

# Differential Privacy for Sum Queries without External Noise<sup>\* †</sup>

Yitao Duan  
NetEase Youdao R&D  
Beijing, China  
duan@rd.netease.com

## ABSTRACT

We consider privacy issues in statistical database and data mining where queries are executed on data collected from a large number of individuals. It is generally established that a strong notion of privacy is guaranteed if the results are perturbed by random noise with sufficient variance (e.g., [5, 27, 44]). In this paper, we point out a vulnerability in such an approach and show that for some types of queries, when the dataset is sufficiently large, the inherent uncertainty associated with unknown quantities is enough to provide similar perturbation and the same privacy can be obtained *without* external noise. One type of such queries is sum queries which aggregate across all records. This is a surprisingly general primitive supporting various data mining algorithms, including many non-linear ones such as SVD, PCA,  $k$ -means, ID3, SVM, EM, and all the algorithms in the statistical query model. We derive privacy conditions for sum queries and, for the first time, provide mathematical proof for the intuition that aggregates across a large number of individuals is private using a widely accepted notion of privacy. Our results are also relevant in query auditing and we show how they can be used to construct simulatable query auditing algorithms that handle online and offline auditing in a uniform way with stronger privacy than ever achieved before.

## Categories and Subject Descriptors

H.2.0 [Information Systems]: Database Management—*security, integrity and protection*; G.3 [Mathematics of Computing]: Probability and Statistics—*statistical computing*

## General Terms

Algorithms, Security and Theory

## Keywords

differential privacy, sum queries, simulatable query auditing

## 1. INTRODUCTION

We consider privacy issues in statistical database and data mining where queries or computation are executed on data

<sup>\*</sup>Part of the work was performed when the author was a Ph.D. student at Computer Science Division, University of California, Berkeley.

<sup>†</sup>Preliminary version of parts of this work appeared in the proceedings of The 18th ACM Conference on Information and Knowledge Management (CIKM '09) as [18].

collected from a large number of individuals. This model can be found in an increasing number of real-world applications ranging from e-commerce to medical research and it offers enormous potential social benefits. The goal of such computation is to discover statistical patterns and the major challenge is to release such aggregate information while preserving the privacy of the individuals.

There is a vast body of relevant work. Statistical database privacy has been extensively studied since the 1970's. The early results were mixed and typically not rigorous, mostly due to the lack of a general notion of privacy. For example, some (e.g., [10, 38]) only consider full disclosure as compromise which is apparently too weak by today's standard. Cryptography provides primitives with provable privacy and various level of efficiency [33, 13, 7]. These tools have been used to construct privacy-preserving data mining schemes (e.g., [39, 17, 49, 52, 50]). The privacy in cryptography is rigorous but the protection does not cover the final results, i.e., cryptographic schemes only guarantee that no information beyond what is implied by the final results is leaked. Great efforts have been made to analyze the actual leakage by published information and devise schemes to maintain adequate privacy. The main body of the work falls into two categories: output perturbation and query auditing. The former adds to each query response some random noise while the latter examines the queries and deny those that are deemed privacy-breaching (the others are answered accurately). Both are being actively pursued but they use different notions of privacy. A newly emerging line of work in the output perturbation approach [15, 5, 27, 25, 23, 44, 2, 41, 6, 40] strive to provide privacy definition and protections as rigorous as those in cryptography. Current results state that query results need to be perturbed by random noise with sufficient variance in order to maintain privacy.

In this paper we show that noise is not essential for privacy. On one hand, the additive noise approach has a vulnerability that allows for easy exploitation in a real-world deployment. On the other hand, for some types of queries, the inherent randomness associated with unknown quantities is enough to provide similar perturbation and the same notion of privacy can be obtained *without* external noise. One example is sum queries that aggregate across many records. As [44] pointed out, it is widely believed in traditional research that, since data mining algorithms are designed to reveal only "global" information, it is safe to apply them "as they are" to sensitive data and publish the results. In fact such belief is in effect in many practical systems in the real-world (e.g., recommendation and voting systems etc.).

However, there is currently no formal substantiation of such an assessment. On the other hand, in probability theory, there are some established results on asymptotic behavior of aggregates of  $n$  random variables. Many of them state that, under some (usually very mild) conditions, when  $n$  is sufficiently large, the aggregates converge in some way to a distribution independent of the individual samples except for a few distribution parameters. This leads to an intuitive conjecture that aggregates are privacy-preserving. Our work is the first to make connection between these results and a concrete privacy definition and *prove* this conjecture. Besides the theoretical significance for furthering our understanding on privacy, our results also have practical implications: as collecting and processing large amount of data is becoming a reality, the asymptotic state is not so far away and the conditions derived in this paper can be applied. This provides practitioners with concrete guidelines for verifying whether their practices indeed preserve privacy. Even the negative cases where the conditions do not apply or are violated are significant too: This means that some practices that are in use everyday do not have provable privacy and people should exercise caution.

Our results are also relevant in distributed data mining and secure multiparty computation. By showing that releasing certain information accurately does not cause privacy breach, one could afford to reveal some intermediate results. This could significantly reduce the complexity and cost of the computation protocol thus a solution that is practical at realistically large scale can possibly be obtained. In particular, our results close a gap in a previously proposed practical data analysis framework that used vector addition to perform the computation of many useful algorithms such as SVD, factor analysis and link analysis etc. [7, 8, 21, 20, 19]. These works all treated the intermediate sums that are produced by the iterative methods as public. This leaves open the question whether these protocols provide enough privacy protection. Using our results, we can answer the question in the affirmative provided certain conditions are met. This shows that the previous works are valid and constitute valuable practical privacy solutions for their applications.

## 2. PRELIMINARIES

Let  $D$  be an arbitrary domain (e.g., real numbers, text, boolean, etc., or a mixture of them). A statistical database is modeled as a vector  $d \in D^n$ . We use  $d_i$  to denote the  $i$ th entry (also called row, or record) of  $d$ . We assume the data records are drawn i.i.d. from  $D$  according to some distribution  $\mathcal{D}$ . Consistent with previous work, the distribution  $\mathcal{D}$  is assumed to be public [42, 37, 6]. Each row contains information about an individual such as salary, weight, purchase history etc. The hamming distance  $H(d, d')$  between two databases  $d, d'$  is the number of entries on which they differ.

### 2.1 Sum Queries

We consider queries in the following form

$$f(d) = \sum_{i=1}^n g(d_i)$$

where  $g(d_i) = [g_1(d_i), g_2(d_i), \dots, g_m(d_i)]^T$  and  $g_j : D \rightarrow [0, 1]$ . In other words, the function is the sum of  $n$   $m$ -dimensional vectors, one computed for each data record.  $g_j$

is also called the  $j$ th query and  $m$  is the maximum number of queries allowed for the lifetime of the database.

This simple form is a surprisingly powerful tool for computing a large number of popular statistical analysis algorithms.<sup>1</sup> The standard algorithms use gradient steps which sum vector data from the users. These steps are *linear* in per user data and can be implemented using the sum query model. Implementing the algorithms using summation forms has been used by other work as a general approach to parallelize the algorithms. For example, [11, 14] showed that many popular algorithms have summation implementations that can be computed with Google's MapReduce framework over clusters. The examples included an EM algorithm for pLSI, Locally Weighted Linear Regression (LWLR), Naive Bayes (NB), PCA, etc., and all the algorithms in the statistical query model [36]. They demonstrated the versatility of summation in implementing statistical learning algorithms. This phenomena has also been observed by the privacy community and used as a way to implement private algorithms. These works include private SVD [7], factor analysis [8], link analysis [21], and [5].

As a side note, the model of addition-based computation also has great practical advantage in distributed data mining because addition has efficient private implementation (as opposed to multiplication) using cryptographic tools [12, 33, 13, 20]. It also admits some extremely efficient zero-knowledge tools that can be used to verify user input and computation [20] which is essential in real-world applications. Thus securing this model has profound practical implications.

### 2.2 Differential Privacy

There are several formalizations of privacy developed for private data analysis over the years (see e.g., [1, 31, 32]) but so far the strongest achievable privacy is *differential privacy*, introduced in [27], further refined by [25, 23], and adopted by many latest works such as [2, 44, 41, 6]. Intuitively, a mechanism is private if it ensures that the risk to one's privacy should not substantially increase as a result of participating in a statistical database. Differential privacy captures this intuition and is defined as

DEFINITION 1 (DIFFERENTIAL PRIVACY [27, 25]).  $\forall \epsilon, \delta \geq 0$ , an algorithm  $\mathcal{A}^f$  gives  $(\epsilon, \delta)$ -differential privacy with respect to a query function  $f$  if for all  $S \subseteq \text{Range}(\mathcal{A}^f)$ , for all  $d, d' \in D^n$  such that  $H(d, d') = 1$

$$\Pr[\mathcal{A}^f(d) \in S] \leq \exp(\epsilon) \Pr[\mathcal{A}^f(d') \in S] + \delta$$

The definition ensures that with a differentially private access mechanism the inclusion or exclusion of a single record does not change the output probability by more than some small amount. Moreover, the formalization is agnostic to the adversary's auxiliary information. It provides strong guarantee for the privacy of individual data record and is the privacy notion we adopt in this paper.

