The tension between anonymity and privacy

Staal A. Vinterbo*

Department of Information Security and Communication Technology
Norwegian University of Science and Technology
Staal.Vinterbo@ntnu.no

Abstract

Privacy in the context of information and data is often defined in terms of anonymity, particularly in regulations such as the GDPR. Operationally, it is appealing to define privacy in terms of computable data properties as this makes it possible to verify compliance. A well known example of privacy defined as such is k-anonymity. At the same time, uncertainty regarding real-world privacy is increasing with the amount of data collected about us all. We present arguments for why focusing on anonymity or computable properties of data is not very helpful in this regard. In particular, we count exploitable failures of privacy defined in terms of computable properties of n-bit data and conclude that these counterexamples to protection cannot be rare.

1 Introduction

Many privacy regulations, including the General Data Protection Regulation (GDPR) [1] and the US Health Insurance Portability and Accountability Act (HIPAA) [18], have anonymity as a decision criterion of whether they apply to the contents of dataset or not.

Now, consider a colleague showing you data and asking “Is this dataset anonymous?”, effectively asking you if it can be shared without running afoul of privacy regulations. Ethics and potential personal harm aside, getting the answer wrong can have financial and legal consequences. Especially as privacy regulations grow teeth, as they are doing in Europe, where the upcoming GDPR threatens with significant penalties. Unfortunately, relying on a positive answer to this question is problematic.

Uncertainty about anonymization and privacy features prominently among barriers to efficient use of information collected from individuals [16, 17]. A perhaps non-obvious reason is that anonymity and anonymization strongly suggests a focus on prohibiting a one-to-one relation from data to identity, while actually providing what most of us associate with privacy requires prohibiting more general inferences. Intuitively, instead of asking “can I figure out who this data comes from” we have to ask “what new inferences about anyone can I make using this data”. As we will see, addressing the latter is difficult and puts additional constraints on how information can be shared. In particular, collecting and sharing anonymized data becomes difficult.

Our goal here is to substantiate the above with simple yet somewhat formal arguments. We also briefly present how differential privacy [9], an emerging standard for data privacy, relates to the identified challenges. From a technical perspective, our main contribution is a quantitative argument that failures of checkable privacy cannot be rare (Section 5.1 and Theorem 5.4).

2 Why anonymity

The distinction between the public and private was understood in Greece and China 400 BC [15]. In the early Renaissance in Europe, the notion of the home as a protected and sovereign sphere

*The author presented this paper at the NISK 2018 conference.

Proceedings of the 11th Norwegian Information Security Conference, Sep. 19-20, 2018.
NISK Journal, e-ISSN 1894-7735. https://ojs.bibsys.no/index.php/NISK

was documented already in 1499 [21]. This idea is also found in the Fourth Amendment of the US Constitution, with a focus on protections from the government. In 1890, Warren and Brandeis published the very influential “The right to privacy”, where they reacted to the newspapers’ overstepping bounds of propriety and decency, particularly with photographs becoming available. Warren and Brandeis declared a “right to be left alone”, laying the foundation of what Ruth Gavison much later defends as the right to hide in the crowd [13].

This view of privacy as anonymity has been widely adopted in regulations. For example, the upcoming (May 2018) GDPR explicitly states:

> “The principles of data protection should therefore not apply [. . .] to personal data rendered anonymous in such a manner that the data subject is not or no longer identifiable.”

From a more technical point of view, anonymity is violated by relating a piece of information or trait to a single identity. A way of preventing this is introducing ambiguity. For data about people this can be stated as:

> if enough people share a trait, then it is not identifying.

The beauty of this idea is that we can use data we have to establish a lower bound on how many people share a trait, or several traits. Sweeney’s well known k-anonymity [22] parameterizes sufficient ambiguity for anonymity as k people having to share all present combinations of values for traits. In particular, introducing ambiguity destroys 1-to-1 relations. This was the explicit motivation behind the definition of k-anonymity [23].

As an example, consider the data in Table 1 where the possession and absence of two traits a and b for five people are indicated by 1 and 0, respectively.

