Journal of Privacy and Confidentiality (2010) **2**, Number 2, pp. 65–81

Polynomial-time Attack on Output Perturbation Sanitizers for Real-valued Datasets

Martin Merener*

1 Introduction

Output Perturbation is one of several strategies in the area of Statistical Disclosure Control (SDC), also known as Private Data Analysis. The general problem in SDC consists of releasing valuable information about individuals in a databasewhile preserving their privacy. Examples of this include databases containing health information about patients, customer electronic transactions, and web browsing history, among others.

These are situations in which a database is made up of records associated with individuals—each record describing attributes of the corresponding individual. In addition, these attributes provide highly valuable information from a statistical point of view, either for commercial purposes or government/private decision-making. However, each individual record contains sensitive information that cannot be publicly released. So instead of releasing the true data, a *curator* with access to it has to create and release a related piece of information, the *sanitized data*, satisfying *utility* and *privacy* requirements.

In the *non-interactive* setting, the curator uses the original data only once to compute the sanitized data, which is then published. In the *interactive* setting, users who aim to get information about the original data, can send queries to the curator, who will respond to them in a way that does not disclose private information, but is useful in some sense [10].

In the interactive *query-response* model there are further subdivisions, such as: whether queries are all submitted at once and then replied (*non-adaptive*) or are allowed to depend on previous answers (adaptive) [6]; and whether the curator provides privacy by modifying the queries or their true answers [8].

We are interested in the *output perturbation* strategy [1, 3, 6, 12, 14, 19], which is non-adaptive, and in which the curator computes the true answer for the original query and releases a perturbation of it to preserve privacy.

Many of the strategies in SDC focus only on the privacy issues and not on the utility side, under the assumption that if the sanitized data is sufficiently closely related to the original data, then some of the utility is preserved. This is stated in [8] as: *Throughout the discussion we will implicitly assume the statistical database has some non-trivial utility, and we will focus on the definition of privacy.* The term *statistical database* refers to the pair database-sanitizer [6].

*York University, Toronto, Canada, mailto:merener@mathstat.yorku.ca

Other strategies focusing on utility aspects attempt to convey utility by creating and releasing fully *synthetic data* satisfying certain conditions related to its usefulness [24], while the privacy requirement is assumed to be satisfied due to the artificial nature of the data. In [23] the author expresses this as: *This can protect confidentiality, since identification of units and their sensitive data can be difficult when the released data are not actual, collected values.*

In the strategy of non-adaptive output perturbation studied here, the rigorous treatment and results apply to privacy aspects. Utility is somehow determined by the variable that is meant to provide privacy, the amount of perturbation applied to the true answers to the queries. Higher perturbation means more protection of privacy and less utility preserved (and vice versa).

In SDC, positive and impossibility results coexist; on one hand there is an interest in developing sanitizers that provide privacy, on the other hand we want to create adversaries showing the limitations to achieve that. Our work is built upon an impossibility result developed by Dinur and Nissim [6], showing that the output perturbation sanitizer with perturbation $o(\sqrt{n})$ fails to provide privacy. They do this by defining an adversary algorithm that makes $O(n \log^2 n)$ random queries to the curator of any database with one attribute and n records, and with the corresponding answers this adversary constructs (with high probability of success) a good approximation of the database.

We revisit their results and more explicitly explain the interplay of the parameters involved in the problem through an inequality that represents the trade-offs among them. Our solution, following Dinur and Nissim's approach, is given in a general context that includes databases with binary values, using a *counting* metric to measure the adversary's error—case of study in [6]—and databases with bounded real-values, which can be assumed to be in the interval $[0, 1]$, where other appropriate metrics are used.[1]

The origin of our results is our new version of a result in [6] called Disqualifying Lemma, whose proof is based on Azuma's Lemma, and is used as a "probability tool". We instead give a proof based on a generalization of the Central Limit Theorem, the Berry-Esseen Theorem [5].

We also show that in the binary case the adversary is more efficient than what was claimed in [6]. However, other works [12, 14] improve [6] using different techniques, obtaining adversary attacks more efficient (see Related work) than the one we study here. The goal of this paper is to give a new presentation with rigorous mathematical proofs of the impossibility results in [6], and to do so in a way that explicitly includes the case of real-valued databases.

1.1 Related work

Different works on SDC from the early 1970s dealt with the privacy breaches that could have been caused by the information released by statistical offices [4, 16]. Since then,

[1] The difference in metrics makes the binary case not to be a special case in the real-value setting.

due to advances in communications, the progressively available data in digital media, and the large amount of databases reachable through the Internet, questions regarding the privacy of individuals increasingly captured the attention of statisticians [7, 26] and computer scientists [2, 15], among others.

A widely cited work in recent years is a paper by Dinur and Nissim [6], whose main result is a polynomial-time algorithm M that reconstructs—with low error and high probability of success—a database from the perturbed replies to queries to such a database. This attack has time complexity determined by $O(n \log^2 n)$ number of queries, can defeat sanitizers with perturbation up to $o(\sqrt{n})$, and is applicable to binary databases with one attribute. The success of this attack represents a breach of privacy, an impossibility result regarding the preservation of privacy throughout perturbation $o(\sqrt{n})$.

This was followed by the study of further constrains on the complexity of the adversary algorithm, which may allow the existence of privacy-provider sanitizers. This is resolved positively in [6] by establishing a sanitizer applying perturbation $\sqrt{T(n)}$, that cannot be defeated by attackers of certain form (the attackers must be modelled in order to prove statements about them) having time complexity bounded by $T(n)$.

Posterior works have generalized this positive result in the case of multiple-attribute databases [13], and in a more general concept of query [3]. Further generalizations have given place to concepts such as *differential privacy* [8, 9, 10, 19], *sensitivity*, and *noise calibration* [11, 20].

There have also been improvements in the impossibility results. A follow up work of [6], with attacks more efficient than the one studied here, is [14]. Dwork and Yekhanin introduce a deterministic attack based on a sub-quadratic algorithm that computes the Hadamard Transformation. This adversary algorithm requires only $O(n \log n)$ steps to reconstruct the database with $o(n)$ error, making (non-random) "subset-sum" queries to a sanitizer applying $o(\sqrt{n})$ perturbation. The setting is binary-valued databases with one attribute.