### 2.3 Adversary Model

In [24], auxiliary information is modeled as some function of the database (and a description of its distribution). In other words, it models what the adversary knows. In contrast, we model the adversary with what is *unknown* to it. By explicitly modeling unknownness, we are able to tap

<sup>1</sup>It is also the form studied in [5].

into the inherent randomness within the data and provide adequate protection without external noise:

*Given a database  $d$  with  $n$  records drawn i.i.d. from some distribution  $\mathcal{D}$ , there are sufficiently large number of records about which the adversary possesses no additional information besides  $\mathcal{D}$ . We call such records intact.*

A few remarks are in order. This model is only slightly weaker than some existing schemes and equivalent to many others. This can be seen from a few aspects:

1. We make no assumption about distribution  $\mathcal{D}$  other than that it is public, which is consistent with many other works such as [42, 37, 6], and satisfies some mild requirements which are made precise in theorem 2. The privacy depends on the asymptotic property of the sums when the dataset is large, which is guaranteed by CLT, not the distribution of individual records.
2. The model requires a substantial number of records be concealed from the adversary.<sup>2</sup> Although not as attractive as a fully informed adversary model as in e.g., [24, 28, 2, 44, 41], it is realistic in many applications. Any large data sets, if collected and stored in a secure manner, satisfy this condition. Examples include voting and census data that only publish aggregate statistics. Due to its huge volume and tight security, it is difficult for an adversary to compromise all but a small number of records (probably by attacking a few other data sources). This is also true in a distributed setting where each data record resides at each user, as in the case of distributed data mining considered in works such as [20]. From another perspective, many algorithms (e.g., voting) cannot tolerate a large fraction of compromised users. This condition is a prerequisite for them to be meaningful in the first place. In addition, works in query auditing that consider probabilistic compromise such as [42, 37] are all in this model as well since they all treat the data as random variables.
3. The model allows other records to be compromised in an *arbitrary* way. Interestingly, these hidden records also protect those compromised ones (in a differential privacy sense) as the protection is measured with the change of output probabilities which is dependent solely on the unknown quantities. Attacks such as [43] that exploit the correlations between attributes within a data record does not work as our privacy conditions detect such efforts in a uniform way (section 6).

In the following, to simplify discussion, we will not explicitly mention compromised records and pretend that all  $n$  are intact. Since the results are independent of the compromised records, it is trivial to generalize them to the real setting.

### 3. RELATED WORK

Several solutions achieve differential privacy [24, 25, 28, 2, 44, 41, 6, 51, 40, 26]. Most of them are based on perturbing the response with additive noise. The initial work [27] used Laplace noise and [25] extended the results to gaussian and binomial. [6, 41, 9] do not use additive noise directly

<sup>2</sup>We defer the discussion of “how many are enough” until section 7.1.

on the results but still perturb the algorithms with external randomness in some way. For example, [9] is a logistic regression scheme with differential privacy that, instead of perturbing the output, perturbs the objective function.

Since the initial works [15, 29, 5], much effort has been made on reducing the amount of noise necessary for privacy, reflecting the need to obtain more accurate statistics. The most successful works are based on calibrating noise to the *sensitivity* of the query function [27] and its non-uniformity on the data instance [44]. For a query function  $f : D^n \rightarrow \mathbb{R}^m$ , the L1-sensitivity of  $f$  is defined as [27]:

$$S(f) = \max_{d, d': H(d, d')=1} \|f(d) - f(d')\|_1$$

where  $\|\cdot\|_1$  denote the L1-norm of a vector. Informally, the main results in [27] is that the access mechanism can have  $(\epsilon, 0)$ -differential privacy if it answers the query with  $f(d) + [Y_1, \dots, Y_m]^T$  where  $Y_j$ 's are drawn i.i.d. from Laplace distribution whose density function is  $h(y) \propto \exp\{-\frac{\|y\|_1}{S(f)/\epsilon}\}$ . Using gaussian or binomial noise is shown to have non-zero  $\delta$  [25]. However, [51] shows that determining the L1-sensitivity for unrestricted queries is NP-hard, so is verifying differential privacy. The applicability of differential privacy in this setting is thus very limited. In addition, there have been a series of work that established lower bounds [15, 28, 27] on the usefulness of such mechanisms, namely they can only answer a sub-linear number of queries on any database during its *lifetime* otherwise the data can be reconstructed [15]. [6] circumvents this limitation by allowing the mechanism to be useful only for a restricted class of learning tasks.

On the other hand, works in query auditing have some positive results. They show that some level of privacy can be maintained even if some queries are answered accurately (e.g., [42, 37]). Although different notions of privacy is used, their work hints that noise may not be essential for privacy.

Given the finding of [51], our strategy is similar to that of [6], i.e., we improve the situation by considering a more restrictive setting. Instead of circumventing the query number limit, our goal is to eliminate noise. And the only restrictions we place is on the adversary's auxiliary information and the type of queries it can pose.

A preliminary version of this work appeared in CIKM '09 as a short paper [18]. Here we provide detailed discussion and full proofs of the theorems as well as additional results. In addition, we also illustrate how our results can be applied to detecting unsafe queries using concrete examples.

## 4. PRIVACY AND NOISE

The idea of achieving privacy by adding noise is natural and intuitive. However, additive noise is neither effective nor necessary for privacy. There is a serious vulnerability with all response perturbation solutions: namely limiting the number of queries, a crucial means for them to prevent leakage, cannot be enforced when there are shared items between multiple databases. We show in section 7.2 that our approach, although still subject to the query limit restriction and not completely immune to the same vulnerability, is more secure since it uses a stronger condition and raises the cost for an adversary to obtain the same amount of information. On the other hand, there are “safe” functions that can be published accurately without sacrificing privacy.

### 4.1 Safe Functions

In all the schemes in [15, 29, 5, 27, 44], the security relies solely on the noise-induced randomness. This means the bound on the probability increase holds as long as the coins are secure, even if the *database* is fully disclosed. This type of protection is analogous to semantic security in encryption [34, 24]. However, there are two discrepancies: (1) the schemes are not protecting the data records but the query responses; (2) the protection is inadequate in an open environment. As an evidence for (1), we can prove the following:

LEMMA 1. *For any privacy mechanism in [15, 29, 5, 27, 44], if the mechanism is  $(\epsilon, \delta)$ -private, then for any  $t, t' \in \text{Range}(f)$  such that  $\|t - t'\|_1 \leq S(f)$  and for any noisy response  $\tau$  output by the mechanism*

$$\Pr[f(d) = t] \leq \exp(\epsilon) \Pr[f(d) = t'] + \delta$$

PROOF. (Sketch) Let  $Y$  be the random variable representing the noise specified by the mechanism. As  $\tau = f(d) + Y$ , it suffices to show that

$$\frac{\Pr[f(d) = t]}{\Pr[f(d) = t']} = \frac{\Pr[Y = \tau - t]}{\Pr[Y = \tau - t']}$$

and noticing that  $\|t - t'\|_1 \leq S(f)$ . The result can be derived from the properties of the random quantity  $Y$ , be it gaussian or laplace, as is done in [15, 29, 5, 27, 44].  $\square$

In other words, these mechanisms achieve differential privacy by bounding the distinguishability between any two close values of  $f(d)$ .<sup>3</sup> Note that this is not equivalent to perturbing each individual record since  $f$  can be non-linear. The discrepancy is, privacy is about the secrecy of individual records while the mechanisms are protecting a (possibly aggregate or even one-way) function of the database. There are deterministic functions that reveal very little, or none at all, information about individual records as long as there are sufficient number of intact ones. Consider the following example: Suppose  $d_1, \dots, d_n$  are uniformly distributed in  $\mathbb{Z}_q$  for some prime  $q$ . Then  $f(d) = \sum_{i=1}^n d_i \pmod q$  is such a safe function. One can prove that, information-theoretically,  $f$  contains no information about any  $d_i$ , provided that there are at least two intact records. Releasing it without adding noise is perfectly safe. Yet this phenomenon cannot be captured by the notion of sensitivity: here  $S(f) = q - 1$ .