Table 1: Two traits and five people

trait	Alice	Bob	Graham	Denise	Eric
a	1	1	0	0	0
b	1	1	1	0	0

We see that for the individual traits, at least two people share both possession and absence. In terms of k-anonymity, the data is 2-anonymous in a and b individually. We also see that only Graham exhibits absence of a and possession of b. Consequently, his pattern of absence and possession is not “anonymous”. If we were to remove Graham from the data, it would be 2-anonymous (in a and b jointly).

In general, definitions of privacy that we can check are called *syntactic*. From the perspective of sharing data, being able to check privacy compliance avoids the burden of having to document data provenance, which in turn could be sensitive.

In summary, anonymity represents a tradition in jurisprudence that is easy to operationalize by syntactical means.

3 The insufficiency of prohibiting 1-to-1 relations

Consider the following story of Alice and Bob. Alice works at the local hospital as an analyst. Lately, she has been working with researchers investigating the connection between HIV and diabetes. Specifically, her work has been to answer the question whether the hospital already has sufficient patient data for a study, and whether recruiting external prospective study participants is necessary. To accomplish this, she has been given access to counts of how many patients in the hospital database have been diagnosed as diabetic, HIV positive and both simultaneously, across gender and age. Last Saturday, she was invited to her neighbor Bob's 40th birthday BBQ. During their chat while Bob was flipping burgers, Bob mentioned that he loves Coca Cola, but that his doctor at the local hospital had told him that he has to be more careful about controlling his diabetes. Prompted by this comment, Alice, more of a beer drinker herself, recalls that there were no male, age 35-40, diabetic, and HIV negative patients in the hospital database a month ago.

The information Alice obtained through hospital data access can be described as the implication $(a \implies b)$, where a is true for males, age 35-40, that are diabetic, and b is being HIV positive. A logically equivalent formulation of the implication is $\neg(a \wedge \neg b)$. This means that if $(a \implies b)$ is true for all elements in the database (note that this is different from a being true for all), then $(a \wedge \neg b)$ is true for none. Neither allows establishing a unique relationship with anyone in the database. Note that $(a \implies b)$ holds for all people in Table 1.

It is only through Bob telling Alice that he is in the database and that a is true for him that Alice is able to infer that Bob is HIV positive. Again, neither of the pieces of information Bob provides represent a one-to-one relationship.

In our above example, $(a \wedge \neg b)$ was true for none. Can we consider the information that something does not exist (in the database), personal information or data? This question is interesting in terms of language semantics, as the GDPR emphasizes that personal data is data related to a real living person.

4 Privacy as protection against adversarial inference

As we have seen above, ambiguity as a privacy criterion is not sufficient. What we arguably care about is any inference about any individual. Paraphrasing Dalenius [4], disclosure by information v happens if we can use it to learn something about x we did not already know. We can define this notion a little more formally using probabilities as follows.

Definition 4.1 (Disclosure)**.** Let M_r be a randomized inference "machine" for a property r. Disclosure of r for object x by information v happens if

$$P[M_r(x,v) = r(x)] > P[M_r(x) = r(x)].$$

Furthermore, we can think of disclosure as *direct* if $r(x)$ can be determined from v alone, or *indirect* if more information and inferential machinery is needed.

We can now cast anonymity of data in terms of disclosure as follows. Let property r be identity, i.e., $r_{id} = x \mapsto \text{Identity}$. Then we can say that data v is anonymous if

$$(\forall x)\ P[M_{r_{id}}(x,v) = r_{id}(x)] \leq P[M_{r_{id}}(x) = r_{id}(x)].$$

Generalizing a little, we can think of r as being any sensitive information. Of course, what sensitive information means is subjective. For example, not all HIV positive patients consider their status as sensitive in all contexts [20].

Unfortunately, as Dalenius informally, and Dwork later formally argues [6], eliminating disclosure while providing useful information is impossible. Intuitively, no information computed from data about individuals can be independent of this data and reflect the contents of the data at the same time. Consequently, we can think of privacy as *controlling* disclosure, or alternatively, controlling how much a particular piece of information aids inferences about someone. Abandoning the prohibition of disclosure also signals a departure from thinking of privacy as controlling access to crisply circumscribed information, a goal of information security.

