Crowd-Blending Privacy

Johannes Gehrke*, Michael Hay**, Edward Lui, and Rafael Pass***

Department of Computer Science, Cornell University
{johannes,mhay,luied,rafael}@cs.cornell.edu

Abstract. We introduce a new definition of privacy called *crowd-blending privacy* that strictly relaxes the notion of differential privacy. Roughly speaking, k-crowd blending private sanitization of a database requires that each individual i in the database "blends" with k other individuals j in the database, in the sense that the output of the sanitizer is "indistinguishable" if i's data is replaced by j's.
We demonstrate crowd-blending private mechanisms for histograms and for releasing synthetic data points, achieving strictly better utility than what is possible using differentially private mechanisms. Additionally, we demonstrate that if a crowd-blending private mechanism is combined with a "pre-sampling" step, where the individuals in the database are randomly drawn from some underlying population (as is often the case during data collection), then the combined mechanism satisfies not only differential privacy, but also the stronger notion of zero-knowledge privacy. This holds even if the pre-sampling is slightly biased and an adversary knows whether certain individuals were sampled or not. Taken together, our results yield a practical approach for collecting and privately releasing data while ensuring higher utility than previous approaches.

1 Introduction

Data privacy is a fundamental problem in today's information age. Large amounts of data are collected from people by government agencies, hospitals, social networking systems, and other organizations, and are stored in databases. There are huge social benefits in analyzing this data, and in many situations, these organizations would like to release the data in some form for people to analyze.

* Gehrke's work on this material was supported by the NSF under Grant IIS-1012593. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

** Hay's work was supported by the Computing Innovation Fellows Project (http://cifellows.org/), funded by the Computing Research Association/Computing Community Consortium through NSF Grant 1019343.

*** Pass is supported in part by a Alfred P. Sloan Fellowship, Microsoft New Faculty Fellowship, NSF CAREER Award CCF-0746990, AFOSR YIP Award FA9550-10-1-0093, and DARPA and AFRL under contract FA8750-11-2-0211. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US government.

However, it is important to protect the privacy of the people that contributed their data; organizations need to make sure that sensitive information about individuals is not leaked to the people analyzing the data.

Many privacy definitions and schemes for releasing data have been proposed in the past (see [1] and [2] for surveys). However, many of them have been shown to be insufficient due to realistic attacks on such schemes (e.g., see [3]). The notion of *differential privacy* [4, 5], however, has remained strong and resilient to these attacks. Differential privacy requires that when one person's data is added or removed from the database, the output distribution of the database access mechanism changes very little (by at most an ϵ amount, where a specific notion of closeness of distributions is used). Differential privacy has quickly become the standard definition of privacy, and mechanisms for releasing a variety of functions (including histogram queries, principal component analysis, learning, and many more; see [6, 7] for a survey) have been developed.

One way to interpret the notion of differential privacy is that an attacker does not learn more about an individual i than what can be deduced from the data of everyone else in the database (see the appendix of [4]). In the context of e.g., social networks, where the data of an individual may be strongly correlated with the data of his/her friends, such a notion may not always provide sufficient privacy guarantees. To address this issue, an even stronger privacy definition, *zero-knowledge privacy*, was introduced in [8]. Roughly speaking, zero-knowledge privacy requires that whatever an adversary learns about an individual i can be "simulated" given just some "aggregate" information about the *remaining* individuals in the database; for instance, this aggregate information could be k random samples of the remaining individuals in the database. If the aggregate information contains all individuals (excluding i), zero-knowledge privacy collapses down to differential privacy, but for more restrictive classes of aggregate information (such as k random samples, where k is smaller than the number of individual in the database) zero-knowledge privacy is strictly stronger, and provides stronger privacy guarantees in contexts where there is correlation between individuals.

Privacy from Random Sampling of Data Both differential privacy and zero-knowledge privacy provide strong privacy guarantees. However, for certain tasks, mechanisms satisfying these privacy definitions have to add a lot of "noise", thus lowering the utility of the released data. Also, many of these mechanisms run in exponential time (e.g., [9, 10]), so efficiency is also an issue. This leaves open the question of whether there exists a practical approach to sanitizing data, without harming utility too much.

