ and ϕ^* satisfies (2).

We can also prove the existence of a particular form of optimal legal functions for a threshold query $\langle p, 0, \theta\rangle$, which is symmetric to Theorem 5.

THEOREM 6. *There exists an optimal legal function ϕ^* for the query $\langle p, 0, \theta\rangle$ that satisfies the recurrence relation in* (3)*.*

4.2 Multiple Consumers

Next, we consider the problem of serving multiple consumers asking the same threshold query. We want to know if there exists a universally utility maximizing mechanism for threshold queries. Surprisingly, unlike the case for count-range queries, we can prove that the range-restricted geometric mechanism is a such mechanism. This is shown in Theorem 7.

THEOREM 7. *The range-restricted geometric mechanism in* (6) *is a universally utility maximizing differentially private mechanism for threshold queries.*

Proof Sketch: We can characterize a range-restricted geometric mechanism by a symmetric matrix M of size $(n + 1) \times (n + 1)$ whose element $m_{i,j} = \Pr[Z(i) = j]$.

$$M=\frac{\alpha-1}{\alpha+1}\begin{pmatrix} \frac{\alpha}{\alpha-1}\cdot 1 & \alpha^{-1} & \alpha^{-2} & \cdots & \frac{\alpha}{\alpha-1}\cdot\alpha^{-n} \\ \frac{\alpha}{\alpha-1}\cdot\alpha^{-1} & 1 & \alpha^{-1} & \cdots & \frac{\alpha}{\alpha-1}\cdot\alpha^{-n+1} \\ \frac{\alpha}{\alpha-1}\cdot\alpha^{-2} & \alpha^{-1} & 1 & \cdots & \frac{\alpha}{\alpha-1}\cdot\alpha^{-n+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\alpha}{\alpha-1}\cdot\alpha^{-n} & \alpha^{-n+1} & \alpha^{-n+2} & \cdots & \frac{\alpha}{\alpha-1}\cdot 1 \end{pmatrix}$$

We can characterize a consumer's optimal differentially private mechanism for threshold queries $\langle p, \theta, +\infty\rangle$ by a vector $\mathbf{z} = (z_0, \ldots, z_n)^t$ satisfying Theorem 5. We can show that M is invertible. Let $\mathbf{t} = M^{-1} \circ \mathbf{z} = (t_0, \ldots, t_n)^t$. Then it suffices to prove that $\mathbf{t}$ is a transformation satisfying Definition 4. Let M_i be the matrix obtained from M by replacing the i^{th} column of M by $\mathbf{z}$. By Cramer's rule, $t_i = \det(M_i)/\det(M)$.

By Theorem 5, for all i, $0 \leq i < n$, $z_{i+1} = \min\{\alpha z_i, 1 - (1 - z_i)/\alpha\}$. We will prove that for all j, $0 \leq j \leq n$, $0 \leq \det(M_j)/\det(M) \leq 1$. Therefore, $\mathbf{t} = (t_0, t_1, \ldots, t_n)^t$ is a transformation. We can prove the same results for threshold queries $\langle p, 0, \theta\rangle$ in a similar way, and we omit the details here. □

The universally utility maximizing differentially private mechanism was first studied for count queries in [9, 10], where only "oblivious" mechanisms were considered. A mechanism is *oblivious* if it sets up an identical distribution over outputs for every pair of databases that has the same unperturbed query result. Naturally, an implementation of an oblivious mechanism only needs to have access to the true query result — the input — and can be oblivious to the database itself. The range-restricted geometric mechanism only depends on the result of a count query instead of the database itself, and thus, it is an oblivious mechanism.

In contrast to previous work, the differentially private mechanisms we are considering in this paper are non-oblivious mechanisms because the true query result of a threshold query is either *yes* or *no*, whereas the mechanisms we have proposed rely on the count of a threshold query rather than just *yes* or *no*. Of course, there are also oblivious mechanisms for threshold queries. We can use a function Φ to characterize the oblivious mechanisms for threshold queries where β_1 (β_2) is the probability of outputting *yes* when the correct answer is *no* (*yes*).

$$\Phi(\mu) = \begin{cases} \beta_1 & \text{if } 0 \leq \mu < \theta \\ \beta_2 & \text{if } \theta \leq \mu \leq n \end{cases}$$

When $0 \leq \beta_2/\alpha \leq \beta_1 \leq \alpha\beta_2 \leq 1$ and $(1-\beta_2)/\alpha \leq (1-\beta_1) \leq \alpha(1-\beta_2)$, it is easy to verify that Φ is a legal function. However, we can show that any legal function Φ is not an optimal legal function unless $\beta_1 = \beta_2 = 1$ or $\beta_1 = \beta_2 = 0$: we construct a function ϕ satisfying (2), and $\phi(\theta) = \beta_2$. It is not difficult to see that ϕ is less likely to commit both types of errors for a threshold query unless $\beta_1 = \beta_2 = 1$ or $\beta_1 = \beta_2 = 0$. Thus, an oblivious differentially private mechanism for threshold queries is not optimal in a general sense.