The notion of indirect disclosure invites the question of what resources an adversary has at her disposal. Knowing what information an adversary can use for inference generally requires omniscience. In order to avoid uncertainty due to assumptions, we have to make the strongest assumption possible. This assumption is that the adversary has enough information to reduce her task to a choice between two alternatives. Importantly, we do not know which alternatives those are.

Not knowing which pair of alternatives the adversary is left to decide between makes it impossible to distinguish between sensitive and non-sensitive properties. We cannot dismiss the possibility that a non-sensitive property allows us to rule out one of the two remaining alternatives. An instance of this problem is deciding which attributes in a data table do not help the adversary when linking to other tables.

One could ask whether such a strong adversary is esoteric enough to not matter in practice. Again, this is hard to know as an adversary will manipulate the context to her advantage. One way of doing this is in terms of an “intersection attack” where a set of a priori known candidate hypotheses is intersected with the hypotheses corresponding to a given response [12]. Furthermore, the specter of Russian manipulation of the 2016 US election could be taken as a cautionary tale against underestimating the resources of an adversary. In particular, this emphasizes the importance of reducing the reliance on assumptions.

5 Syntactic privacy

Since only considering one-to-one relationships such as identity is insufficient, can we abandon the anonymity tradition but somehow keep the operational advantage of checkable privacy? Unfortunately, barring a definition where essentially all data is sensitive, we can always find a formal example of privacy breach for any syntactic definition of privacy. Moreover, as we will see, such counterexamples cannot be rare.

We will consider databases and encodings of information somewhat interchangeably as finite bit-strings, i.e., elements from the set $\{0,1\}^*$.

Definition 5.1 (Syntactic definition of privacy)**.** A function $\sigma : \{0,1\}^* \to \{0,1\}$ that returns 1 if its input is sensitive and 0 otherwise is a *syntactic definition of privacy.*

We will assume that all our syntactic privacy definitions are computable. Importantly, computable σ means that syntactic privacy captures the notion of checkable privacy of data. From a disclosure control standpoint, syntactic privacy allows for deciding whether a given statistic (or data) causes disclosures with some likelihood.

Anna has a database that contains sensitive information and wants to answer Ben’s query, but without sharing sensitive information. A *sanitizer* is a mechanism by which she can extract the queried information from the data in a safe manner.

Definition 5.2 (Deterministic sanitizer)**.** Given a non-constant syntactic definition of privacy σ, an algorithm that computes function $f : \{0,1\}^* \to \{0,1\}^*$ is a *deterministic sanitizer* for σ if $\sigma(f(x)) = 0$ for all $x \in \{0,1\}^*$.

We will think of a sanitizer as a response algorithm for a query q on a bit-string database. We generally want the answer that is most useful for that query, i.e., that the function f

computed by the sanitizer approximates the function q well. This is difficult in general. For example, finding the least generalized k-anonymous database is NP-hard and it is known that an approximation that is not worse than $O(k \log k)$ times than needed can be computed, however with a running time exponential in k [14].

Now consider the situation where Anna has a sensitive database $s0$ where s is a bit string of length $n-1$. Also, let Ben know s, meaning he knows all bits of Anna's database except the last one. For sanitizer f to be safe, Anna should be able to give Ben $f(s0)$ without divulging the value of the last bit. If f is deterministic and $f(s0) \neq f(s1)$, then Ben can infer what Anna's database is by checking which of $s0$ and $s1$ yield the value $f(s0)$ he receives from Anna. This leads us to the following definition.

Definition 5.3 (Counterexample)**.** Let f be a deterministic sanitizer for definition of privacy σ on n-bit databases. Any pair of databases x and y differing in one bit such that $\sigma(x) + \sigma(y) > 0$ and $f(x) \neq f(y)$ constitutes a *counterexample* of f. Furthermore, if (x, y) is a counterexample for any sanitizer for σ, (x, y) is a counter-example of σ.