One approach for circumventing the above-mentioned issues is to rely on the fact that in many cases of interest, the data to be sanitized has been collected via *random sampling* from some underlying population. Intuitively, this initial random sampling already provides some basic privacy guarantees, and may thus help us in decreasing the amount of noise added during sanitization. Indeed, there are several results in the literature indicating that random sampling helps in providing privacy: In [11] the authors quantify the level of the privacy that

may be obtained from just random sampling of data (without any further sanitization); in [12] the authors consider a certain type of "sample-and-aggregate" mechanism for achieving differential privacy (but the sampling technique here is more elaborate than just random sampling from a population); a result in [13] shows that random pre-sampling can be used to amplify the privacy level of a differentially private mechanism; finally, in a manuscript [14], the authors demonstrate that a random pre-sampling step applied to a particular mechanism leads to a differentially private mechanism.

In this paper, we continue the investigation of using random sampling as a means to achieve privacy. In particular, our goal is to provide a *general* definition of privacy that allows us to achieve both differential and zero-knowledge privacy in situations where the data is collected using random sampling from some population. In order to be realistic, we allow the random sampling during data collection to be *biased*, and an adversary may even know whether certain individuals were sampled or not. (Although the mechanisms in the earlier papers rely on random sampling, the random sampling is usually thought of as being part of the sanitization procedure and thus the mechanisms are only analyzed under the assumption that the sampling has been done "ideally".) Additionally, we will require that the privacy notion is meaningful in its own right, also without any pre-sampling; we believe this requirement is crucial for guaranteeing a strong fall-back guarantee even in case the result of the pre-sampling is leaked (and thus the attacker knows exactly who was sampled).

1.1 Towards a Weaker Notion of Privacy

We aim to develop a new privacy definition that allows us to design mechanisms that have greater utility or efficiency than differentially private mechanisms, but still provide a meaningful notion of privacy; furthermore, we want mechanisms satisfying the new definition to achieve differential and zero-knowledge privacy when the underlying data was collected via biased random sampling from some population. To this end, we begin by reconsidering some older notions of privacy.

k-Anonymity and Blending in a Crowd k-anonymity [15] is a privacy definition specifically for releasing data tables, where a data table is simply a table of records (rows), each of which has values for the attributes (columns) of the table. Roughly speaking, a released data table satisfies *k-anonymity* if every record in the table is the same as $k-1$ other records in the table with respect to certain "identifying" attributes (chosen beforehand). k-anonymity imposes constraints on the syntax of the released data table, but does not consider the way the released data table was computed from the underlying database; this issue has led to several practical attacks against the notion of k-anonymity (e.g., see [16, 17]). k-anonymity can be viewed as being based on the intuition of *"blending in a crowd"*, since the records in the released output are required to "blend" with other records. Intuitively, in many cases, if an individual blends in a crowd of many people in the database, then the individual's privacy is sufficiently protected. However, as demonstrated by known attacks, k-anonymity does not

properly capture this intuition as it does not impose any restrictions on the algorithm/mechanism used to generate the released output. Indeed, one of the key insights behind the notion of differential privacy was that privacy should be a property of the sanitization mechanism and not just the output of it.

Relying on this insight, we aim to develop a privacy notion that captures what it means for a mechanism to guarantee that individuals "blend in a crowd". (Another definition partly based on the intuition of blending in a crowd is (c, t)-*isolation* [18], which requires adversaries to be unable to isolate an individual, represented by a data point in $\mathbb{R}^d$, by roughly determining the individual's location in $\mathbb{R}^d$; we formalize the intuition of blending in a crowd in a very different way.)

Crowd-Blending Privacy – A New Privacy Definition Let us now turn to describing our new privacy definition, which we call *crowd-blending privacy*. We say that an individual *blends* with another individual with respect to a mechanism *San* if the two individuals are *indistinguishable by the mechanism San*, i.e., whenever we have a database containing either one or both of the individuals, we can replace one of the individual's data with the other individual's data, and the mechanism's output distribution remains essentially the same. We say that an individual t *blends in a crowd of* k *people in the database* D *with respect to the mechanism San* if there exist at least $k-1$ other individuals in the database D that blend with individual t with respect to *San*. The intuition behind this notion is that if an individual t blends in a crowd of k people in the database, then the mechanism essentially does not release any information about individual t beyond the general characteristics of the crowd of k people; in particular, the mechanism does not release any personal information that is specific to individual t and no one else.