In another work [12], new results on LP decoding are given, allowing successful attacks on output perturbation sanitizers, that are more efficient than the one described in this article. The adversary submits $O(n)$ queries, but instead of asking the value of $\sum_{i \in q} d_i$ for a random $q \subseteq [n]$ (which is the same as $\langle a, d \rangle$ for $a = \mathit{supp}(q)$), the adversary asks the value of $\langle a, d \rangle$, where a is a random vector with independent $N(0, 1)$ entries. If $\rho < \rho^* \approx 0.239$ and a ρ-fraction of the queries have perturbation at most α (with the rest having arbitrary perturbation), then with negligible probability the adversary fails to learn a proportion greater than $c_\rho \alpha^2 / n$ of the coordinates of the binary database d. When $\alpha \in o(\sqrt{n})$, this means that with overwhelming probability the adversary learns $\Omega(n)$ of the coordinates of d. A similar result with a corresponding smaller $\rho^*_{\pm 1}$ is obtained for $a \in \{\pm 1\}^n$ uniformly at random. The LP decoding theorems in [12] can be used to extend the adversary attack to real-valued databases. Although we do not make that analysis here, these extensions of [12] to real-valued databases should be more efficient than our results, whose main contribution is to provide an alternative and more general proof and to present the result explicitly for real-valued databases.

More recent extensions and generalizations of [6] can be found in [17] and [18], although the privacy definitions used are different than the one considered here and in [6]. In [17], databases are real-valued and queries are given by linear functions $F : \mathbb{R}^n \to \mathbb{R}^d$ with coefficients in $[-1, 1]$, and Hardt and Talwar study how much error any differentially private mechanism must add, while [18] considers *minimal privacy* and *differential privacy*.

1.2 This work

Our work focuses on the attacker algorithm M described in [6]. We find a property involving the function *dist* used to measure the error of the output of M with respect to the true database, which is sufficiently strong to prove the success of M.

This property is sufficiently general to include the real-valued case (and the binary case in [6]) with different alternatives for *dist*, such as the metrics induced by the l_1 and l_2 norms, and a function that counts the number of coordinates that differ by at least a certain threshold.

The results are proved in this general context, the key being the following inequality, where Δ^n_{dist} denotes the maximum possible error of M with respect to *dist* (in the binary case this is equal to n), M^{A_d} is the attacker algorithm that uses as an oracle a query-response sanitizer A accessing the database d of size n, $t(n)$ is the number of calls to A, and τ is the perturbation of A:

$$P[dist(M^{A_d}(1^n), d) \geq \epsilon\Delta^n_{dist}] \leq (n+1)^n \cdot \Big(\frac{\alpha_\epsilon + \beta_\epsilon \cdot \tau(n)}{\sqrt{n}}\Big)^{t(n)}$$

where $\alpha_\epsilon, \beta_\epsilon$ are explicit functions of ϵ.

The inequality—which bounds the probability of the adversary having at least ϵ relative error—relates all the parameters of the problem and can be used to specialize some of them and see how the others are determined. For example, if the perturbation τ is $O(n^{1/3})$, then it turns out that certain $t(n) \in O(n)$ are good enough to make the attacker M successful, i.e., make the right side of the inequality negligible on n. Likewise, if the perturbation is $o(\sqrt{n})$, then certain $t \in o(n \log n)$ suffice to make M successful.

1.3 Organization of the paper

In Section 2 we provide the definitions used later, all of which (except Definition 4) were established in [6]. In Section 3 we review the attacker algorithm presented in [6]. Then we review different functions that can be used to measure the adversary error in the real-valued case (with essentially the same adversary algorithm from [6]). After that we establish a property about the error-measure function *dist* (which could be a metric or not) and we show that our cases of interest satisfy this property. In Section 4 we prove the impossibility results based only on this common property and our version of the Disqualifying Lemma, whose proof is given in detail in the Appendix (Section 6).

Section 5 contains some final remarks.

2 Preliminaries

Notation. For each $m \in \mathbb{N}$, denote by $[m]$ the set $\{1, ..., m\}$. Also, if $g, h : \mathbb{N} \to \mathbb{R}_{\geq 0}$ and $\lim_{n\to\infty} \frac{g(n)}{h(n)} = 0$, denote this as $g \in o(h(n))$. Finally, if $\exists B > 0$ such that $\forall n \in \mathbb{N}$, $\frac{g(n)}{h(n)} \leq B$, denote this as $g \in O(h(n))$.

The databases considered in this work describe a single real-valued variable, an *attribute*, about n different *individuals*. For simplicity assume that the range of this attribute is $[0, 1]$, but notice that all the results established can as well be given for any other interval $[a, b] \subseteq \mathbb{R}$.

A database is accessed by *users* through queries sent to a *curator* of the database, who in turn replies to the queries using a *sanitizer* responsible for protecting the privacy of the individuals whose data is in the database. In this model, a query is determined by a subset of $[n]$, and its answer is the sum of the corresponding values of the database.

To preserve the privacy of the individuals, the curator uses the sanitizer to release a perturbation of the true answer instead of the true answer. The only restriction given to the sanitizer is that the difference between the true answer of a query and its reply is bounded from above by a publicly known value.

An *adversary* or *attacker* is an individual who tries to obtain a good approximation of the values of the database, based on the replies of some queries sent to the curator.

Definition 1. An n-dimensional *database* is an element $d \in [0, 1]^n$. A *query* about the database d is a subset $q \subseteq [n]$, and its *true answer* is $\sum_{i\in q} d_i$.

Definition 2. Given $n \in \mathbb{N}$ and $d \in [0, 1]^n$, a (*query-response*) *sanitizer* is a function $A_d : \{q | q \subseteq [n]\} \to \mathbb{R}$. The sanitizer A_d is said to have *perturbation* $\tau > 0$ *with respect to* d if $\forall q \subseteq [n], |\sum_{i\in q} d_i - A_d(q)| \leq \tau$.

Definition 3. An *adversary* is an algorithm M with input 1^n, for any $n \in \mathbb{N}$. If the input is 1^n (for a particular n) then M can use, as an oracle, a certain function $O : \{q | q \subseteq [n]\} \to \mathbb{R}$. M is also allowed to use random choices from $\{0, 1\}$ to produce an output.

One need not explicitly define O in Definition 3. Only when we want to state specific results will we say what O is. If $O = A_d$ for some $d \in [0, 1]^n$ and $n \in \mathbb{N}$, then the output of M on input 1^n is denoted $M^{A_d}(1^n)$. The vector $M^{A_d}(1^n) \in [0, 1]^n$ is meant to approximate the database d.

The approximation is evaluated using a function *dist* taking non-negative real-values and having both d and $M^{A_d}(1^n)$ as an argument. We can think *dist* as a metric, although we also consider a case in which the function "measures" proximity (the closer to zero the better the approximation), but is not an actual metric.

Each call of M to O contributes one unit of time towards the computational complexity of M. The reason for considering 1^n as the input of M (instead of n) is that otherwise M would not have polynomial-time complexity, simply because creating a query of size $\frac{n}{2}$ (the expected size of a random subset $q \subseteq [n]$ is $\frac{n}{2}$) would require an exponential number of steps with respect to the size of the input.