In terms of our discussion of adversaries in Section 4, a counterexample consists of two specific alternative hypotheses that an adversary can reduce to a single correct hypothesis.

Unless otherwise indicated, we consider sanitizers to be deterministic. The reason for this is that if we rely on non-determinism or randomness for privacy, we go beyond what can be checked[1]. We now describe the syntactic definitions of privacy that allow counterexamples.

Definition 5.4 (Useful syntactic privacy)**.** A non-constant syntactic definition of privacy is *useful* if there are at least two non-sensitive databases.

The reason for Definition 5.4 is that a definition that is not useful, only allows sanitizers that are constant (trivial), and therefore useless.

Variations of the Anna and Ben example above are common in presentations on theory about disclosure control and differential privacy. The following theorem is a formalization of the idea behind these examples, and can therefore be considered a "folklore" theorem.

Theorem 5.1 (Folklore)**.** *For any useful syntactic definition σ of privacy on n-bit databases, there exists a counterexample.*

Proof. There exists a pair (x, y) of n-bit databases for which σ yields different values since σ is non-trivial. We can create a sequence $(x_0, x_1, \ldots, x_k)$ such that $k \leq n$, $x_0 = x$, $x_k = y$, and x_{i+1} equals x_i with a single bit inverted. Since $\sigma(x_0) \neq \sigma(x_k)$, there must exist i such that $\sigma(x_i) \neq \sigma(x_{i+1})$. Since σ is useful, there exists non-sensitive distinct databases u and v. Let sanitizer f be such that $u = f(x_i) \neq f(x_{i+1}) = v$. The pair (x_i, x_{i+1}) is therefore a counterexample. □

5.1 Counting counterexamples

Being completely safe is trivial: respond to queries using a constant sanitizer. This is generally not a helpful observation since sharing information is the the reason we are interested in sanitizers in the first place. Providing utility implies the ability to distinguish between databases. But, from the proof of Theorem 5.1, being able to discriminate between neighboring databases can yield counterexamples. A natural question now is whether we can find a suitable syntactic definition of privacy that allows answering many questions but only has few counterexamples. If

[1] more precisely, a given finite string cannot be proven random [3].

such a definition exists, then we could argue that since the number of counterexamples is low, Theorem 5.1 does not matter in practice. We approach this question by showing that only a negligible fraction of syntactic definitions on n-bit databases do *not* exhibit a full complement of counterexamples.

5.1.1 Syntactic privacy on the n-cube

For each n-bit database $x \in \{0,1\}^n$ there are n other n-bit databases that differ from x in a single bit. We denote that two databases $x, y \in \{0,1\}^n$ differ in a single bit by $x \sim y$, and call them neighbors. If we take the set of all n bit databases and connect all the neighboring n bit databases by an edge, we get a hypercube graph, or n-cube.

Define the weight w of a database $x \in \{0,1\}^n$ to be the number of 1 bits in it, i.e., $w(b_1, b_2, \ldots, b_n) = \sum_{i=1}^n b_i$. Now define the balance of x as $\beta(x) = (-1)^{w(x)}$. The balance of x tells us whether the weight of x is odd or even with values -1 and 1, respectively.

Proposition 5.2. *For any $S \subset \{0,1\}^n$ such that $\exists\ u, v \notin S\ \beta(u) \neq \beta(v)$, there exists $f : \{0,1\}^n \to \{0,1\}^n$ such that*

$$\forall x \in S, \forall y \in \{0,1\}^n (x \sim y \implies f(x) \neq f(y)).$$

Proof. By assumption we can fix $u, v \in \{0,1\}^n$ such that $\beta(u) \neq \beta(v)$. Now define $f : \{0,1\}^n \to \{0,1\}^n$ as

$$f(x) = \begin{cases} u & \text{if } \beta(x) = \beta(u) \\ v & \text{otherwise,} \end{cases}$$

For all $x, y \in \{0,1\}^n$ we have

a. $x \sim y \implies \beta(x) \neq \beta(y)$
since $x \sim y \implies |w(x) - w(y)| = 1$.
b. $\beta(x) = \beta(f(x))$
by definition of f.
c. $\beta(x) \neq \beta(y) \implies x \neq y$
since $\beta(x) \neq \beta(y) \implies w(x) \neq w(y)$ and $w(x) \neq w(y) \implies x \neq y$.