Roughly speaking, we say that a mechanism *San* is *crowd-blending private* if the following property holds: For every database and every individual in the database, either the individual *blends in a crowd of* k *people in the database with respect to San*, or the mechanism *San essentially ignores the individual's data.*

We do not claim that crowd-blending privacy provides sufficiently strong privacy protection in *all* scenarios: the key weakening with respect to differential privacy is that an attacker who knows the data of everyone in an individual i's crowd (except i) may learn information about individual i, as long as this information is "general" in the sense that it applies to the entire crowd. For instance, if the attacker knows everyone in the crowd of individual i, it may deduce that i has, say, three children, as long as everyone in i's crowd has three children. Although to some extent, this may be viewed as a privacy violation (that would not be allowed by the notion of differential privacy), we would argue that the attribute leaked about individual i is "non-sensitive" as it is shared by a sufficiently large crowd. Thus, we view this weakening as desirable in many contexts as it allows us to trade privacy of "non-sensitive information" for improved utility.

A potentially more serious deficiency of the definition is that (in contrast to differential and zero-knowledge privacy) crowd-blending privacy is not closed

under composition: San_1 and San_2 may both be crowd-blending private, but the crowds for an individual with respect to San_1 and San_2 could be essentially disjoint, making the individual's crowd for the combination of San_1 and San_2 very small. Although we view composition as an important property of a privacy definition, our goal here is to study the weakest possible "meaningful" definition of "stand-alone" privacy that when combined with pre-sampling leads to strong privacy notions (such as differential and zero-knowledge privacy) that themselves are closed under composition.

1.2 New Database Mechanisms

As it turns out, achieving crowd-blending privacy is significantly easier than achieving differential privacy, and crowd-blending private mechanisms may yield significantly higher utility than differentially private ones.

Privately Releasing Histograms with No Noise for Sufficiently Large Counts We show that we can release histograms with crowd-blending privacy where no noise is added to bins with a sufficiently large count (and only a small amount of noise is added to bins with a small count). Intuitively, individuals in the same bin blend with each other; thus, the individuals that belong to a bin with a sufficiently large count already blend in a crowd, so no noise needs to be added to the bin. It is easy to see that it is impossible to release the exact count of a bin in a histogram while satisfying differential privacy or zero-knowledge privacy. Using crowd-blending privacy, we can overcome this limitation (for bins with a sufficiently large count) and achieve better utility. These results can be found in Section 3.1.

Privately Releasing Synthetic Data Points in $\mathbb{R}^d$ for Computing Smooth Functions Given a class $\mathcal{C}$ of counting queries whose size is not too large, it is shown in [10] how to release a synthetic database for approximating all the queries in $\mathcal{C}$ simultaneously while satisfying differential privacy; however, the mechanism is not necessarily efficient. It is known that it is impossible (assuming the existence of one-way functions) to *efficiently* and privately release a synthetic database for approximating certain classes of counting queries, such as the class of all 2-way marginals (see [19, 20]). However, these query functions are non-smooth in the sense that even slightly changing one row of the input database can affect the output of the query functions quite a lot. Here, we focus on efficiently and privately releasing synthetic data for approximating *all* "smooth" functions $g : (\mathbb{R}^d)^* \to \mathbb{R}^m$.

Roughly speaking, a function $g : (\mathbb{R}^d)^* \to \mathbb{R}^m$ is *smooth* if the value of g does not change much when we perturb the data points of the input slightly. We show that we can *efficiently* release synthetic data points in $\mathbb{R}^d$ for approximating *all* smooth functions simultaneously while satisfying crowd-blending privacy. On the other hand, we show that there are smooth functions that cannot even be approximated with non-trivial utility from any synthetic data that has been released with differential privacy (even if the differentially private mechanism is *inefficient*). These results can be found in the full version of this paper.

1.3 From Crowd-Blending Privacy to Zero-Knowledge Privacy

Our main technical result shows that if we combine a crowd-blending private mechanism with a natural pre-sampling step, then the combined algorithm satisfies zero-knowledge privacy (and thus differential privacy as well). We envision the pre-sampling step as being part of the data collection process, where individuals in some population are sampled and asked for their data. Thus, if data is collected using random sampling of individuals from some population, and next sanitized using a crowd-blending private mechanism, then the resulting process ensures zero-knowledge privacy.