Another indication that noise is not essential for privacy comes from the response perturbation solutions themselves. In all these schemes [15, 29, 5, 27, 44], conditional on the number of queries, the amount of noise for maintaining adequate privacy is *independent* of the size of the database  $n$ . This implies that noise is not necessary for large  $n$  since one could get quite accurate estimates in the presence of constant noise when the sample size is large. In this case the function itself is insensitive to changes to a single record.

## 4.2 Multi-Database Vulnerability

A more serious problem with response perturbation solutions is that they fail to provide adequate protection. In

<sup>3</sup>Note that this lemma does not contradict the results established by previous works that some utility could still be obtained from the noisy responses. This is because, although the noisy responses do not allow one to distinguish between two possible true answers, they do provide approximations. Putting it another way, lemma 1 bounds the distinguishability of any two points within a hypercube with each side bounded by  $S(f)$ , but the noisy responses give an approximate position of the hypercube.

such a mechanism, as the randomness in the case of semantic security is independently generated for each encryption, the noise is independently generated for each query. However, the two differ by the way they blend the randomness with the plaintext (data) to produce ciphertext (noisy response): The database access mechanisms use simple addition which is a *linear* operation. And since the noise is zero-mean and is drawn from the same distribution independently, its effectiveness maybe canceled by posing related queries. A simple trick is to pose the same query multiple times, as in the case of a Dinur-Nissim style attack [15]. One could try to use the same noise for the same query to mitigate this problem. Even if this is feasible, which is hard to establish as there could be queries that are syntactically different but semantically equivalent, it is problematic: (1) This already deviates from the principles of semantic security which by definition stipulates probabilistic output. (2) It is not effective. Due to the linearity, the adversary does not need to use identical queries. It can for example asks for  $d_1 \pm d_i$  for a number of  $i$ 's to achieve the same noise reduction.

[15, 29, 5, 27, 44] handle this issue by restricting the total number of queries that can be made for the life time of the database. However, in reality, the same data record may appear in multiple databases, each administrated by an autonomous entity. An adversary could query these databases about the same data record. It is impossible for one database to adjust the noise level and the query bound to account for possible leakage by *other* databases. Although the queries obtained from one database are within a safe limit, by pooling all the queries from different databases together, the number could exceed the threshold and allow the adversary to get substantial information about the record. And such a threat is real: It is very common nowadays for a user to have profiles in many online vendors and the above attack can be easily mounted if the vendors allow statistical queries with the privacy mechanisms in [15, 29, 5, 27, 44].

In summary, restricting the number of queries to prevent Dinur-Nissim style attack [15] is impossible to enforce in real deployment. This demonstrates that relying on noise alone is not effective in protecting privacy. Different and/or additional means, such as restricting the queries as is done in query auditing [42, 37] or enforcing other conditions such as presented in this paper, should be considered. We show in section 7.2 that our condition is actually *stronger* than additive noise thus is more secure against this attack.

**Database Privacy  $\neq$  Encryption** The above vulnerability shows that, although following a similar model, response perturbation mechanisms fail to provide the same level of protection as semantic security. The reason is that the two have very different settings. In encryption, there is a decryption key that separates a recipient from the adversary so the ciphertext could be made to provide *zero* utility to the adversary while providing full utility to the recipient. In database, however, there is no such separation and the goal is actually to provide *maximum* utility to the user/adversary while maintaining acceptable privacy. As a result, the a priori knowledge of an adversary about the response is very different. In cryptography, the adversary does not know a priori whether the sender is transmitting the same messages so a deterministic encryption which allows the adversary to infer such information is insecure. In database privacy, on the other hand, an output is determined by both the data and the *queries*. The adversary possesses complete a priori

knowledge about the latter. Some relations of the queries (e.g., identical queries) are *independent* of the data. Disclosing such relations in the response is *not* a leakage. Giving different noisy answers to identical queries, as the response perturbation solutions do, is equivalent to providing multiple samples of the same random variable and each sample leaks some information. This randomized response feature, which is an intrinsic property of semantic security, actually hurts database privacy.

### 4.3 Sources of Randomness

It has been shown that a semantic security-like privacy (i.e., one that compares the adversary’s prior and posterior view) against arbitrary auxiliary information is impossible to achieve [24]. Differential privacy is a relaxation (it shifts to comparing the risk to an individual when participating in the database versus not). Under this definition, existing solutions achieve security against arbitrary auxiliary information. Another reasonable relaxation is to restrict the adversary’s auxiliary information. As we mentioned in section 2.3, in many situations the adversary is unlikely to obtain full knowledge of the database and there are substantially large number of intact records. This allows us to tap into another source of randomness that has not been utilized by existing works in differential privacy (but has been used by many other works such as [42, 37]). The benefit is increased accuracy.

Fundamentally, privacy is achieved by maintaining the adversary’s unpredictability about the data. For any mechanism to be useful, its *output* must maintain certain unpredictability that stems from the data [24]. The unpredictability is reduced once the output is revealed. Response perturbation solutions try to maintain the unpredictability on the *output* by introducing external noise and the inherent unpredictability on the *data* is ignored (the data is treated as a constant). However, non-randomized output does not necessarily reduce the unpredictability on individual data records: knowing that out of 1000 independent fair coin flips 500 come up head does not allow one to guess the outcome of each individual flip with probability better than half-half as long as the coins are kept hidden. In essence, quantities not known to the adversary can be modeled as random variables which accumulate in the sum queries. An access mechanism can harness such randomness and maintain privacy provided it can uphold the unknownness.

It is interesting to note that response perturbation solutions are in fact also relying on unknownness: Once a noisy answer is given out, they must keep the noise values secret. In this case while both the database and the output are *fixed*, the bound on the probability change still holds. By the same token, the following statement regarding our scheme does not make sense either: “fix  $d$  and  $d'$ , the the ratio  $\Pr[\mathcal{A}(d) = \tau] / \Pr[\mathcal{A}(d') = \tau]$  is unbounded if  $\mathcal{A}(d) \neq \mathcal{A}(d')$ ”. This is because here the randomness is to model the adversary’s uncertainty. Fixing  $d$  and  $d'$  does not necessarily provide more information as long as they are kept hidden. As an analogous example, consider the game where the adversary is to guess the output of a coin flip. The outcome is fixed once it is flipped. However, the fixedness does not help the adversary’s guess about the result if coin is not revealed.

## 5. PRIVACY OF SUM QUERIES

### 5.1 A Probability Tool

Our results rely on a very important probability theorem, multidimensional central limit theorem, which is summarized below:

**THEOREM 1** (MULTIDIMENSIONAL CLT [22]). *Let  $X_1, \dots, X_n$  be i.i.d. random vectors in  $\mathbb{R}^m$  with  $EX_i = \tau$  and finite covariance matrix  $V$ . If  $Y = \sum_{i=1}^n X_i$ , then  $(Y - n\tau) / \sqrt{n}$  converges in distribution to the  $m$ -dimensional gaussian distribution with zero mean and covariance matrix  $V$ .*

### 5.2 Single Query Case

In our scheme, we reveal the sums accurately. We show that this is safe under certain conditions. We first examine the case of a single query ( $m = 1$ ), and then extend the result to multiple (possibly correlated) queries.

**LEMMA 2.** *A mechanism  $\mathcal{A}$  answering the sum queries accurately is  $(\epsilon, \delta)$ -private if  $\forall d \in D^n, \forall i \in \{1, \dots, n\}, \Delta_i = \sum_{j=1, j \neq i}^n g(d_j)$  follows a distribution with probability density function (probability mass function if  $\Delta_i$  is discrete)  $p(x)$  satisfying the following conditions:*

1.  $\exists \mu$  such that  $p(\mu + x) = p(\mu - x)$ .
2.  $\forall x \geq y \geq \mu, p(x) \leq p(y)$ .
3.  $\exists x_0 \in [\mu, n - 1]$  such that  $\forall x \leq x_0, p(x) \leq p(x + 1) \exp(\epsilon)$  and  $\int_{x_0}^{\infty} p(x) dx \leq \delta$ .