Combining a., b., and c., we get

$$x \sim y \overset{a.}{\implies} \beta(x) \neq \beta(y) \overset{b.}{\implies} \beta(f(x)) \neq \beta(f(y)) \overset{c.}{\implies} f(x) \neq f(y).$$ □

Recall that we have $m = 2^n$ different n-bit databases. There are 2^m different subsets S of databases, and one of them is empty, and one of them is $\{0,1\}^n$. Consequently, there are $2^m - 2$ non-empty proper subsets of $\{0,1\}^n$. We now turn to how many of these do not fit the requirement for application of Proposition 5.2.

Proposition 5.3. *Let $m = 2^n$. Then there are $2^{\frac{m}{2}+1} - 2$ non-empty sets $T \subseteq \{0,1\}^n$ such that $x, y \in T \implies \beta(x) = \beta(y)$.*

Proof. Let $U_j = \{x \in \{0,1\}^n | \beta(x) = j\}$. Then

a. $U_1 \cup U_{-1} = \{0,1\}^n$
b. $U_1 \cap U_{-1} = \emptyset$

c. $|U_i| = 2^{\frac{m}{2}}$ for $i \in \{-1, 1\}$

Let T be non-empty such that $x, y \in T \implies \beta(x) = \beta(y)$. Since this means that all elements in T must have the same balance, we have that $T \subset U_i$, where i is this shared balance. The converse is also trivially true. This means that T can be any non-empty subset of either U_{-1} or U_1. From c. we have that U_{-1} and U_1 each have $2^{\frac{m}{2}}$ subsets, out of which one is empty. Consequently, there are $2(2^{\frac{m}{2}} - 1) = 2^{\frac{m}{2}+1} - 2$ possible non-empty sets $T \subseteq \{0,1\}^n$ such that $x, y \in T \implies \beta(x) = \beta(y)$. □

Since each syntactic privacy definition σ is uniquely defined by its set S of sensitive databases, we have that there are $2^m - 2$ non-trivial such definitions. Proposition 5.2 tells us that if $T = \{0,1\}^n - S$ contains two elements that have different balance, we can find a sanitizer that for all elements in S discerns this element from all its neighbors. Proposition 5.3 tells us that there are at most $2^{m/2+1} - 2$ non-empty sets T where all elements have equal balance. Since $1/(x-2) \leq 2/x$ for $x \geq 4$, and since $n \geq 1 \implies m \geq 2$ which in turn implies $2^m \geq 4$, we get

$$\begin{aligned} \frac{2^{m/2+1} - 2}{2^m - 2} &\leq 2 \cdot 2^{m/2} \frac{1}{2^m - 2} \\ &\leq 2^{m/2} \frac{4}{2^m} = \frac{4}{(\sqrt{2})^m} = \frac{4}{(\sqrt{2})^{2^n}}. \end{aligned} \tag{1}$$

This means that the fraction of useful σ's that Proposition 5.2 does *not* apply to is exponentially small in m and doubly exponentially small in n.

We now summarize the above as follows.

Theorem 5.4. *For all but a negligible fraction of possible non-trivial syntactic definitions of privacy on n-bit databases, there exists a sanitizer such that every sensitive database is a part of n counterexamples.*

Proof. By Propositions 5.2 and 5.3, (1), that $x \mapsto 4(\sqrt{2})^{-x}$ is a function that decreases super-polynomially, and the isomorphy between proper non-empty subsets of $\{0,1\}^n$ and non-trivial syntactical definitions of privacy σ. □

In the above, we constrained the adversary to only consider hypotheses pairs arising from single bit differences under a fixed encoding of n-bit data. Doing this results in a much weaker adversary than the adversary we described in Section 4. Theorem 5.4 tells us that almost all syntactical definitions of privacy allow at least one way of answering questions that exposes *every* sensitive database to *all* its possible vulnerabilities for this constrained and much weaker adversary.