By Theorem 7, the database utilizes the range-restricted geometric mechanism to perturb the count of a threshold query only once, and stores that noisy count. For each consumer asking the same threshold query, the database randomly transforms the stored noisy count to *yes* or *no* using the transformation which maximizes that consumer's utility. For consumers asking threshold queries with the same predicate but different thresholds, the database still only needs to perturb the count once since the counts for those queries are the same. Note that Theorem 5 is independent of the threshold θ, and thus, for each consumer, there is an optimal legal function satisfying (2). By Theorem 7, there is a transformation for that consumer such that the induced differentially private mechanism of the range-restricted geometric mechanism and that transformation guarantees optimal utility for that consumer. Therefore, the range-restricted geometric mechanism also simultaneously guarantees optimal utility for all consumers asking threshold queries with the same predicate but different thresholds.

COROLLARY 3. *For consumers asking threshold queries with the same predicate but different thresholds, there is a transformation for each consumer such that the induced differentially private mechanism of the range-restricted geometric mechanism and that transformation guarantees optimal utility for that consumer.*

The range-restricted geometric mechanism also simultaneously maximizes utility for different privacy levels. We refer interested readers to [10] for a complete and precise description of that property.

5. RELATED WORK

The notion of differential privacy was proposed by Dwork et al. in [5]. The same authors also proposed the addition of Laplacian noise to guarantee differential privacy [7] for count queries. McSherry et al. proposed a universal differentially private mechanism for general queries in [15]; see [6] for a recent survey of privacy.

Dinur and Nissim [4] are pioneers in establishing the upper bounds on the number of queries that can be answered with reasonable accuracy. Count queries [4, 8], and more general queries [16, 7, 2] have been studied from that perspective. Recently, Hardt and Talwar [11] gave tight upper and lower bounds on the amount of noise needed to ensure differential privacy for a given number of linear queries. Hay et al. and Li et al. [12, 14] both proposed exploiting consistency constraints to increase accuracy when answering multiple queries.

Ghosh et al. [9] were the first to formally define a universally utility maximizing differentially private mechanism that simultaneously maximizes every consumer's utility for count queries. Their results indicate that the range-restricted geometric mechanism is a universally utility maximizing differentially private mechanism for a single count query such that every consumer can combine her own information and utility function in a way that maximizes her utility, and that transformation is effectively enough to result in an optimal mechanism. Gupte et al. proved a similar result in [10] for a different utility model. Our work extends that idea to threshold queries. Brenner [3], shows that the universally utility maximizing mechanism only exists for a limited class of queries, and gives a criterion that partially characterizes when they do not exist. In our work we give an example of a class of queries (count-range queries) for which no universally utility maximizing mechanism exists that is not covered by Brenner's criterion.

6. CONCLUSION

In this paper, we propose an optimal differentially private mechanism for count-range queries. However, when considering serving multiple consumers, in contrary to previous positive results for count queries, we prove that for count-range queries there is no differentially private utility maximizing mechanism that guarantees optimal utility for every consumer. Despite this negative result, we prove that the range-restricted geometric mechanism is a 2-approximate universally utility maximizing for count-range queries. Furthermore, we show that for threshold queries (a natural restriction on count-range queries), a universally utility maximizing differentially private mechanism that simultaneously maximizes every information consumer's utility does exist.

The optimal mechanisms for both threshold queries and count-range queries we have proposed are non-oblivious, in that they take a count as the input instead of *yes* or *no*. It would be interesting to investigate non-oblivious optimal differentially private mechanisms for other classes of queries. An application of our results to differentially private frequent itemset mining is also an interesting direction for future research, as determining whether an itemset is frequent is akin to answering a threshold query.

7. REFERENCES

[1] `http://pages.cs.wisc.edu/~zeng/icdt_submission.pdf`.

[2] 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 '08.

[3] H. Brenner and K. Nissim. Impossibility of differentially private universally optimal mechanisms. *Foundations of Computer Science, Annual IEEE Symposium on*, 2010.

[4] I. Dinur and K. Nissim. Revealing information while preserving privacy. In *In PODS*, 2003.

[5] C. Dwork. Differential privacy. In *in ICALP*, pages 1–12. Springer, 2006.

[6] C. Dwork. Differential privacy: a survey of results. In *Proceedings of the 5th international conference on Theory and applications of models of computation*, TAMC'08, 2008.

[7] C. Dwork, F. Mcsherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference*, 2006.

[8] C. Dwork and K. Nissim. Privacy-preserving datamining on vertically partitioned databases. In *In CRYPTO*, 2004.

[9] A. Ghosh, T. Roughgarden, and M. Sundararajan. Universally utility-maximizing privacy mechanisms. In *Proceedings of the 41st annual ACM symposium on Theory of computing*, STOC '09, 2009.

[10] M. Gupte and M. Sundararajan. Universally optimal privacy mechanisms for minimax agents. In *Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems of data*, PODS '10, 2010.

[11] M. Hardt and K. Talwar. On the geometry of differential privacy. In *Proceedings of the 42nd ACM symposium on Theory of computing*, 2010.

[12] M. Hay, V. Rastogi, G. Miklau, and D. Suciu. Boosting the accuracy of differentially-private queries through consistency. In *VLDB*, 2010.

[13] D. Kifer and B.-R. Lin. Towards an axiomatization of statistical privacy and utility. In *Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, PODS '10, 2010.

[14] C. Li, M. Hay, V. Rastogi, G. Miklau, and A. McGregor. Optimizing histogram queries under differential privacy. In *PODS*, 2010.

[15] F. McSherry and K. Talwar. Mechanism design via differential privacy. In *FOCS '07: Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, 2007.

[16] K. Nissim, S. Raskhodnikova, and A. Smith. Smooth sensitivity and sampling in private data analysis. In *STOC '07: Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, 2007.