**PROOF.** For any  $\tau \in \text{Range}(\mathcal{A})$ , for any  $d, d' \in D^n$  such that  $H(d, d') = 1$ , suppose they differ by the  $i$ th entry. Then

$$\frac{\Pr[\mathcal{A}(d) = \tau]}{\Pr[\mathcal{A}(d') = \tau]} = \frac{p(\Delta_i = \tau - g(d_i))}{p(\Delta_i = \tau - g(d'_i))}$$

$\Delta_i$  is a bell-shaped distribution that does not drop too quickly at one side. Let  $t = \tau - g(d_i)$  and  $t' = \tau - g(d'_i)$ . Similar to the proofs in [25], it suffices to consider the case where both  $t, t' \geq \mu$ . Noticing that  $t' \leq t + 1$ , then for any  $t \leq x_0$ , by the first half of condition (3)

$$\frac{p(t)}{p(t')} \leq \exp(\epsilon)$$

The integrated probability beyond  $x_0$  is

$$\int_{x_0}^{\infty} p(x) dx \leq \delta \quad (\text{second half of condition (3)})$$

As a consequence we get  $(\epsilon, \delta)$ -differential privacy.  $\square$

This lemma can be seen as providing an equivalent definition of differential privacy in the context of sum queries. Unlike the original definition, this formulation is defined with regard to *individual* data records. The result is similar to those in the response perturbation solutions in that they all perturb the data with some random quantity. The difference is that the protection is shown to be directly on individual data records, not the query function result. Despite the differences, we can use the results of existing work to derive privacy conditions in our context, because the mathematic reasonings are the same as they all rely on the properties of the perturbation distribution. Following the results of [5, 25, 27], some example distributions of  $\Delta_i$  satisfying the above conditions include

1. Gaussian  $\mathcal{N}(\mu, \sigma^2)$  with  $\sigma^2 \geq 2 \log(2/\delta) / \epsilon^2$  [5, 25].

2. Binomial  $B(n, 1/2)$  with  $n \geq 64 \log(2/\delta)/\epsilon^2$  [25].
3. Laplace  $Lap(\lambda)$  with standard deviation  $\lambda \geq 1/\epsilon$  (it can achieve  $\delta = 0$ ) [27].

### 5.3 Multidimensional Sum Queries

The result extends trivially to the case of  $m > 1$  *independent* queries. For each  $i$ ,  $g_1(d_i), \dots, g_m(d_i)$  are independent. To maintain privacy, it suffices to have each element of  $\Delta_i$  follow the same conditions specified above.

Independent queries are unrealistic. In particular, queries could be *adaptive* which poses greater threat to privacy. Existing works handle general, non-independent queries with *independent perturbation*: they add to each element of the response independent random noise. For example, [25] showed that independent perturbations could provide adequate protection against adaptive queries, provided the magnitude increases with  $m$ . Analogous to their results, we have the following for multidimensional adaptive sum queries:

LEMMA 3. *A mechanism  $\mathcal{A}$  answering the sum queries accurately is  $(\epsilon, \delta)$ -private if  $\forall d \in D^n, \forall i \in \{1, \dots, n\}$ , elements of  $\Delta_i = \sum_{j=1, j \neq i}^n g_j(d_i)$  are independent and each follows a distribution with probability density function (probability mass function if  $\Delta_i$  is discrete)  $p(x)$  satisfying the following:*

1.  $\exists \mu$  such that  $p(\mu + x) = p(\mu - x)$ .
2.  $\forall x \geq y \geq \mu, p(x) \leq p(y)$ .
3.  $\exists x_0 \in [\mu, n - 1]$  such that  $\forall x \leq x_0, p(x) \leq p(x + 1) \exp(\epsilon/m)$  and  $\int_{x_0}^{\infty} p(x) dx \leq \delta/m$ .

PROOF. A perturbation following a distribution satisfying the above conditions provides  $(\epsilon/m, \delta/m)$ -privacy for each query (lemma 2). By the composition theorem [25],  $(\epsilon, \delta)$ -privacy is thus guaranteed for the whole sequence.  $\square$

The conditions for some example distributions are

1. Gaussian  $\mathcal{N}(\mu, \sigma^2)$  with  $\sigma^2 \geq 2m^2 \log(2m/\delta)/\epsilon^2$ .
2. Binomial  $B(n, 1/2)$  with  $n \geq 64m^2 \log(2m/\delta)/\epsilon^2$ .
3. Laplace  $Lap(\lambda)$  with standard deviation  $\lambda \geq m/\epsilon$ .

Lemma 3 is an artificial proposition in that the conditions are unrealistic.  $\Delta_i$  is the sum of  $n - 1$  independent  $m$ -dimensional random vectors whose elements could be dependent of each other. It is thus impossible to have the elements of  $\Delta_i$  to be independent of each other. However, it is different from the case of independent queries as there is no restriction on the relationship among the elements of  $g(d_i)$ . It says that privacy can be obtained *if* each elements of  $g(d_i)$  is perturbed by an independent random quantity for all  $i$ . This is unattainable in our setting. Response perturbation mechanisms achieve this with external noise. The purpose of introducing such a lemma is to use it as a baseline. We then show that, under certain conditions, the situation that is attainable in our setting provides *more* protection thus privacy is also guaranteed.

### 5.4 Non-independent Gaussian Perturbation

Let  $\mathbf{I}_m$  be the  $m \times m$  identity matrix. From lemma 3 we know that, differential privacy can be achieved if each

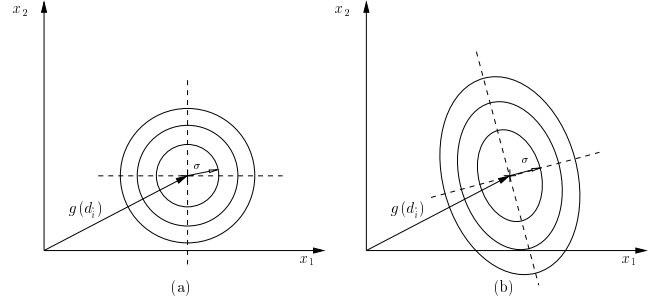


Figure 1: (a) Independent and (b) non-independent gaussian perturbations in 2-dimensional case. (b) has variance  $\sigma^2$  along its minor axis. Note how the perturbation in (b) “envelops” that in (a).

$g(d_i)$  is perturbed with a random vector drawn from an  $m$ -dimensional multivariate gaussian distribution with covariance matrix  $\sigma^2 \mathbf{I}_m$  (which corresponds to perturbing each elements with independent gaussian random quantities), provided  $\sigma^2$  is sufficiently large. We already pointed out that this is not attainable in our setting. However, when  $n$  is large, under certain conditions,  $\Delta_i$  converges in distribution to a multivariate gaussian distribution which potentially could provide adequate protection. The difficulty is that the limit distribution may have a non-diagonal covariance matrix. In other words, the perturbations to each of the  $g_j(d_i)$  may not be independent.

However, it is still possible that this type of perturbation could guarantee at least the same privacy. The intuition is, in the case of independent perturbation such as [5, 27, 15] etc., the noise added to the vector  $g(d_i)$  corresponds to an  $m$ -dimensional multivariate gaussian random variable with covariance matrix  $\sigma^2 \mathbf{I}_m$ . The surfaces of equal probability are  $m$ -dimensional hyperspheres. This means the amount of noise is the same in all directions. In our case, the covariance matrix of  $\Delta_i$  may not even be diagonal. The surfaces of equal probability are  $m$ -dimensional ellipsoids whose axes of symmetry are given by the principal components (the eigenvectors) of the covariance matrix. The length of the ellipsoid along the  $i$ th axis is  $c\sqrt{\lambda_i}$  where  $\lambda_i$  is the eigenvalue associated with the corresponding eigenvector. The perturbation to different directions are asymmetric and  $c$  is a constant determined by a given probability value. However, it is reasonable to speculate that, if the noise in the direction of the eigenvector associated with the *smallest* eigenvalue, which corresponds to the smallest variance, is greater than the required threshold, then perhaps this perturbation, although maybe non-independent for each query, can also provide the same level of privacy, because the perturbations in the other directions are all greater. The idea is illustrated in figure 1. This intuition is indeed correct and we provide a rigorous proof. We first state the asymptotic results and discuss convergence later.

THEOREM 2 (MAIN). *Let  $a_i = g(d_i) \in [0, 1]^m$ . Assuming  $a_1, \dots, a_n$  are i.i.d with  $E[a_i] = \tau$  and  $E[a_i a_i^T] = \tau \tau^T = V < \infty$ , the summation is  $(\epsilon, \delta)$ -private if  $n$  is sufficiently large and*

$$\lambda_{\min}(V) > \frac{2m^2 \log(2m/\delta)}{(n-1)\epsilon^2} \quad (1)$$

where  $\lambda_{\min}(V)$  is the smallest eigenvalue of matrix  $V$ .