We want query-response sanitizers to confront adversaries, so these mechanisms have to be compatible. So far, each sanitizer depends on a particular $d \in [0,1]^n$ (for a fixed $n \in \mathbb{N}$), while each adversary works for any $n \in \mathbb{N}$. Let D^n be the set of possible databases of size n, i.e., $D^n = \{0,1\}^n$ or $D^n = [0,1]^n$.

Definition 4. An *assembled* (*query-response*) *sanitizer* consists on a collection $A = \{A_d\}_{d \in D^n, n \in \mathbb{N}}$, where $\forall n \in \mathbb{N}, \forall d \in D^n$, A_d is a sanitizer as in Definition 2. The assembled sanitizer A has *perturbation* $\tau : \mathbb{N} \to \mathbb{R}_{>0}$ if $\forall n \in \mathbb{N}$, $\forall d \in D^n$, A_d has perturbation $\tau(n)$ with respect to d.

This work is about the fact that given an assembled sanitizer with a certain perturbation, it is possible to describe an adversary and conditions on the relevant parameters of the problem so that the adversary causes a privacy breach in a sense defined below.

Since both d and $M^{A_d}(1^n)$ are in $[0,1]^n$, the maximum error that the adversary can make is $\Delta^n_{dist} = \max_{x,y \in [0,1]^n} dist(x,y)$, i.e., the *diameter* of $[0,1]^n$ with respect to $dist$. An assembled sanitizer is considered to cause a breach of privacy if there is an efficient adversary algorithm that reconstructs the values of any database with low error and high probability by using only the assembled sanitizer as an oracle. Recall that a function $f : \mathbb{N} \to \mathbb{R}_{\geq 0}$ is *negligible* if:

$$\forall a \in \mathbb{N}, \exists n_0 \in \mathbb{N}, \forall n \geq n_0, f(n) < \frac{1}{n^a}.$$

Definition 5. An assembled sanitizer A causes an ϵ-privacy-breach if there exists a polynomial-time adversary M and a negligible function $p : \mathbb{N} \to \mathbb{R}_{\geq 0}$ such that $\forall n \in \mathbb{N}$, $\forall d \in D^n$,

$$P[dist(M^{A_d}(1^n), d) \geq \epsilon \Delta^n_{dist}] \leq p(n)$$

where the probability is taken with respect to the randomness in the adversary algorithm.[2]

The model described has five relevant parameters: (1) the dimension n of the database d; (2) the adversary relative error $\epsilon \in (0,1)$; (3) the perturbation τ of the query-response sanitizer; (4) the complexity of the adversary algorithm (which depends on the number of queries it makes); and (5) the probability of failure (or success) of the adversary.

Our goal is to understand how these parameters interact and find conditions on them that guarantee a privacy breach. Statements such as the following are the ones we are interested in: for any given $\epsilon > 0$ and any given $\tau \in o(\sqrt{n})$, if A is an assembled

[2]This is the only source of randomness considered.

sanitizer with perturbation τ, then there exists a polynomial-time adversary M and an n_0 such that if $n \geq n_0$ then $\forall d \in D^n$,

$$P[dist(M^{A_d}(1^n), d) \geq \epsilon \Delta^n_{dist}] \leq 2^{-n}.$$

In other words, for any $\epsilon > 0$, every assembled sanitizer whose perturbation is in $o(\sqrt{n})$ causes an ϵ-privacy-breach.

3 Specifying the $dist$ function

In this section we consider two possible scopes for the database d. The first one is the *binary* case, with databases taking values in $\{0,1\}^n$ and *dist* given by the metric counting the number of coordinates in which two vectors differ.

In the other case databases are *real-valued*, and three different *dist* functions are analyzed. Two of them are the metrics induced by the l_1 and l_2 norms, while the third one is a function that, not being a metric, measures proximity by counting the number of coordinates that differ by at least a certain threshold.

3.1 Binary Databases

In [6] the databases are $d \in \{0,1\}^n$, and they are compared with the adversary's guess through $dist(x,y) = |\{i \in [n] : x_i \neq y_i\}|$, which can be at most $\Delta^n_{dist} = \max_{x,y \in \{0,1\}^n} dist(x,y) = n$.

The following is the algorithm exhibited in [6]. It depends on the parameter $t = t(n)$, referred to as *number of queries*. As we will state in Remark 2, the number of queries is closely related to, and determines, the time complexity of the algorithm. We treat $t(n)$ as an integer, so if it is not, assume it denotes $\lceil t(n) \rceil$.

Algorithm M; input 1^n with $n \in \mathbb{N}$; output c'; access to oracle O.

(1) Choose $t(n)$ subsets of $[n]$, independently and unif. at random, $q_1, ..., q_{t(n)}$.

(2) Solve the LP: find $c \in [0,1]^n$ so that $\forall j \in [t(n)]$, $|\sum_{i \in q_j} c_i - O(q_j)| \leq \tau(n)$.

(3) If $c_i > \frac{1}{2}$ then $c'_i = 1$, else $c'_i = 0$.

We assume that O is associated with a publicly known perturbation τ, because we are not concerned about M being useful otherwise. We want M to be effective approximating d when $O = A_d$, and it has a publicly known perturbation τ with respect to d. In that case the LP in (2) has at least one solution, $c = d$. To obtain a random query in (2) the adversary flips a fair coin for each element of $[n]$.

3.2 Real-valued Databases

For real-valued databases $d \in [0,1]^n$ consider the same adversary algorithm, except that the output does not have to be rounded to a binary vector.

Algorithm M; input 1^n with $n \in \mathbb{N}$; output c; access to oracle O.

(1) Choose $t(n)$ subsets of $[n]$, independently and unif. at random, $q_1, ..., q_{t(n)}$.

(2) Solve the LP: find $c \in [0,1]^n$ so that $\forall j \in [t(n)]$, $|\sum_{i \in q_j} c_i - O(q_j)| \leq \tau(n)$.

Consider three different $dist$ functions:

$dist(x,y) = ||x-y||_1$, where $||x||_1 = \sum_{i=1}^n |x_i|$, so $\Delta_{dist}^n = n$.

$dist(x,y) = ||x-y||_2$, where $||x||_2 = (\sum_{i=1}^n x_i^2)^{1/2}$, so $\Delta_{dist}^n = \sqrt{n}$.

$dist_\gamma(x,y) = |\{i \in [n] : |x_i - y_i| \geq \gamma\}|$, for any fixed $\gamma \in (0,1)$, so $\Delta_{dist_\gamma}^n = n$.

Although $dist_\gamma$ is not a metric since it does not satisfy the triangle inequality (or the fact that it is 0 only when $x = y$), it seems a reasonable way to measure how well the guess made by the adversary approximates the database. Restricted to $\{0,1\}^n$, $dist_\gamma$ becomes the $dist$ function used in the binary case.