We first prove our main theorem for the case where the pre-sampling step samples each individual in the population with probability p independently. In reality, the sampling performed during data collection may be slightly biased or done slightly incorrectly, and an adversary may know whether certain individuals were sampled or not. Thus, we next extend our main theorem to also handle the case where the sampling probability is not necessarily the same for everybody, but the sampling is still "robust" in the sense that most individuals are sampled independently with probability in between p and p' (this probability can even depend on the individual's data), where p and p' are relatively close to one another, while the remaining individuals are sampled independently with arbitrary probability. As a result, we have that in scenarios where data has been collected using any robust sampling, we may release data which both ensures strong utility guarantees and satisfies very strong notions of privacy (i.e., zero-knowledge privacy and differential privacy). In particular, this methodology can allow us to achieve zero-knowledge privacy and differential privacy while guaranteeing utility that is better than that of previous methods (such as for releasing histograms or synthetic data points as described above). Our main theorems can be found in Section 4.

It is worthwhile to note that the particular mechanism considered in [14] (which in fact is a particular mechanism for achieving k-anonymity) can easily be shown to satisfy crowd-blending privacy; as a result, their main result can be derived (and significantly strengthened) as a corollary of our main theorem.[1] (See Section 3.1 and 4 for more details.)

2 Preliminaries and Existing Privacy Definitions

A *database* is a finite *multiset* of data values, where a data value is simply an element of some fixed set X, which we refer to as the *data universe*. Each data value in a database belongs to an individual, so we also refer to a data value in a database as an *individual* in the database. For convenience, we will sometimes

[1] As mentioned, none of the earlier work using random pre-sampling focus on the case when the sampling is biased; furthermore, even for the case of perfect random sampling, the authors of [14] were not able to provide a closed form expression of the level of differential privacy achieved by their mechanism, whereas a closed form expression can be directly obtained by applying our main theorem.

order the individuals in a database in an arbitrary way and think of the database as an element of X^*, i.e., a vector with components in X (the components are referred to as the *rows* of the database). Given a database D and a data value $v \in X$, let (D, v) denote the database $D \cup \{v\}$. A (database) *mechanism* is simply an algorithm that operates on databases.

Given $\epsilon, \delta \geq 0$ and two distributions Z and Z' over $\{0,1\}^*$, we shall write $Z \approx_{\epsilon,\delta} Z'$ to mean that for every $Y \subseteq \{0,1\}^*$ we have

$$\Pr[Z \in Y] \leq e^{\epsilon} \Pr[Z' \in Y] + \delta$$

and

$$\Pr[Z' \in Y] \leq e^{\epsilon} \Pr[Z \in Y] + \delta.$$

We shall also write $Z \approx_{\epsilon} Z'$ to mean $Z \approx_{\epsilon,0} Z'$. Differential privacy (see [4, 5]) can now be defined in the following manner:

Definition 1 ([4, 5]). *A mechanism San is said to be* **ϵ-differentially private** *if for every pair of databases D and D' differing in only one data value, we have $San(D) \approx_{\epsilon} San(D')$.*

There are two definitions in the literature for "a pair of databases D and D' differing in only one data value", leading to two slightly different definitions of differential privacy. In one definition, it is required that D contains D' and has exactly one more data value than D'. In the other definition, it is required that $|D| = |D'|$, $|D \setminus D'| = 1$, and $|D' \setminus D| = 1$. Intuitively, differential privacy protects the privacy of an individual t by requiring the output distribution of the mechanism to be essentially the same regardless of whether individual t's data is included in the database or not (or regardless of what data value individual t has).

We now begin describing zero-knowledge privacy, which is a privacy definition introduced in [8] that is strictly stronger than differential privacy. In the definition of zero-knowledge privacy, *adversaries* and *simulators* are simply randomized algorithms that play certain roles in the definition. Let San be any mechanism. For any database D, any adversary A, and any auxiliary information $z \in \{0,1\}^*$, let $Out_A(A(z) \leftrightarrow San(D))$ denote the output of A on input z after interacting with the mechanism San operating on the database D. San can be interactive or non-interactive. If San is non-interactive, then $San(D)$ simply sends its output (e.g., sanitized data) to A and then halts immediately.

Let agg be any class of randomized algorithms. agg is normally a class of randomized aggregation functions that provide aggregate information to simulators, as described in the introduction.