PROOF. For any  $d \in D^n$  and any  $\hat{i} \in \{1, \dots, n\}$ , any possible output  $s$  can be written as  $s = g(d_{\hat{i}}) + \Delta_{\hat{i}}$  where  $\Delta_{\hat{i}} = \sum_{i=1, i \neq \hat{i}}^n g(d_i)$ . By central limit theorem (theorem 1), when  $n$  is large,  $\Delta_{\hat{i}}$  converges in distribution to  $\mathcal{N}(\mu, \Sigma)$  where  $\mu = (n-1)\tau$  and  $\Sigma = (n-1)V$ . Take  $\sigma^2 = 2m^2 \log(2m/\delta)/\epsilon^2$ . Assuming  $\tilde{\Sigma} = \Sigma - \sigma^2 \mathbf{I}_m$  is a well-defined covariance matrix (we will derive the conditions later), the addition of  $\Delta_{\hat{i}}$  can be seen as a two-step process: the first step adds  $\Delta_{\hat{i}}^{(1)}$  and the second step adds  $\Delta_{\hat{i}}^{(2)}$  where

$$\Delta_{\hat{i}}^{(1)} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_m) \text{ and } \Delta_{\hat{i}}^{(2)} \sim \mathcal{N}(\mu, \tilde{\Sigma})$$

This is equivalent to perturbing  $g(d_{\hat{i}})$  with two independent random quantities:  $\Delta_{\hat{i}}^{(1)}$  and  $\Delta_{\hat{i}}^{(2)}$ .  $\Delta_{\hat{i}}^{(1)}$  corresponds perturbing each elements of  $g(d_{\hat{i}})$  using independent gaussian randomness with variance  $\sigma^2 = 2m^2 \log(2m/\delta)/\epsilon^2$ . By lemma 3, this already guarantees  $(\epsilon, \delta)$ -privacy. Adding another independent random vector  $\Delta_{\hat{i}}^{(2)}$  does not reduce the level of privacy (otherwise an adversary can simply add more noise to  $g(d_{\hat{i}}) + \Delta_{\hat{i}}^{(1)}$  and break its privacy), thus the whole scheme is at least  $(\epsilon, \delta)$ -private.

The above can go through under the condition that  $\tilde{\Sigma}$  is a well-defined covariance matrix for an  $m$ -dimensional gaussian random variable. For this to hold, we require that  $\tilde{\Sigma}$  be symmetric and positive-definite. This translates to  $\lambda_{\min}(\tilde{\Sigma}) > 0$  which is equivalent to

$$\lambda_{\min}(V) > \frac{2m^2 \log(2m/\delta)}{(n-1)\epsilon^2}$$

□

## 6. SIMULATABLE QUERY AUDITING

Query auditing is another approach to statistical database privacy. Instead of perturbing the responses, it restricts the queries that can cause privacy breach. The work was initiated by [46, 16]. Latest results regarding sum queries are [37, 42]. Since they return accurate answers, the notion of privacy is different from that used by the perturbation approach. Until recently, most works (e.g., [10, 38]) used the *classic* notion, i.e., a compromise happens if a record is fully disclosed. [37] introduced *probabilistic* compromise for bounded range data where a significant change in the adversary's confidence about the range of a data record is considered privacy breach. We are not aware of any query auditing work that maintains differential privacy.

Query auditing can operate in one of two modes: offline or online. Given a series of queries and their answers that have already been given out, an offline auditor's task is to determine whether privacy breach has occurred whereas an online auditor must decide if answering a new query will infringe privacy. [37] demonstrated the difficulty in applying an offline algorithm to online auditing: An online auditor's denial may leak information. To handle this, [37] introduced the notion of *simulatable auditing*. The idea is that an online auditor's decision should be based on information that is available to the adversary who can then simulate the decision process thus no information is leaked by the denials. [42, 37] achieve simulatability for sum and max queries by basing the decision not on the data but a sample of the underlying distribution. We will show later that this is too strong and

disallows some cases that conform to the above intuition (thus are private).

## 6.1 Our Contributions

Using our privacy conditions, we can handle online and offline auditing in a uniform way. Compared with existing solutions, our scheme has a few advantages. (1) The privacy obtained is *differential privacy*. Not only is this result stronger, it also unifies the two lines of work and offers applications meaningful criteria for choosing suitable privacy mechanisms. A solution based on our scheme can be seen as a combination of both approaches: We use auditing to verify the privacy conditions that stem from perturbation results. (2) We introduce a relaxed definition of simulatability that still guarantees privacy but allows the auditor to use the data in its decision making process. We prove that our scheme is simulatable under the new definition. (3) Our method is agnostic to query language. Unlike [42, 37], where sum queries are restricted to subsets of rows to be aggregated, we support more general queries that allow for arbitrary functions on each record.<sup>4</sup>

## 6.2 Verifying the Privacy Conditions

Let a random matrix  $X \in \mathbb{R}^{m \times n}$  be such that  $X_{ji} = g_j(d_i)$ , i.e., its  $(j, i)$  entry is the  $j$ th query evaluated on user  $i$ 's data. Let  $X_i \in \mathbb{R}^{m \times 1}$  be the column vector representing user  $i$ 's query values. Assuming that data are already shifted so that  $X_i$ 's are zero-mean, and since all  $X_i$ 's are drawn i.i.d. from the same distribution, an unbiased estimator of the covariance matrix is

$$\hat{V} = \frac{1}{n-1} \sum_{i=1}^n X_i X_i^T = \frac{1}{n-1} X X^T \quad (2)$$

The privacy condition in theorem 2 transforms into

$$\lambda_{\min}(X X^T) > \frac{2m^2 \log(2m/\delta)}{\epsilon^2}$$

Notice that the singular values of a matrix  $X$  are the non-negative square roots of the eigenvalues of  $X X^T$ , the above condition is equivalent to

$$\sigma_m(X) > \frac{m\sqrt{2 \log(2m/\delta)}}{\epsilon} \quad (3)$$

where  $\sigma_m(X)$  is the  $m$ th largest singular value of  $X$ .

Inequality 3 provides a straightforward way to check the condition in theorem 2. Verifying it is a manageable task, especially for offline auditing. There are mature techniques for solving large scale eigenvalue problems. In particular some (e.g., the power method [48]) are easily parallelized. For example, link analysis algorithms such as PageRank [45] are routinely computed at the web scale. The problems they solve are very similar to ours and it is straightforward to adapt the technique to verify the condition.

Online auditing requires quick response. We now derive a simple method for detecting unsafe queries quickly. Suppose we have answered  $k$  queries which are all deemed safe. Let  $X|_k$  be the matrix of  $X$  restricted to the first  $k$  rows. Given the  $(k+1)$ th query, the condition becomes

$$\sigma_{k+1}(X|_{k+1}) > \frac{m\sqrt{2 \log(2m/\delta)}}{\epsilon} \quad (4)$$

<sup>4</sup>These are also the type in e.g., [5, 27].

Adding a new row to  $X|_k$  can be modeled as

$$X|_{k+1} = \begin{bmatrix} X|_k \\ 0 \end{bmatrix} + E_{k+1}, \text{ where } E_{k+1} = \begin{bmatrix} 0 \\ x_{k+1} \end{bmatrix}$$

$x_{k+1}$ , called the  $(k+1)$ th query vector, is a  $1 \times n$  vector whose  $i$ th element is the current query evaluated on the  $i$ th record and shifted by its mean. Such perturbation will cause changes to the singular values of matrix  $X$ . Using the results from matrix perturbation theory, we can derive a simple necessary condition for the privacy of this  $(k+1)$ th query. First we introduce Weyl's theorem which bounds the changes to the singular values with the norms of the perturbation:

**THEOREM 3** (WEYL [47]). *Let  $\tilde{\sigma}$  denote the perturbed quantity and  $\sigma_i$  the  $i$ -th singular value of a matrix  $A$ . Let  $E := \tilde{A} - A$ , then*

$$\max_i |\tilde{\sigma}_i - \sigma_i| \leq \|E\|_2$$

where  $\|E\|_2 = \sqrt{\lambda_{\max}(E^T E)}$  is the spectral norm.