3.3 A common property

We introduce some notation. For $n \in \mathbb{N}$, let $K_n = \{0, \frac{1}{n}, \frac{2}{n}, ..., 1\}$. Given $x \in [0,1]^n$, denote by $\hat{x}$ the point in K_n^n such that each $\hat{x}_j$ is the closest point in K_n to the corresponding x_j (if there are two choices take the smallest). Similarly, x' is the point in D^n such that each x'_j is the closest to x_j. Although for the real-valued case this is trivial since $x' = x$, it is convenient to keep the notation. The property we are interested in is the following:

$$\forall \epsilon \in (0,1), \exists \rho \in (0,1), \exists n_\epsilon, \forall n \geq n_\epsilon, \forall c \in [0,1]^n, \forall d \in D^n,$$

$$dist(c', d) \geq \epsilon \Delta_{dist}^n \;\Rightarrow\; ||\hat{c} - d||_2 \geq \rho\sqrt{n}. \tag{3.1}$$

Note that the previous implication is equivalent to $\frac{||\hat{c}-d||_2}{\sqrt{n}} < \rho \Rightarrow \frac{dist(c',d)}{\Delta_{dist}^n} < \epsilon$, where the $\frac{||\hat{c}-d||_2}{\sqrt{n}}$ and $\frac{dist(c',d)}{\Delta_{dist}^n}$ are "normalized metrics" with which $[0,1]^n$ has diameter 1. Now we verify that (3.1) holds in both the binary and real-valued cases.

Binary. Let $c \in [0,1]^n$ and $d \in \{0,1\}^n$. First, $||\hat{c}-d||_2 \geq \frac{1}{3}\sqrt{|\{i : |\hat{c}_i - d_i| \geq \frac{1}{3}\}|}$. Also, if $n \geq 3$, then $c'_i \neq d_i \Rightarrow |\hat{c}_i - d_i| \geq \frac{1}{3}$, so $|\{i : |\hat{c}_i - d_i| \geq \frac{1}{3}\}| \geq dist(c', d)$. Hence $||\hat{c} - d||_2 \geq \frac{1}{3}\sqrt{dist(c', d)}$, so: $dist(c', d) \geq \epsilon n \Rightarrow ||\hat{c} - d||_2 \geq \frac{1}{3}\sqrt{\epsilon n}$, meaning that (3.1) holds for $n_\epsilon = 3$, $\rho = \frac{\sqrt{\epsilon}}{3}$.

Real-valued; $||.||_1$. Let $c, d \in [0,1]^n$. First, $\forall x \in \mathbb{R}^n, ||x||_2 \geq \frac{1}{\sqrt{n}}||x||_1$ (by Cauchy-Schwarz inequality). Also, $dist(c, \hat{c}) \leq \frac{1}{2}$, so by triangular inequality, $dist(\hat{c}, d) \geq dist(c, d) - 1/2$, which gives $||\hat{c} - d||_2 \geq \frac{1}{\sqrt{n}}\big(dist(c, d) - 1/2\big)$. Hence, $dist(c, d) \geq \epsilon n \Rightarrow ||\hat{c} - d||_2 \geq \frac{\epsilon n - 1/2}{\sqrt{n}}$. Finally, if $n \geq \frac{1}{\epsilon}$ then $\frac{\epsilon n - 1/2}{\sqrt{n}} \geq \frac{\epsilon}{2}\sqrt{n}$, so (3.1) is true if $n_\epsilon = \lceil \frac{1}{\epsilon} \rceil$ and $\rho = \frac{\epsilon}{2}$.

Real-valued; $||.||_2$. Let $c, d \in [0,1]^n$. Since $dist(c, \hat{c}) \leq \frac{\sqrt{n}}{2n} = \frac{1}{2\sqrt{n}}$, then $||\hat{c} - d||_2 \geq dist(c, d) - \frac{1}{2\sqrt{n}}$, giving: $dist(c, d) \geq \epsilon\sqrt{n} \Rightarrow ||\hat{c} - d||_2 \geq \epsilon\sqrt{n} - \frac{1}{2\sqrt{n}}$. But if $n \geq \frac{1}{\epsilon}$, then $\epsilon\sqrt{n} - \frac{1}{2\sqrt{n}} \geq \frac{\epsilon}{2}\sqrt{n}$, so (3.1) holds with $n_\epsilon = \lceil \frac{1}{\epsilon} \rceil$ and $\rho = \frac{\epsilon}{2}$.

Real-valued; $dist_\gamma$. Let $c, d \in [0,1]^n$. First, $||\hat{c} - d||_2 \geq \frac{\gamma}{2}\sqrt{dist_{\gamma/2}(\hat{c}, d)}$. Also, if $n \geq \frac{1}{\gamma}$, then $|c_i - d_i| \geq \gamma \Rightarrow |\hat{c}_i - d_i| \geq \frac{\gamma}{2}$, so $dist_{\gamma/2}(\hat{c}, d) \geq dist_\gamma(c, d)$, which gives $||\hat{c} - d||_2 \geq \frac{\gamma}{2}\sqrt{dist_\gamma(c, d)}$. Then, $dist_\gamma(c, d) \geq \epsilon n \Rightarrow ||\hat{c} - d||_2 \geq \frac{\gamma}{2}\sqrt{\epsilon n}$, so to make (3.1) true suffices $n_\epsilon = \lceil \frac{1}{\gamma} \rceil$ and $\rho = \frac{\gamma\sqrt{\epsilon}}{2}$.

4 Results

To prove the results we are interested in, we use the following lemma, whose proof is given in the Appendix. In particular, this lemma implies the Disquali-fying Lemma in [6]. To abbreviate, denote $I_\rho^n = \{y \in [-1,1]^n : ||y||_2 \geq \rho\sqrt{n}\}$.

Lemma 6. *Let $\rho \in (0,1)$ and $n \in \mathbb{N}$. If $\lambda_\rho = \frac{64}{\rho^3}$ and $\mu_\rho = \frac{4}{\sqrt{2\pi}\rho}$, then $\forall \tau > 0$, $\forall y \in I_\rho^n$,*

$$P\big[\big|\sum_{i \in q} y_i\big| \leq \tau\big] \leq \frac{\lambda_\rho + \mu_\rho \cdot \tau}{\sqrt{n}}$$

where the probability is taken with respect to a random $q \subseteq [n]$, obtained by n independent tosses of a fair coin to determine, for each $j \in [n]$, if $j \in q$.

Corollary 7. *(Disqualifying Lemma [6].) Let $\epsilon \in (0,1)$, let $\tau \in o(\sqrt{n})$, and take, for each n, any two points $x, d \in [0,1]^n$ such that $P_i[|x_i - d_i| \geq \frac{1}{3}] > \epsilon$. Then there exists $\delta > 0$, independent of n, such that for n sufficiently large (depending on ϵ and τ),*

$$P\big[\big|\sum_{i \in q}(x_i - y_i)\big| > 2\tau(n) + 1\big] > \delta$$

where the probability is taken as in Lemma 6.