6 Differential privacy and the single unknown bit

The way to avoid the above problem is to introduce uncertainty into the inference sketched above. In other words, given bit string s and an unknown bit b, there should be uncertainty whether $f(sb) = f(s0)$ or $f(sb) = f(s1)$. This means that f cannot be deterministic and consequently must be randomized. We can think of f as a random algorithm that on input d first chooses a probability density or mass p_d and then returns a random variate according to this. Note that we can let the choice of density or mass be deterministic so that only the returned variate is chosen randomly.

Now, let f be randomized and let $L_i(y) = P(f(si) = y) = p_{si}(y)$ for $i \in \{0, 1\}$ describe the likelihood of $b = i$ on receiving $y = f(sb)$, and let

$$\Lambda_i(y) = \frac{L_i(y)}{L_{1-i}(y)} = \frac{P(f(si) = y)}{P(f(s(1-i)) = y)}.$$

From a Bayesian perspective, $\Lambda_i(y)$ describes the degree to which we on seeing y should update our a priori preference of $b = i$ over $b = 1-i$. Alternatively, from a hypothesis testing perspective, the likelihood ratio $\Lambda_i(y)$ is the test statistic used to determine whether to reject hypothesis $b = i$, and for simple hypotheses such as ours, the Neyman-Pearson lemma states that this test is the most powerful. Consequently, we can interpret $\Lambda(y) = \max_{i\in\{0,1\}} \Lambda_i(y)$ to represent the upper probabilistic bound on disclosure of b from y. The closer $\Lambda(y)$ is to 1, the less we learn about b from y. Differential privacy generalizes this bound to all databases and single record differences.

Let a record be an element from a set V, and let two databases $d_1, d_2 \in V^n$ for some positive integer n be neighbors if they differ at most in a single record.

Definition 6.1 (Differential Privacy [9])**.** A randomized algorithm f taking input from V^n and range W is ϵ-differentially private if

$$\frac{P(f(d_1) \in S)}{P(f(d_2) \in S)} \leq \exp(\epsilon)$$

for any neighboring databases $d_1, d_2 \in V^n$ and measurable $S \subseteq W$.

Worth noting is that in order to prove the above bound, the probabilities must be taken over what we can control, which is the randomness in the algorithm as opposed to the randomness in the data. Consequently, the above definition is independent of the data, and we cannot check whether a given value was produced in a differentially private manner. Furthermore, as suggested by the Bayesian view presented above, the differential privacy bound is valid independently of any a priori knowledge.

If we want a particular function q computed from data d, it makes sense to choose sanitizer f such that its output is concentrated around $q(d)$. If we do this, f is a randomized version of the query response algorithm for q. Much research into differential privacy goes into designing ϵ-differentially private versions of query response algorithms that maximize the concentration around $q(d)$ for some q under the ϵ constraint. Consequently, the parameter ϵ represents a "knob" that trades off usefulness (concentration) against the ability to infer something about an individual, i.e., privacy.

Two further important properties that differential privacy has are: additive composition and post-processing invariance. We restate these more formally as (for more in-depth discussion and proofs see, e.g., [10]) follows.

Theorem 6.1 (Composition of differential privacy [8])**.** *Let f_1 and f_2 be algorithms that are differentially private with ϵ_1 and ϵ_2, respectively. Then, the query $q(D_1, D_2) = (y, f_2(y, D_2))$ for $y = f_1(D_1)$ on any two databases D_1 and D_2 is $(\epsilon_1 + \epsilon_2)$-differentially private.*

The composition property means that the utility – privacy knob we have in the parameter ϵ can be used to adaptively budget for accumulated privacy "expenditure" incurred over time across different queries and databases. This is particularly important as Dinur and Nissim showed in 2003 that there is only a finite and even small number of questions we can answer about a database in a useful way before we start leaking potentially sensitive information [5]. A nice example of how synergy by collaboration is achieved is given by Sarwate et al. [19].