Note that in our case it must be true that  $\text{rank}(X|_k) = k$  otherwise  $\sigma_k(X|_k) = 0$  and the previous  $k$  queries are not safe. This implies that  $\sigma_{k+1}(X|_k) = 0$ . In order for inequality 4 to hold,  $E_{k+1}$  must be sufficiently large so that it perturbs  $\sigma_{k+1}(X|_k)$  away from 0 by adequate amount (this also implies that  $E_{k+1}$  must increase the rank of  $X|_k$  so that  $\text{rank}(X|_{k+1}) = k+1$ ). Notice that for our particular form of perturbation,  $\|E_{k+1}\|_2 = \sqrt{x_{k+1} x_{k+1}^T} = \|x_{k+1}\|_2$ , i.e., the L2-norm of the row vector  $x_{k+1}$ . Immediately, we have:

**THEOREM 4.** *The  $(k+1)$ th query is unsafe if*

$$\|x_{k+1}\|_2 \leq \frac{m\sqrt{2\log(2m/\delta)}}{\epsilon}$$

Note that the bound, both here and in inequality 3, is independent of  $k$ . Even at early times when  $k$  is small,  $\|x_{k+1}\|_2$  still must be checked against such bound. This is consistent with the results in [15, 29, 5, 27, 44] in that the amount of perturbation to each query must be calibrated to the *total* number of queries. Otherwise a Dinur-Nissim style attack [15] can be mounted.

This is a simple condition.  $\|x_{k+1}\|_2$  can be easily computed as a by-product of the query. This result also has an intuitive interpretation. Note that elements of  $x_{k+1}$  are the current query evaluated at each user's data and  $\|x_{k+1}\|_2$  is the standard deviation (the values are already shifted so that they are zero-mean), the theorem states if the query results have small variance over the users, the aggregate may cause privacy breach. In the next section, we provide mathematical explanations to some intuitive judgements about certain types of queries regarding their privacy implications. Such judgements used to have only heuristic justifications.

**A Tighter Bound** One interesting observation about the perturbation theorem (3) and our privacy theorem 4 is that the bounds for the changes on the singular values in both are independent of the original matrix. This reflects the generality of the perturbation theorem which derives the bound only from the difference between the original and the perturbed matrices. Given the special structure of our problem, we can actually improve this bound by considering existing covariance matrix. Note that at the  $(k+1)$ th query,

we are interested in the changes in the  $(k+1)$ th singular value which was 0 for  $X|_k$ . The rows of  $X|_k$  span a subspace of dimensionality  $k$  which we denoted  $\mathcal{X}_k$ . For any row vector  $v \in \mathcal{X}_k$ , the augmented matrix  $\tilde{X}|_{k+1} = \begin{bmatrix} X|_k \\ v \end{bmatrix}$  shares the first  $k+1$  singular values with  $X|_k$ . Then for the  $(k+1)$ th query vector  $x_{k+1}$ , we can also bound the changes in singular values with  $\|\tilde{X}|_{k+1} - X|_{k+1}\|_2$ . In particular we can find the vector  $v$  that minimizes this quantity and obtain a tighter bound. It is well known that such  $v$  is simply the projection of  $x_{k+1}$  onto  $\mathcal{X}_k$ . Thus we have

**THEOREM 5.** *The  $(k+1)$ th query is unsafe if*

$$\|x_{k+1} - X|_k^T X|_k x_{k+1}^T\|_2 \leq \frac{m\sqrt{2\log(2m/\delta)}}{\epsilon}$$

### 6.3 Examples of Unsafe Queries

Leakage can be caused by either a single query or correlations between multiple ones. Our result handles them in a uniform way and provides mathematical explanations for both types of leakages. In the following we provide a few examples of such queries that we know, either intuitively or empirically, are privacy-breaching and show how they violate the condition in theorem 2. Essentially any new query that does not sufficiently increase the minimum singular value of existing data matrix  $X$  is privacy-breaching. Some queries may be more damaging in that they do not even increase the rank of  $X$ . Unsafe queries are classified into two types: those that are revealing by themselves and those that cause privacy breach when issued together with others. They manifest themselves in the following ways:

**Low Variation Queries:** A query that has very small variation across user data corresponds to a "small" row vector in  $X$  (recall that  $X$  is already made zero-mean by shifting). For example, since each  $g_j$  maps to a real number in  $[0, 1]$ , a sum that equals to  $n$  or 0 will fully expose every number. Such queries result in a row of all 0's in  $X$  and will be easily detected by the condition in theorem 4 which also quantifies the minimum variation for maintaining privacy.

A special case are queries that evaluate to 0 for a lot of records. Aggregates of sparse data are known to offer little privacy. Theorem 4 shows that the fundamental reason for this leakage is that the variance contained in the non-zero elements may not be enough to give adequate protection as defined by differential privacy. It provides a quantitative threshold for sparseness from privacy perspective.

**Correlated Queries:** Information could also be gleaned by correlating the answers to multiple queries. Consider the following example: Suppose the database contains information about individual's name and salary. One could issue two queries:  $g_1$  evaluates to one iff the employee's salary is at least \$100,000 and  $g_2$  evaluates to one iff the individual draws salary at least \$100,000 and is not named Joe. Accurate answers to these queries allow one to determine whether Joe's salary exceeds \$100,000. This was given in [5] as an example to show that exact answers to sum queries can be non-private.

We can examine the situation using our privacy criteria. Corresponding to these queries, the two rows of  $X$  are either (1) identical or (2) differ by only one element. Case (1) clearly violates the privacy condition as it causes  $\sigma_2(X) = 0$ . For case (2), without loss of generality, suppose Joe's

record is the last in the database. Let  $\alpha = \frac{1}{n} \sum_{i=1}^n g_1(d_i)$ ,  $\beta = \frac{1}{n} \sum_{i=1}^n g_2(d_i)$ . Notice that  $\alpha - \beta = 1/n$ . Subtracting the means from the query responses we obtain the  $X$  matrix

$$X = \begin{bmatrix} a_1 - \alpha, & a_2 - \alpha, & \dots, & a_{n-1} - \alpha, & 1 - \alpha \\ a_1 - \beta, & a_2 - \beta, & \dots, & a_{n-1} - \beta, & -\beta \end{bmatrix}$$

Compare  $X$  with

$$\tilde{X} = \begin{bmatrix} a_1 - \beta, & a_2 - \beta, & \dots, & a_{n-1} - \beta, & -\beta \\ a_1 - \beta, & a_2 - \beta, & \dots, & a_{n-1} - \beta, & -\beta \end{bmatrix}$$

the difference  $E := \tilde{X} - X$  is

$$E = \begin{bmatrix} 1/n, & 1/n, & \dots, & 1/n, & 1/n - 1 \\ 0, & 0, & \dots, & 0, & 0 \end{bmatrix}$$

And its spectral norm is  $\|E\|_2 = \sqrt{1 - 1/n}$ . By Weyl's theorem,  $\sigma_2(X) \leq \sigma_2(\tilde{X}) + \|E\|_2 = 0 + \|E\|_2$ . The bound is close to 1 for large  $n$  and is lower than the threshold specified in inequality 3 for any reasonable  $\epsilon$  and  $\delta$ .

A similar attack that queries the sums of the salaries instead of the counts can also be analyzed in a similar fashion since the salary numbers should be scaled to be between 0 and 1 according to the requirement of the query. <sup>5</sup>

## 6.4 Simulatability – A Revised Definition

The notion of simulatability is used in query auditing to model the possible leakage caused by the denials. The definition in [37] prohibits the query auditor from using the data when making a decision. We argue that this is too restrictive. The essence of simulatability is ensuring that the adversary, who has no access to the data, can derive the same information as it could from seeing the auditor's response. Forbidding the auditor from using the data in its decision making process is only a sufficient but not necessary means to achieve it. A very similar notion of simulation-based privacy has been used extensively in cryptography (e.g., in defining zero-knowledge), yet it is rarely the case in cryptography that the information to be protected (e.g., the message or a secret knowledge) is not allowed to be used in the construction. In fact in the construction of zero-knowledge proofs, the secret must be used otherwise by definition (specifically the soundness requirement) the proof will *not* be accepted. The point is, as long as the adversary could simulate the output using only public information, it does not reveal more. This perspective is orthogonal to whether the secret is used or not. We thus modify the definition in keeping with the cryptography approach:

**DEFINITION 2** (( $\alpha, \beta$ )-SIMULATABILITY). *Let  $d$  be drawn from distribution  $\mathcal{D}^n$ . Let  $Q_k$  be the set of  $k$  queries and  $A_{k-1}$  the answers to the first  $(k-1)$  ones. A safety checking mechanism *Safe* is a polynomial time algorithm that takes as input  $d, Q_k, A_{k-1}$  and  $\mathcal{D}$ . *Safe* is  $(\alpha, \beta)$ -simulatable if, for any  $\alpha > 0, \beta > 0$ , for any  $d \in \mathcal{D}^n, Q_k$  and  $A_{k-1}$ , there exists a polynomial time (in  $n$  and  $k$ ) simulator *Sim* such that, for sufficiently large  $n$*

$$\Pr[|\text{Sim}(Q_k, A_{k-1}, \mathcal{D}) - \text{Safe}(d, Q_k, A_k, \mathcal{D})| < \alpha] > 1 - \beta$$

*The probability is taken over the randomness in the distribution  $\mathcal{D}$  and the coin tosses of *Safe* and the simulator.*

<sup>5</sup>The scaling is necessary to bound the sensitivity of the query functions. See [27].