Definition 2 ([8]). *A mechanism San is said to be* **(ϵ, δ)-zero-knowledge private with respect to** *agg if there exists a $T \in agg$ such that for every adversary A, there exists a simulator S such that for every database D, every individual $t \in D$, and every auxiliary information $z \in \{0,1\}^*$, we have*

$$Out_A(A(z) \leftrightarrow San(D)) \approx_{\epsilon,\delta} S(z, T(D \setminus \{t\}), |D|).$$

Intuitively, zero-knowledge privacy requires that whatever an adversary can compute about individual t by accessing (i.e., interacting with) the mechanism can also be essentially computed without accessing the mechanism but with certain aggregate information about the remaining individuals; this aggregate information is provided by an algorithm in agg. The adversary in the latter scenario is represented by the simulator S. This ensures that the adversary essentially does not learn any additional information about individual t beyond the aggregate information provided by an algorithm in agg on the remaining individuals.

agg is normally some class of randomized aggregation functions, such as the class of all functions T that draws r random samples from the input database and performs any computation (e.g., computes the average or simply outputs the samples) on the r random samples (note that in the definition, T is applied to $D \setminus \{t\}$ instead of D so that the aggregate information from T does not depend directly on individual t's data). Zero-knowledge privacy with respect to this class of aggregation functions ensures that an adversary essentially does not learn anything more about an individual beyond some "r random sample aggregate information" of the other individuals. One can also consider zero-knowledge privacy with respect to other classes of aggregation functions, such as the class of (randomized) functions that first sample each row of the input database with probability p (or in between p and p') independently and then performs any computation on the samples. We will actually use such classes of aggregation functions when we prove our main theorems later. It can be easily shown that zero-knowledge privacy (with respect to any class agg) implies differential privacy (see [8]).

3 Crowd-Blending Privacy – A New Privacy Definition

We now begin to formally define our new privacy definition. Given $t, t' \in X$, $\epsilon \geq 0$, and a mechanism San, we say that t and t' are *ϵ-indistinguishable* by San, denoted $t \approx_{\epsilon,San} t'$, if $San(D, t) \approx_\epsilon San(D, t')$ for every database D. Intuitively, t and t' are indistinguishable by San if for any database containing t, we can replace the t by t' and the output distribution of San remains essentially the same. Usually, t and t' are the data values of two individuals, and if t and t' are indistinguishable by San, then this roughly means that San cannot distinguish these two individuals regardless of who else is in the database. If t and t' are ϵ-indistinguishable by San, we also loosely say that t *blends* with t' (with respect to San). We now describe what it means for an individual to blend in a crowd of people in the database (with respect to a mechanism).

Definition 3. *Let D be any database. An individual $t \in D$* **ϵ-blends in a crowd of k people in D with respect to the mechanism** *San if $|\{t' \in D : t' \approx_{\epsilon,San} t\}| \geq k$.*

In the above definition, $\{t' \in D : t' \approx_{\epsilon,San} t\}$ should be regarded as a multiset. When the mechanism San is clear from context, we shall simply omit

the "with respect to the mechanism San". Intuitively, an individual $t \in D$ blends in a crowd of k people in D if t is indistinguishable by San from at least $k-1$ other individuals in D. Note that by the definition of two individuals being indistinguishable by San, $t \in D$ must be indistinguishable by San from each of these $k-1$ other individuals *regardless of what the database is*, as opposed to only when the database is D. (A weaker requirement would be that for each of these $k-1$ other individuals t', t and t' only need to be "indistinguishable by San with respect to D", i.e., if we take D and replace t by t' or vice versa, the output distributions of San on D and the modified D are essentially the same; we leave investigating this and other possible weaker requirements for future work.) We are now ready to state our new privacy definition.

Definition 4 (Crowd-blending privacy). *A mechanism San is (k, ϵ)-**crowd-blending private** if for every database D and every individual $t \in D$, either t ϵ-blends in a crowd of k people in D, or $San(D) \approx_\epsilon San(D \setminus \{t\})$ (or both).*

Crowd-blending privacy requires that for every individual t in the database, either t blends in a crowd of k people in the database, or the mechanism essentially ignores individual t's data (the latter case is captured by $San(D) \approx_\epsilon San(D \setminus \{t\})$ in the definition). When an individual t blends in a crowd of k people in the database, the mechanism essentially does not release any information about individual t beyond the general characteristics of the crowd of k people. This is because the mechanism cannot distinguish individual t from the people in the crowd of k people, i.e., individual t's data can be changed to the data of another person in the crowd of k people and the output distribution of the mechanism remains essentially the same. A consequence is that the mechanism does not release any personally identifying information about individual t.