Proof. It is enough to show $P\big[\big|\sum_{i\in q}(x_i - y_i)\big| \le 2\tau(n)+1\big] < \frac{1}{2}$, i.e., $\delta = \frac{1}{2}$. Note that $P_i[|x_i - d_i| \ge \frac{1}{3}] > \epsilon$ means $|\{i \in [n] : |x_i - d_i| \ge \frac{1}{3}\}| > \epsilon n$. Let $\rho = \frac{\sqrt{\epsilon}}{3}$. If $|\{i \in [n] : |x_i - d_i| \ge \frac{1}{3}\}| > \epsilon n$, then $x - d \in I^n_\rho$; this follows from $||x-d||_2 \ge \sqrt{\epsilon n/9} = \rho\sqrt{n}$. So by Lemma 6, it suffices to show $\frac{\lambda_\rho + \mu_\rho \cdot (2\tau(n)+1)}{\sqrt{n}} < \frac{1}{2}$. Now, $\frac{\lambda_\rho+\mu_\rho\cdot(2\tau(n)+1)}{\sqrt{n}} = \frac{1728}{\epsilon^{3/2}\sqrt{n}} + \frac{12}{\sqrt{2\pi\epsilon}\sqrt{n}} + \frac{24}{\sqrt{2\pi\epsilon}} \cdot \frac{\tau(n)}{\sqrt{n}}$. Since $\tau \in o(\sqrt{n})$, there is n_0 (depending on ϵ and τ) such that if $n \ge n_0$ then $\frac{\lambda_\rho+\mu_\rho\cdot(2\tau(n)+1)}{\sqrt{n}} < \frac{1}{2}$. □

Assuming that property (3.1) is true, given any $\epsilon \in (0,1)$, let ρ_ϵ and n_ϵ be the corresponding values from (3.1). Then define for each $n \ge n_\epsilon$ and $d \in D^n$, $X^{d,n}_\epsilon = \{x \in K^n_n : x - d \in I^n_{\rho_\epsilon}\}$.

Again, given that $x \in [0,1]^n$, $x' \in D^n$ denotes the closest point to x in D^n (with respect to $||.||_\infty$), we can easily unify the two descriptions of the adversary algorithm M into one that applies to both cases. In step (3) of M just say that the output is c', where c is the solution of LP. In the binary case this leaves M unchanged, while in the real-valued case this adds a trivial step at the end. Now we can prove the results for both cases at once.

In the following propositions, the probability is taken over the random choices $q_1, ..., q_{t(n)}$ made by M.

Lemma 8. *Assume (3.1) and let A be an assembled sanitizer with perturbation τ. Then $\forall \epsilon \in (0,1)$, $\forall n \ge n_\epsilon$, $\forall d \in D^n$,*

$$P[dist(M^{A_d}(1^n), d) \ge \epsilon\Delta^n_{dist}] \le (n+1)^n \cdot \max_{x \in X^{d,n}_\epsilon} \prod_{j=1}^{t(n)} P\big[\big|\sum_{i\in q_j}(x_i - d_i)\big| \le 2\tau(n)+1\big].$$

Proof. If c is the solution of LP found by M, then $c' = M^{A_d}(1^n)$. Hence

$$\begin{aligned} P[dist(c', d) \ge \epsilon\Delta^n_{dist}] &\le^{(1)} P\big[\hat{c} \in X^{d,n}_\epsilon\big] \\ &\le^{(2)} P\big[\exists x \in X^{d,n}_\epsilon : \forall j \in [t(n)], |\sum_{i\in q_j}(x_i - d_i)| \le 2\tau(n)+1\big] \\ &\le^{(3)} (n+1)^n \cdot \max_{x \in X^{d,n}_\epsilon} \prod_{j=1}^{t(n)} P\big[|\sum_{i\in q_j}(x_i - d_i)| \le 2\tau(n)+1\big]. \end{aligned}$$

(1) Follows from property (3.1).

(2) Put $x = \hat{c}$. Since c solves LP and A has perturbation τ, then $\forall j \in [t(n)]$,

$$\big|\sum_{i\in q_j}(\hat{c}_i - d_i)\big| \le \big|\sum_{i\in q_j}(\hat{c}_i - c_i)\big| + \big|\sum_{i\in q_j} c_i - A_d(q_j)\big| + \big|A_d(q_j) - \sum_{i\in q_j} d_i\big| \le 2\tau(n)+1$$

(3) Holds since $|X_\epsilon^{d,n}| \leq (n+1)^n$, and the $t(n)$ subsets q_j are chosen independently, turning the probability of "$\forall j \in [t(n)]$" into a product of probabilities. □

A similar result is stated in [6], after which the Disqualifying Lemma is applied, leading to the inequality $P[dist(M^{A_d}(1^n), d) \geq \epsilon n] \leq (n+1)^n(1-\delta)^{t(n)}$. Since the goal is to cause a privacy breach with M, meaning that the right hand side of the inequality is negligible with respect to n, it is enough to put $t(n) = n\log^2 n$.

Analogously, we combine Lemma 8 with Lemma 6. Since the latter is a stronger version of the Disqualifying Lemma, we can make the corresponding probability negligible with a lower value of $t(n)$.

Corollary 9. *Assume (3.1) holds and let A be an assembled sanitizer with perturbation τ. Let $\alpha_\epsilon = \frac{64}{\rho_\epsilon^3} + \frac{4}{\sqrt{2\pi}\rho_\epsilon}$ and $\beta_\epsilon = \frac{8}{\sqrt{2\pi}\rho_\epsilon}$. Then $\forall \epsilon > 0$, $\forall n \geq n_\epsilon$, $\forall d \in D^n$,*

$$P[dist(M^{A_d}(1^n), d) \geq \epsilon\Delta_{dist}^n] \leq (n+1)^n\Big(\frac{\alpha_\epsilon + \beta_\epsilon \cdot \tau(n)}{\sqrt{n}}\Big)^{t(n)}. \tag{4.1}$$

Proof. If $x \in X_\epsilon^{d,n}$, then $x - d \in I_{\rho_\epsilon}^n$, so by Lemma 6, $\forall j \in [t(n)]$,

$$P\big[\big|\sum_{i\in q_j}(x_i - d_i)\big| \leq 2\tau(n)+1\big] \leq \frac{\lambda_{\rho_\epsilon} + \mu_{\rho_\epsilon}\cdot(2\tau(n)+1)}{\sqrt{n}}.$$