Theorem 6.2 (Post processing for differential privacy [10])**.** *For any non-private randomized algorithm g on databases, if f is ϵ-differentially private, then $h(x) = g(f(x))$ is ϵ-differentially private.*

The post-processing property means that we can use differentially private results in any manner we wish without losing the privacy protection that differential privacy gives.

6.1 Revisiting Alice and Bob

Returning to the example involving Alice and Bob from Section 3, we can restrict Alice to receive a database count (i.e., number of rows in a database for which some predicate is true) as a variate from a Laplace distribution centered on the true count with standard deviation proportional to ϵ^{-1}. Adding carefully chosen Laplace noise to query results is known as the Laplace Mechanism [9]. Due to the composition properties of differential privacy we can keep track of the overall inference likelihood change that Alice accrues even when she uses multiple queries. For Alice, the utility – privacy trade-off knob ϵ means that being relatively insensitive to individuals does not imply poor population statistics.

6.2 Revisiting disclosure control

We can use the post-processing property to close the circle back to Section 4 and Definition 4.1 of disclosure in terms of an inference machine M_r for a property r.

Proposition 6.3 (Disclosure control by differential privacy)**.** *If f is ϵ-differentially private, then for any neighboring databases $d, d' \in V^n$ and any x, r and M_r,*

$$P[M_r(x, f(d)) = r(x)] \leq \exp(\epsilon) P[M_r(x, f(d')) = r(x)].$$

Proof. Fix x, r, and M_r, and let $g(y) = M_r(x, y)$. The theorem then follows directly from Theorem 6.2. □

In particular, Proposition 6.3 holds for any x such that $x \in d$ but $x \notin d'$. This means that someone wanting to recruit for a study can say that "any disclosure about you will become at most $\exp(\epsilon)$ more likely on you entering the study as we are only releasing ϵ-differentially private computations." Importantly, differential privacy simultaneously holds for all properties r, including identity. Consequently, we do not need to take potentially subjective choices regarding sensitivity of properties or attributes into account.

6.3 Limitations

While many types of questions about data can be answered well with differential privacy, there are questions that are hard to answer with reasonable accuracy if we require differential privacy. In general, this applies to queries q that are very sensitive to single point substitutions. Examples of such include questions regarding connectivity in graphs; a single node deletion can change connectivity radically. Theoretical impossibility results also exist, for example regarding useful threshold queries on infinite domains [2]. What is not clear is whether these negative results are exclusive to differential privacy or are inherent to a larger notion of "strong" privacy.

7 Discussion

As we have argued above, anonymity and syntactic privacy will always leave doubt regarding protection of privacy. This situation is not helped by empirical risk analyses in support of syntactic approaches that implicitly only consider 1-to-1 relationships [11]. Defining privacy in terms of randomization independent of the data avoids the problems with anonymity and syntactic privacy, and allows answering questions regarding privacy risk quantitatively. However, a challenge with this lies in that such definitions are incompatible with current information exchange that depends on sharing data that has been anonymized according to some plausibly checkable criterion. Such exchange supports the massive data broker industry and is crucial to current secondary use of health information [24]. In this, abandoning a focus on anonymity and syntactic privacy represents a potentially expensive paradigm shift.

The ability to quantify privacy risk is also relevant for public policy formation. It is a requirement for making decisions based on rational risk – benefit analyses where we need to quantify both sides reliably. Such rational support for privacy policy might become even more important if trust in public management of population data erodes.

Worth noting when considering the above is that there is no necessary contradiction between strong, quantifiable privacy and utility, in fact it can enable use that is otherwise difficult [7]. That said, modern approaches to data and informational privacy such as differential privacy are not a technical panacea. It seems clear that protection of privacy will always require regulatory and contractual means. Nevertheless, we should strive to continually identify applications and areas where we can apply the strongest technical protections available.

References

[1] Home Page of EU GDPR, 2017.