*A query auditing algorithm  $A^{\text{Safe}}$  that uses *Safe*'s output to make decisions about whether to deny a query is  $(\alpha, \beta)$ -simulatable if *Safe* is  $(\alpha, \beta)$ -simulatable.*

## 6.5 Achieving Simulatability

A simulatable query auditing algorithm with differential privacy is presented in figure 2. For simplicity, we only used the original condition in equation 4 which is equivalent to the one in theorem 2. It is straightforward to use theorem 4 to filter out unsafe queries quickly.

**Safe:** Inputs:  $d, Q_k, A_k$  and  $\mathcal{D}$ .

1. Construct/update the query matrix  $X$ .
2. Output  $\hat{\sigma} = \sigma_m(X)$ .

**A<sup>Safe</sup>:** Inputs:  $d, Q_k$ , and  $\mathcal{D}$ . Parameters  $m, n, \delta, \epsilon$ .

1. Compute answers  $a_k$  to the query  $g_k$  and update  $Q_k, A_k$ .
2. Compute  $\hat{\sigma} = \text{Safe}(d, Q_k, A_k, \mathcal{D})$ .
3. Return  $a_k$  if  $\hat{\sigma} > m\sqrt{2\log(2m/\delta)}/\epsilon$ , and DENY otherwise.

**Figure 2: Safety Check and Query Auditing Algorithm.**

**THEOREM 6.** *The online auditing scheme is simulatable.*

**PROOF.** The simulator works as follows:

1. Sample a dataset  $d'$  of size  $n$ , with each element sampled i.i.d. from  $\mathcal{D}$ .
2. At the  $k$ -th query, construct the query matrix  $X'$  where  $X'_{ji} = g_j(d'_i)$ ,  $i = 1, \dots, n, j = 1, \dots, k$ .
3. Return  $\hat{\sigma}' = \sigma_m(X')$ .

As a direct consequence of law of large numbers, the sample covariance matrix  $\hat{V}$  converges in probability to  $V$  [35]. That means for any  $\alpha, \beta > 0$ , there exists an integer  $N$  such that for all  $n > N$ , for all  $i, j = 1, \dots, k$ ,

$$\Pr[|\hat{V}_{ij} - V_{ij}| < \frac{\alpha}{2m}] > 1 - \frac{\beta}{2} \quad (5)$$

Notice that Weyl's theorem also applies to eigenvalues (see [47]) which means

$$|\lambda_{\min}(\hat{V}) - \lambda_{\min}(V)| \leq \|\hat{V} - V\|_2$$

The spectral norm can be bounded from above by Frobenius norm. And since  $\hat{\sigma} = \lambda_{\min}(\hat{V})$  and  $\sigma = \lambda_{\min}(V)$ , we have

$$|\hat{\sigma} - \sigma| \leq \|\hat{V} - V\|_F$$

$|\hat{V}_{ij} - V_{ij}| < \frac{\alpha}{2m}$  for all  $i, j = 1, \dots, k$  implies that  $\|\hat{V} - V\|_F < \alpha/2$ . Inequality 5 becomes

$$\Pr[|\hat{\sigma} - \sigma| < \frac{\alpha}{2}] > 1 - \frac{\beta}{2}$$

Similar result can be derived for  $\hat{\sigma}'$  since it is computed from sampling of the same distribution:

$$\Pr[|\hat{\sigma}' - \sigma| < \frac{\alpha}{2}] > 1 - \frac{\beta}{2}$$

By union bound and triangle inequality, we have

$$\Pr[|\hat{\sigma}' - \hat{\sigma}| < \alpha] > 1 - \beta$$

And it is easy to verify that the running time of the simulator is polynomial in  $n$  and  $k$ .  $\square$

## 7. DISCUSSION

### 7.1 Convergence

The analysis depends on the convergence of the sums to multivariate gaussian distribution when  $n$  is large. The Berry-Esséen theorem establishes that, for a sequence of i.i.d. random variables with finite third absolute moment, the uniform norm of the difference between the distribution functions of the sum and that of the gaussian (DF metric) is  $O(\frac{1}{\sqrt{n}})$  for the one-dimensional case [4, 30]. Bergström obtained similar result for multi-dimensional case with the additional requirement that the distribution admits an invertible covariance matrix [3]. This is the same as the standard error of the mean (the sum queries are equivalent to calculating the means). Applications requiring high accuracy may already mandate dataset large enough for CLT's convergence. And the above rate is for general distribution. Some data are known to be close to gaussian already (e.g., many biological characteristics) so they only require very few samples to converge.

The use of CLT is for mathematical convenience. The reliance on gaussian distribution is in two ways: (1) it allows one to decompose the perturbation into two independent steps and, more importantly, (2) with multivariate gaussian, the independence among the elements is equivalent to their uncorrelatedness. (1) is not essential. In fact the sum does not have to be gaussian to provide adequate protection. Consider the single query case, the particular distribution of the perturbation is not important (be it gaussian or laplace), as long as its pdf has the properties specified in lemma 2. The critical one is that  $p(x)/p(x+1)$  is bounded. If the distribution of one term has positive variance, and since variance accumulates with addition, the pdf of the sum tends to "flat out" and this condition is surely satisfied with large  $n$  (and this is another indication that aggregates across large population preserves privacy). So even without the decomposition the perturbation still protects privacy. In fact if the distribution of the terms is say laplace with large enough variance (note that the lower bound is independent of  $n$ ), then the proof can still go through even when  $n = 2$ , i.e., one data record provides protection for the other.

(2) is more difficult to forgo for proving the multiple query case. It allows us to reduce the security to that of independent perturbations. It is interesting open problem to see if and how this can be shown without using the reduction.

### 7.2 Issue of Multiple Databases

While our scheme provides *accurate* responses, it is more secure than the response perturbation solutions in the context of multiple databases. First, our scheme incorporates the idea from query auditing and handles repeated or related queries by denying them. The simple attack via repeated or related queries to reduce the variance of the perturbation does not work. Second, we show in the following that from querying a single database that implements our access mechanism, the adversary could obtain less accurate estimate of a record. Thus to obtain the same amount of information about a shared item, the adversary must query more databases, which increases its difficulty and cost. It may appear counterintuitive that an access mechanism that provides noisy responses is less secure than one that returns accurate results. This is not the case because in addition to limiting the number of queries, our method also enforces con-

ditions on the covariance matrix of the result vector which is *more* stringent.

Wlog, suppose  $d_1 \in \mathbb{R}$  is a record shared among database  $d$  and others and the adversary is trying to obtain information on  $d_1$ . The adversary could obtain  $m$  queries containing  $d_1$  from one database, represented using the  $m$ -vector  $s = d_1 a + \Delta$  where  $a = [1, 1, \dots, 1]^T$  and  $\Delta \sim \mathcal{N}((n-1)\mu, \Sigma)$ . Let  $\sigma^2$  be the lower bound on the safe variance (as defined in theorem 2), it must be that  $\lambda_{min}(\Sigma) \geq \sigma^2$ .

By averaging, the adversary can obtain

$$\mu = \frac{1}{n}s = s - (n-1)\mu = d_1 a + \Delta_0$$

where  $\Delta_0 \sim \mathcal{N}(0, \Sigma)$ .

The adversary could reduce the perturbation on  $d_1$  by computing the linear combinations of the elements of  $\mu$ . Let  $u \in \mathbb{R}^m$  and  $u^T a = \beta$  (i.e. the sum of  $u$ 's elements is  $\beta$ ), then  $u^T \mu = d_1 \beta + u^T \Delta_0$  and an estimate of  $d_1$  is  $\hat{d}_1 = \frac{u^T \mu}{\beta} = d_1 + \frac{u^T \Delta_0}{\beta}$  where  $\frac{u^T \Delta_0}{\beta} \sim \mathcal{N}(0, \frac{u^T \Sigma u}{\beta^2})$ . Consider the variance (note that the r.v. is a scalar)

$$\frac{u^T \Sigma u}{\beta^2} \geq \frac{u^T u}{\beta^2} \lambda_{min}(\Sigma) = \frac{u^T u}{\beta^2} \sigma^2$$

Since  $\beta$  is a scalar, we have  $\beta^2 = \beta \beta^T = u^T a a^T u = u^T A u$  where  $A = a a^T$  is a  $m \times m$  matrix with all entries being 1s. The above becomes

$$\frac{u^T \Sigma u}{\beta^2} \geq \frac{u^T u}{u^T A u} \sigma^2 \geq \frac{\sigma^2}{\lambda_{max}(A)} = \frac{\sigma^2}{m}$$

$\frac{\sigma^2}{m}$  is the case when the perturbations are all independent. It is interesting to notice that independent perturbation, as is done in the output perturbation solutions, allows the most effective noise reduction thus they are most vulnerable to the multiple databases attack. The equality holds iff  $a$  is an eigenvector of  $\Sigma$ . In all other situations, our scheme is strictly better.