Now, α_ϵ and β_ϵ were chosen to make $\lambda_{\rho_\epsilon} + \mu_{\rho_\epsilon}\cdot(2\tau(n)+1) = \alpha_\epsilon + \beta_\epsilon\cdot\tau(n)$, so:

$$\max_{x\in X_\epsilon^{d,n}} \prod_{j=1}^{t(n)} P\big[\big|\sum_{i\in q_j}(x_i - d_i)\big| \leq 2\tau(n)+1\big] \leq \prod_{j=1}^{t(n)}\Big(\frac{\alpha_\epsilon + \beta_\epsilon\tau(n)}{\sqrt{n}}\Big) = \Big(\frac{\alpha_\epsilon + \beta_\epsilon\tau(n)}{\sqrt{n}}\Big)^{t(n)}$$

and the result follows from Lemma 8. □

Remark 1. *In the general setting, when we only rely on (3.1), we cannot specify n_ϵ and ρ_ϵ any better, but we can in each particular case. In the binary case (see analysis at the end of Section 3) Lemma 9 holds $\forall n \geq n_\epsilon = 3$ and $\rho_\epsilon = \frac{\sqrt{\epsilon}}{3}$, which leads to $\alpha_\epsilon = \frac{1728}{\epsilon^{3/2}} + \frac{12}{\sqrt{2\pi\epsilon}} \leq \frac{1733}{\epsilon^{3/2}}$ and $\beta_\epsilon = \frac{24}{\sqrt{2\pi\epsilon}} \leq \frac{10}{\epsilon^{1/2}}$. The real-valued cases have analogous specializations.*

Corollary 10. *Assume (3.1) and let A be an assembled sanitizer with perturbation $\tau \in o(\sqrt{n})$. Then $\forall \epsilon > 0$, $\exists n_0 \in \mathbb{N}$ such that $\forall n \geq n_0$, $\forall d \in D^n$,*

$$P[dist(M^{A_d}(1^n), d) \geq \epsilon\Delta_{dist}^n] \leq 2^{-n}$$

for certain $t \in o(n\log n)$.

Proof. If $n \geq n_\epsilon$, then by Corollary 9 it suffices that $(n+1)^n\big(\alpha_\epsilon + \beta_\epsilon\cdot\tau(n)\big)/\sqrt{n}\big)^{t(n)} \leq 2^{-n}$, which is equivalent to $t(n) \geq \frac{n(1+\log_2(n+1))}{\log_2(\sqrt{n}/(\alpha_\epsilon+\beta_\epsilon\cdot\tau(n)))}$. Since $\tau \in o(\sqrt{n})$, there exists

n_1 depending on ϵ and τ such that $\forall n \geq n_1$, $\log_2(\sqrt{n}/(\alpha_\epsilon + \beta_\epsilon \cdot \tau(n))) > 0$. So put $n_0 = \max(n_\epsilon, n_1)$ and $t(n) = \lceil \frac{n(1+\log_2(n+1))}{\log_2(\sqrt{n}/(\alpha_\epsilon+\beta_\epsilon \cdot \tau(n)))} \rceil$, which is in $o(n \log n)$ since the denominator tends to infinity as $n \to \infty$. □

Corollary 9 describes the interplay among the relevant parameters of the model. Corollary 10 is an example of how those parameters can be adjusted according to given restrictions. For instance, if we want τ independent of ϵ, then decreasing ϵ (the accuracy of the adversary) makes t larger, increasing its time-complexity (as can be seen in the proof since $\epsilon \searrow 0$ makes $\alpha_\epsilon, \beta_\epsilon \nearrow \infty$, roughly).[3] Another example is that if $\tau \in O(n^{1/3})$ then t can be take in $O(n)$.

Although the function t in the previous proof is in $o(n \log n)$, t actually depends on ϵ, and as $\epsilon \searrow 0$ we have $n_0 \nearrow \infty$. This is not a problem since from a practical point of view, ϵ would be chosen appropriately and kept fixed.

Remark 2. *If $t(n)$ is bounded by a polynomial, then the adversary algorithm M is polynomial. When M receives the input 1^n, it creates $t(n)$ random subsets, which requires $n \cdot t(n)$ operations. Then it makes the corresponding $t(n)$ calls to the oracle and finally solves an LP of size $n \cdot t(n)$, for which there exists an algorithm finding a solution in poly$(n \cdot t(n))$ [25]. So if $t(n)$ is polynomial on n, the number of steps made by M is polynomial on n, the size of the input.*

Theorem 11. *Assume (3.1), let $\epsilon > 0$ and $\tau \in o(\sqrt{n})$. Then, any assembled sanitizer with perturbation τ causes an ϵ-privacy-breach.*

Proof. From Corollary 10, for any such assembled sanitizer A, there is a polynomial q_ϵ of degree 2 (with constants depending on ϵ) and an adversary M such that $\forall n$, $t(n) \leq q_\epsilon(n)$, and $\exists n_0$ (depending on ϵ) such that $\forall n \geq n_0$, $\forall d \in D^n$, $P[dist(M^{A_d}(1^n), d) \geq \epsilon \Delta^n_{dist}] \leq 2^{-n}$. M is a polynomial-time algorithm by Remark 2, and the function given by $f(n) = 2^{-n}$ if $n \geq n_0$ and $f(n) = 1$ otherwise, is negligible with respect to n. □

5 Concluding remarks

Dinur-Nissim proof of the Disqualifying Lemma. Our work follows the strategy of Dinur and Nissim [6]. Whereas [6] proves the Disqualifying Lemma using martingales and Azuma's inequality, we use the Berry-Esseen Theorem, which allows to obtain results for real-values databases as well. It may be possible to extend the (martingale) analysis of Dinur-Nissim to real-valued databases, however, our intention (and partly the motivation of this work) was to avoid Dinur-Nissim's proof of the Disqualifying Lemma, due to an incompatibility between two conditions needed on a constant T (see proof in the Appendix of [6]).

In the proof it is required that $1 - 2e^{-T^2/8} > 0$, which means $T > \sqrt{8 \ln 2} > 2$. Later, T is chosen to be $T = \sqrt{\frac{\alpha}{12}}$, with $\alpha = \frac{\sum_{i=1}^n (x_i - d_i)^2}{4n}$. Since $x, d \in [0,1]^n$, then

[3] It is reasonable to assume that in (3.1), as ϵ gets smaller, ρ_ϵ gets smaller, so α_ϵ and β_ϵ get larger.

$\alpha \leq \frac{1}{4}$, and therefore $T \leq \frac{1}{\sqrt{48}}$, which is incompatible with $T > 2$.