[2] Mark Bun, Kobbi Nissim, Uri Stemmer, and Salil Vadhan. Differentially Private Release and Learning of Threshold Functions. *arXiv:1504.07553 [cs]*, April 2015.

[3] Gregory. J. Chaitin. Randomness and Mathematical Proof. *Scientific American*, 232:47–52, May 1975.

[4] Tore Dalenius. Towards a methodology for statistical disclosure control. *Statistik Tidskrift*, 15(429-444):2–1, 1977.

[5] Irit Dinur and Kobbi Nissim. Revealing information while preserving privacy. In *PODS '03: Proceedings of the Twenty-Second ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pages 202–210, New York, NY, USA, 2003. ACM.

[6] Cynthia Dwork. Differential Privacy. In *Proceedings of the 33rd International Conference on Automata, Languages and Programming - Volume Part II*, ICALP'06, pages 1–12, Berlin, Heidelberg, 2006. Springer-Verlag.

[7] Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. The reusable holdout: Preserving validity in adaptive data analysis. *Science*, 349(6248):636–638, August 2015.

[8] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our Data, Ourselves: Privacy Via Distributed Noise Generation. In *Advances in Cryptology (EUROCRYPT 2006)*, volume 4004, pages 486–503, Saint Petersburg, Russia, May 2006. Springer Verlag.

[9] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating Noise to Sensitivity in Private Data Analysis. In *Proceedings of the Conference on Theory of Cryptography*, 2006.

[10] Cynthia Dwork and Aaron Roth. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, August 2014.

[11] Khaled El Emam and Fida Kamal Dankar. Protecting Privacy Using k-Anonymity. *Journal of the American Medical Informatics Association*, 15(5):627–637, September 2008.

[12] Srivatsava Ranjit Ganta, Shiva Prasad Kasiviswanathan, and Adam Smith. Composition Attacks and Auxiliary Information in Data Privacy. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 265–273, New York, NY, USA, 2008. ACM.

[13] Ruth Gavison. Privacy and the Limits of Law. *The Yale Law Journal*, 89:421–471, 1980.

[14] Adam Meyerson and Ryan Williams. On the Complexity of Optimal K-anonymity. In *Proceedings of the Twenty-Third ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '04, pages 223–228, New York, NY, USA, 2004. ACM.

[15] Barrington Moore. *Privacy: Studies in Social and Cultural History.* M.E. Sharpe, 1984.

[16] Paul Ohm. Broken Promises of Privacy: Responding to the Surprising Failure of Anonymization. SSRN Scholarly Paper ID 1450006, Social Science Research Network, Rochester, NY, August 2009.

[17] Willem G. van Panhuis, Proma Paul, Claudia Emerson, John Grefenstette, Richard Wilder, Abraham J. Herbst, David Heymann, and Donald S. Burke. A systematic review of barriers to data sharing in public health. *BMC Public Health*, 14:1144, November 2014.

[18] Office for Civil Rights (OCR). Privacy, May 2008.

[19] Anand D. Sarwate, Sergey M. Plis, Jessica A. Turner, Mohammad Reza Arbabshirani, and Vince D. Calhoun. Sharing privacy-sensitive access to neuroimaging and genetics data: A review and preliminary validation. *Frontiers in Neuroinformatics*, 8, 2014.

[20] Cynthia Schairer, Sanjay R. Mehta, Staal A. Vinterbo, Martin Hoenigl, Michael Kalichman, and Susan Little. Perceptions of molecular epidemiology studies of HIV among stakeholders. *Journal of Public Health Research*, 6(3), December 2017.

[21] Daniel J. Solove. A Brief History of Information Privacy Law. SSRN Scholarly Paper ID 914271, Social Science Research Network, Rochester, NY, July 2006.

[22] Latanya Sweeney. K-anonymity: A Model for Protecting Privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5):557–570, October 2002.

[23] Latanya Sweeney. Personal communication, December 2010.

[24] Adam Tanner. Strengthening Protection of Patient Medical Data, January 2017.