## 8. CONCLUSION

In this paper we address the issue of privacy of sum queries over large databases. Using the notion of differential privacy, we provide the first mathematical proof for the intuition that aggregates are private and show that accurate computation can be privacy-preserving. We also derive conditions that can be used to verify whether certain computation respects privacy. As collecting and processing large amount of data is becoming a reality, our results allow for a new paradigm for performing privacy-preserving data analysis upon which practical solutions can be based.

## 9. REFERENCES

- [1] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *SIGMOD '00*.
- [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 *PODS '07*, pages 273–282, New York, NY, USA, 2007. ACM Press.
- [3] H. Bergström. On the central limit theorem in the space  $R_k$ ,  $k > 1$ . *Skand. Aktuarietidskr*, 28:106–127, 1945.

- [4] A. C. Berry. The accuracy of the gaussian approximation to the sum of independent variates. *Trans. Amer. Math. Soc.*, 49:122–136, 1941.
- [5] A. Blum, C. Dwork, F. McSherry, and K. Nissim. Practical privacy: the SuLQ framework. In *PODS '05*.
- [6] A. Blum, K. Ligett, and A. Roth. A learning theory approach to non-interactive database privacy. In *STOC '08*.
- [7] J. Canny. Collaborative filtering with privacy. In *IEEE Symposium on Security and Privacy. 2002*.
- [8] J. Canny. Collaborative filtering with privacy via factor analysis. In *SIGIR '02*, pages 238–245, Tampere, Finland, 2002. ACM Press.
- [9] K. Chaudhuri and C. Monteleoni. Privacy-preserving logistic regression. In *NIPS 2008*, 2008.
- [10] F. Chin and G. Ozsoyoglu. Auditing for secure statistical databases. In *ACM 81: Proceedings of the ACM '81 conference*, pages 53–59, 1981.
- [11] C.-T. Chu, S. K. Kim, Y.-A. Lin, Y. Yu, G. Bradski, A. Y. Ng, and K. Olukotun. Map-reduce for machine learning on multicore. In *NIPS 2006*, 2006.
- [12] R. Cramer and I. Damgård. Zero-knowledge proof for finite field arithmetic, or: Can zero-knowledge be for free? In *CRYPTO '98*. Springer-Verlag, 1998.
- [13] R. Cramer, I. Damgård, and J. B. Nielsen. Multiparty computation from threshold homomorphic encryption. In *EUROCRYPT '01*. Springer-Verlag, 2001.
- [14] A. S. Das, M. Datar, A. Garg, and S. Rajaram. Google news personalization: scalable online collaborative filtering. In *WWW '07*, 2007.
- [15] I. Dinur and K. Nissim. Revealing information while preserving privacy. In *PODS '03*.
- [16] D. Dobkin, A. K. Jones, and R. J. Lipton. Secure databases: protection against user influence. *ACM Trans. Database Syst.*, 4(1):97–106, 1979.
- [17] W. Du, Y. Han, and S. Chen. Privacy-preserving multivariate statistical analysis: Linear regression and classification. In *SIAM International Conference on Data Mining*, pages 222–233, 2004.
- [18] Y. Duan. Privacy without noise. In *CIKM '09: Proceeding of the 18th ACM conference on Information and knowledge management*, pages 1517–1520, New York, NY, USA, 2009. ACM.
- [19] Y. Duan and J. Canny. Practical private computation of vector addition-based functions. In *PODC 2007*.
- [20] Y. Duan and J. Canny. Practical private computation and zero-knowledge tools for privacy-preserving distributed data mining. In *SDM '08*, 2008.
- [21] Y. Duan, J. Wang, M. Kam, and J. Canny. A secure online algorithm for link analysis on weighted graph. In *Proceedings of the Workshop on Link Analysis, Counterterrorism and Security at the SIAM Data Mining Conference, 2005*, pages 71–81, April 2005.
- [22] R. Durrett. *Probability: theory and examples*. Duxbury Press, Belmont, CA, 2 edition, 1996.
- [23] C. Dwork. Ask a better question, get a better answer a new approach to private data analysis. In *ICDT 2007*.
- [24] C. Dwork. Differential privacy. In *ICALP 2006*, pages 1–12, 2006.
- [25] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT 2006*.
- [26] C. Dwork and J. Lei. Differential privacy and robust statistics. In *STOC '09: Proceedings of the 41st annual ACM symposium on Theory of computing*, pages 371–380, New York, NY, USA, 2009. ACM.
- [27] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *TCC 2006*.
- [28] C. Dwork, F. McSherry, and K. Talwar. The price of privacy and the limits of lp decoding. In *STOC '07*, New York, NY, USA, 2007. ACM Press.
- [29] C. Dwork and K. Nissim. Privacy-preserving datamining on vertically partitioned databases. In *CRYPTO 2004*, pages 528–544, 2004.
- [30] C.-G. Esséen. Fourier analysis of distribution functions. a mathematical study of the laplace-gaussian law. *Acta Mathematica*, 77(1):1–125, 1945.
- [31] A. Evfimievski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *PODS '03*, pages 211–222, 2003.
- [32] A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. In *KDD '02*, pages 217–228. ACM Press, 2002.
- [33] R. Gennaro, M. O. Rabin, and T. Rabin. Simplified vss and fast-track multiparty computations with applications to threshold cryptography. In *PODC '98*, pages 101–111. ACM Press, 1998.
- [34] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and Systems Sciences*, 28(2):270–299, 1984.
- [35] R. A. Johnson and D. W. Wichern. *Applied Multivariate Statistical Analysis (5th Edition)*. Prentice Hall, 2001.
- [36] M. Kearns. Efficient noise-tolerant learning from statistical queries. In *STOC '93*, 1993.
- [37] K. Kenthapadi, N. Mishra, and K. Nissim. Simulatable auditing. In *PODS '05*, 2005.
- [38] J. Kleinberg, C. Papadimitriou, and P. Raghavan. Auditing boolean attributes. In *PODS '00*, 2000.
- [39] Y. Lindell and B. Pinkas. Privacy preserving data mining. *Journal of cryptology*, 15(3):177–206, 2002.
- [40] F. McSherry and I. Mironov. Differentially private recommender systems: Building privacy into the netflix prize contenders. In *KDD '09*, pages 627–636, New York, NY, USA, 2009. ACM.
- [41] F. McSherry and K. Talwar. Mechanism design via differential privacy. In *FOCS '07*.
- [42] S. U. Nabar, B. Marthi, K. Kenthapadi, N. Mishra, and R. Motwani. Towards robustness in query auditing. In *VLDB '06*, pages 151–162, 2006.
- [43] A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. In *Proc. of 29th IEEE Symposium on Security and Privacy*, pages 111–125. IEEE Computer Society, 2008.
- [44] K. Nissim, S. Raskhodnikova, and A. Smith. Smooth sensitivity and sampling in private data analysis. In *STOC '07*, pages 75–84. ACM, 2007.
- [45] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1998.

- [46] S. P. Reiss. Security in databases: A combinatorial study. *J. ACM*, 26(1):45–57, 1979.
- [47] G. W. Stewart and J.-G. Sun. *Matrix Perturbation Theory*. Academic Press, 1990.
- [48] G. Strang. *Linear Algebra and Its Applications, 2nd Edition*. Academic Press, 1980.
- [49] J. Vaidya and C. Clifton. Privacy-preserving k-means clustering over vertically partitioned data. In *KDD '03*, pages 206–215. ACM Press, 2003.
- [50] R. Wright and Z. Yang. Privacy-preserving bayesian network structure computation on distributed heterogeneous data. In *KDD '04*, pages 713–718, New York, NY, USA, 2004. ACM Press.
- [51] X. Xiao and Y. Tao. Output perturbation with query relaxation. In *Proc. VLDB 2008*, volume 1, pages 857–869. VLDB Endowment, 2008.
- [52] Z. Yang, S. Zhong, and R. N. Wright. Privacy-preserving classification of customer data without loss of accuracy. In *SDM 2005*, 2005.