A different $\mathbf{D}^n$. Besides $D^n = \{0,1\}^n$ and $D^n = [0,1]^n$, our results apply to any $D^n \subseteq [0,1]^n$ provided that: (i) for each x we have well defined $x' \in D^n$ as the "best approximation" of x (whatever that means); (ii) the criteria for such an approximation is the same used to create c' in the last step of M; (iii) property (3.1) holds for the chosen criteria $x \mapsto x'$ and the *dist* function used. Cases other than $D^n = \{0,1\}^n$ and $D^n = [0,1]^n$ may be considered if the attacker has auxiliary information about d which induces a convenient choice for D^n.

Multiple attributes. It would be interesting to extend the attack to a database with many attributes. It seems possible to use the same attack on each column of a database of size $n \times m$, but it would be interesting to assume some dependency/correlation among the attributes (considered as random variables), and make the adversary algorithm exploit this fact.

In the one dimensional case, the adversary algorithm uses the true database d as certain property publicly known, namely, that all query responses have error bounded by τ. This property is imposed over certain queries $q_1, ..., q_t$ to the c that induces the output c'. In this way c' and d become related by the property $\left[\forall j \in [t] : |\sum_{i \in q_j}(c'_i - d_i)| \leq 2\tau + 1\right]$. The question is how to relate this property with an upper bound on $||c' - d||_2$. This is done with Lemma 6: if over a random query q, $P[|\sum_{i \in q}(c'_i - d_i)| \leq 2\tau + 1] > \frac{\lambda_\rho + \mu_\rho \cdot (2\tau+1)}{\sqrt{n}}$, then $||c' - d||_2 < \rho\sqrt{n}$. And since by construction of c it is true that for each q_j all such sums are $\leq 2\tau + 1$, then the probability considered in the hypothesis of Lemma 6 is high, making large the probability of $||c' - d||_2$ being small.

Thus, the attack is based on the fact that d has a property that is publicly known, allowing the attacker to efficiently create a good candidate. If now d has many attributes, then it may be possible to find out an extra and verifiable property that d has due to the high dimension and make the adversary exploit it to create its output. Then it could happen that the new relationship between c' and d becomes more restrictive, making $||c' - d||_2$ even smaller.

Utility. Theorem 11 establishes the limitation of the output perturbation sanitizer in producing privacy-protected sanitized data, when the perturbation is $o(\sqrt{n})$. This means that to provide privacy by this method, the perturbation must be higher than $o(\sqrt{n})$. It would be interesting to have a rigorous result establishing the relationship between the utility of perturbed data and the level of perturbation, and eventually show that sanitized data that has been perturbed more than $o(\sqrt{n})$ is significantly useless.

6 Appendix

To prove Lemma 6 we use a generalization of the Central Limit Theorem called Berry-Esseen Theorem (BET).

Theorem 12. ***BET.*** *Let $n \in \mathbb{N}$ and let $Y_1, ..., Y_n$ be independent random variables such that $\forall i \in [n]$, $E[Y_i]$, $Var(Y_i)$, and $E[|Y_i - E[Y_i]|^3]$ are finite. If $S_n = \sum_{i=1}^n Y_i$ and F_n is the distribution of $\frac{S_n - E[S_n]}{\sqrt{Var(S_n)}}$, then there exist a universal constant A such that:*

$$\sup_{x \in \mathbb{R}} |F_n(x) - \Phi(x)| \leq A \cdot \Gamma_n$$

where $\Gamma_n = \frac{\sum_{i=1}^n E[|Y_i - E[Y_i]|^3]}{Var(S_n)^{3/2}}$, $\Phi(x) = \int_{-\infty}^x \phi(t)dt$, and $\phi(t) = \frac{1}{\sqrt{2\pi}} e^{-t^2/2}$.

See [22] for a proof of this result, or [5] (Chap. 11) for more references. See [21] for a proof of the fact that A can be chosen as $A = 32$.

Lemma 6. Let $\rho \in (0, 1)$, let $n \in \mathbb{N}$, and $I_\rho^n = \{y \in [-1, 1]^n : ||y||_2 \geq \rho\sqrt{n}\}$. If $\lambda_\rho = \frac{64}{\rho^3}$ and $\mu_\rho = \frac{4}{\sqrt{2\pi}\rho}$, then $\forall \tau > 0$, $\forall y \in I_\rho^n$,

$$P\big[\big|\sum_{i \in q} y_i\big| \leq \tau\big] \leq \frac{\lambda_\rho + \mu_\rho \cdot \tau}{\sqrt{n}}$$

where the probability is taken with respect to a random $q \subseteq [n]$ obtained by n independent tosses of a fair coin to determine for each $j \in [n]$, if $j \in q$.

Proof. Fix $\rho \in (0, 1)$, $n \in \mathbb{N}$, $y \in I_\rho^n$, and $\tau > 0$. Let $Y_1, ..., Y_n$ be independent r.v.'s, such that Y_i takes values in $\{y_i, 0\}$ with uniform probability. Note that $\forall i \in [n]$, $E[Y_i] = \frac{1}{2} y_i$, $Var(Y_i) = \frac{1}{4} y_i^2$, and $E[|Y_i - E[Y_i]|^3] = \frac{1}{8}|y_i|^3$, so expectation, variance, and absolute third moment are finite, and we can apply Theorem 12 to $(Y_i)_{i \leq n}$. Denote $l_n = \frac{-\tau - E[S_n]}{\sqrt{Var(S_n)}}$ and $L_n = \frac{\tau - E[S_n]}{\sqrt{Var(S_n)}}$. Then:

$$P\Big[\big|\sum_{i \in q} y_i\big| \leq \tau\Big] = P\Big[|S_n| \leq \tau\Big] = P\Big[\frac{-\tau - E[S_n]}{\sqrt{Var(S_n)}} \leq \frac{S_n - E[S_n]}{\sqrt{Var(S_n)}} \leq \frac{\tau - E[S_n]}{\sqrt{Var(S_n)}}\Big] =$$

$$= P\Big[l_n \leq \frac{S_n - E[S_n]}{\sqrt{Var(S_n)}} \leq L_n\Big] = F_n(L_n) - \lim_{y \to l_n^-} F_n(y). \tag{6.1}$$

Since $\sup_x |F_n(x) - \Phi(x)| \leq A \cdot \Gamma_n$, we get:

$$|F_n(L_n) - \Phi(L_n)| \leq A \cdot \Gamma_n \tag{6.2}$$

and:

$$|\lim_{y \to l_n^-} F_n(y) - \lim_{y \to l_n^-} \Phi_n(y)| = \lim_{y \to l_n^-} |F_n(y) - \Phi_n(y)| \leq A \cdot \Gamma_n. \tag{6.3}$$

Recall that:

$$\Phi(L_n) - \lim_{y \to l_n^-} \Phi(y) = \int_{l_n}^{L_n} \phi. \tag{6.4}$$

Adding and subtracting terms appropriately, equations (6.2), (6.3) and (6.4) give:

$$F_n(L_n) - \lim_{y \to l_n^-} F_n(y) =$$

$$= F_n(L_n) - \Phi(L_n) + \Phi(L_n) - \lim_{y \to l_n^-} \Phi(y) + \lim_{y \to l_n^-} \Phi(y) - \lim_{y \to l_n^-} F_n(y) \leq$$

$$\leq 2A \cdot \Gamma_n + \int_{l_n}^{L_n} \phi$$

which together with (6.1) imply:

$$P\big[|\sum_{i \in q} y_i| \leq \tau\big] \leq 2A\Gamma_n + \int_{l_n}^{L_n} \phi \tag{6.5}$$

where $A \leq 32$ is given by BET. First we bound the integral. Since $\phi(t) = \frac{1}{\sqrt{2\pi}} e^{-t^2/2}$, then $\sup_{t \in \mathbb{R}} \phi(t) = \phi(0) = \frac{1}{\sqrt{2\pi}}$. Hence:

$$\int_{l_n}^{L_n} \phi \leq (L_n - l_n) \sup_{t \in \mathbb{R}} \phi(t) = \frac{1}{\sqrt{2\pi}} \frac{2\tau}{\sqrt{Var(S_n)}}.$$

The Y_i's are independent, so $Var(S_n) = \frac{1}{4} \sum_{i=1}^n y_i^2 = \frac{1}{4} ||y||_2^2$. Since $y \in I_\rho^n$, then:

$$\int_{l_n}^{L_n} \phi \leq \frac{1}{\sqrt{2\pi}} \frac{4\tau}{||y||_2} \leq \frac{1}{\sqrt{2\pi}} \frac{4\tau}{\rho\sqrt{n}}. \tag{6.6}$$

Next, bound Γ_n:

$$\Gamma_n = \frac{\sum_{i=1}^n E\big[|Y_i - E[Y_i]|^3\big]}{Var(S_n)^{3/2}} = \frac{\sum_{i=1}^n |y_i|^3}{||y||_2^3} \leq \frac{n}{(\rho\sqrt{n})^3} = \frac{1}{\rho^3} \frac{1}{\sqrt{n}}. \tag{6.7}$$

From equations (6.5), (6.6), (6.7) follows:

$$P\big[|\sum_{i \in q} y_i| \leq \tau\big] \leq \frac{64}{\rho^3\sqrt{n}} + \frac{4}{\sqrt{2\pi}\rho} \cdot \frac{\tau}{\sqrt{n}} = \frac{\lambda_\rho + \mu_\rho \cdot \tau}{\sqrt{n}}.$$

□

References

[1] Adam, N. R., Wortmann, J. C. (1989). Security-control methods for statistical databases: A comparative study. *ACM Computing Surveys*, 21(4):515–556.

[2] Agrawal, R., Srikant, R. (2000). Privacy-preserving data mining. *ACM SIGMOD Record*, 29(2):439–450.

[3] Blum, A., Dwork, C., McSherry, F., Nissim, K. (2005). Practical privacy: The SuLQ framework. In *Proceedings of the Twenty-fourth Symposium on Principles of Database Systems (PODS)*, 128–138.

[4] Dalenius, T. (1977). Towards a methodology for statistical disclosure control. *Statistik Tidskrift* 15, 429–444.

[5] DasGupta, A. (2008). *Asymptotic Theory of Statistics and Probability*. New York: Springer.

[6] Dinur, I., Nissim, K. (2003). Revealing information while preserving privacy. In *Proceedings of the Twenty-second Symposium on Principles of Database Systems (PODS)*, 202–210.

[7] Duncan, G. T., Pearson, R. W. (1991). Enhancing access to microdata while protecting confidentiality: Prospects for the future, *Statistical Sciences*, 6(3):219–232.

[8] Dwork, C. (2007). An ad omnia approach to defining and achieving private data analysis. In *PinKDD*, vol. 4890 of *LNCS*. Springer-Verlag. 1–13.

[9] Dwork, C. (2008). Differential privacy: A survey of results. In *Proceedings of the 5th TAMC*, 1–19.

[10] Dwork, C. (2006). Differential privacy. In *Proceedings of the 33rd International Colloquium on Automata, Languages and Programming (ICALP)*, (2):1–12.

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

[12] Dwork, C., McSherry, F., Talwar, K. (2007). The price of privacy and the limits of LP decoding. In *Proceedings of the Thirty-ninth Annual ACM Symposium on Theory of Computing*, 85–94.

[13] Dwork, C., Nissim, K. (2004). Privacy-preserving datamining on vertically partitioned databases. In *CRYPTO*, vol. 3152 of *LNCS*. Springer-Verlag. 528–544.

[14] Dwork, C., Yekhanin, S. (2008). New efficient attacks on statistical disclosure control mechanisms. In *Proceedings of the 28th Annual Conference on Cryptology*, vol. 5157 of *LNCS*, 469–480.

[15] Evfimievski, A. (2002). Randomization in privacy preserving data mining. *ACM SIGKDD Explorations Newsletter*, 4(2):43–48.

[16] Fellegi, I. P. (1972). On the question of statistical confidentiality. *Journal of the American Statistical Association*, 67(337):7–18.

[17] Hardt, M., Talwar, K. (2010). On the geometry of differential privacy. In *Proceedings of the 42nd ACM Symposium on Theory of Computing*, 705–714.

[18] Kasiviswanathan, S. P., Rudelson, M., Smith, A., Ullman, J. (2010). The price of privately releasing contingency tables and the spectra of random matrices with correlated rows. In *Proceedings of the 42nd ACM Symposium on Theory of Computing*, 775–784.

[19] Nissim, K. (2008). Private data analysis via output perturbation. In *Privacy-preserving data mining, models and algorithms*, 383–414.

[20] Nissim, K., Raskhodnikova, S., Smith, A. (2007). Smooth sensitivity and sampling in private data analysis. In *Proceedings of the Thirty-ninth Annual ACM Symposium on Theory of Computing*, 75–84.

[21] Paditz, L. (1989). On the analytical structure of the constant in the nonuniform version of the Esseen inequality. *Statistics*, 20(3):453–464.

[22] Petrov, V. V. (1995). Limit Theorems of Probability Theory. Oxford University Press.

[23] Reiter, J. P. (2003). Inference for partially synthetic, public use microdata sets. *Statistics Canada*, 29(2):181–188.

[24] Rubin, D. B. (1993). Discussion: Statistical disclosure limitation, *Journal of Official Statistics*, 9(2):461–468.

[25] Tardos, E. (1986). A strongly polynomial algorithm to solve combinatorial linear programs. *Operations Research*, 34(2):250–256.

[26] Willenborg, L., Waal de, T. (2001). *Elements of Statistical Disclosure Control*. Springer-Verlag.