As mentioned in the introduction, crowd-blending privacy is not closed under composition (see the full version of this paper for an example); however, we note that the privacy guarantee of blending in a crowd of k people in the database (described above) holds regardless of the amount of auxiliary information the adversary has (i.e., the definition is agnostic to the adversary's auxiliary information). Additionally, as mentioned previously, we show in Section 4 that when crowd-blending privacy is combined with "robust pre-sampling", we get zero-knowledge privacy and thus differential privacy as well, both of which satisfy composition in a natural way. Thus, as long as robust sampling is used during data collection before running a crowd-blending private mechanism on the collected data, independent releases from crowd-blending private mechanisms do compose and satisfy zero-knowledge privacy and differential privacy. (We also mention that one can compose a crowd-blending private mechanism with a differentially private mechanism to obtain a crowd-blending private mechanism; see the full version of this paper for details.)

Relationship with Differential Privacy Differential privacy implies crowd-blending privacy.

Proposition 1 (Differential privacy $\Longrightarrow$ Crowd-blending privacy). *Let San be any ϵ-differentially private mechanism. Then, San is (k, ϵ)-crowd-blending private for every integer $k \geq 1$.*

Proof. This immediately follows from the two privacy definitions.

(k, ϵ)-crowd-blending privacy for some integer k does not imply differential privacy in general; this will be clear from the examples of crowd-blending private mechanisms that we give later. Crowd-blending privacy requires that for every database D and every individual $t \in D$, at least one of two conditions hold. The second condition $San(D) \approx_\epsilon San(D \setminus \{t\})$ is similar to the condition required in differential privacy. Thus, we can view crowd-blending privacy as a relaxation of differential privacy. If we remove the first condition "t ϵ-blends in a crowd of k people in D" from crowd-blending privacy, we clearly get the same definition as differential privacy. If we remove the second condition instead, it turns out that we also get differential privacy. (When we remove the second condition $San(D) \approx_\epsilon San(D \setminus \{t\})$, we also change the definition to only consider databases of size at least k, since otherwise it would be impossible for individual t to blend in a crowd of k people in the database.)

Proposition 2 (Removing the condition $San(D) \approx_\epsilon San(D \setminus \{t\})$ in crowd-blending privacy results in differential privacy). *Let San be any mechanism, let $\epsilon \geq 0$, and let k be any integer ≥ 2. Then, San is ϵ-differentially private[2] if and only if San satisfies the property that for every database D of size at least k and every individual $t \in D$, t ϵ-blends in a crowd of k people in D with respect to San.*

See the full version of this paper for the proof of Proposition 2.

3.1 Examples of Crowd-Blending Private Mechanisms

Given a partition P of the data universe X, and given a database D, one can compute the histogram with respect to the partition P using the database D; the histogram specifies for each block of the partition (which we refer to as a "bin") the number of individuals in D that belong to the block (which we refer to as the "count" of the bin). We first give an example of a crowd-blending private mechanism that computes a histogram and suppresses (i.e., sets to 0) bin counts that are considered too small.

Example 1 (Histogram with suppression of small counts). Let P be any partition of X. Fix $k \in \mathbb{Z}_{\geq 0}$. Let *San* be a mechanism that, on input a database D, computes the histogram with respect to the partition P using the database D, suppresses each bin count that is $< k$ (by setting the count to 0), and then releases the resulting histogram.

[2] Here, we are using the version of differential privacy that considers a pair of databases of equal size.

Then, San is $(k, 0)$-crowd-blending private. To see this, we note that an individual t in a database D is 0-indistinguishable by San from all the individuals in D that belong to the same bin as t. If there are at least k such people, then individual t blends with k people in D; otherwise, we have $San(D) \approx_0 San(D \setminus \{t\})$ since San suppresses each bin count that is $< k$.

It is easy to see that it is impossible to release the exact count of a bin while satisfying differential privacy. Thus, crowd-blending privacy is indeed weaker than differential privacy. For crowd-blending privacy, we can actually get better utility by adding a bit of noise to bins with low counts instead of completely suppressing them.

Example 2 (Histogram with noise for small counts and no noise for large counts). Let P be any partition of X. Fix $\epsilon > 0$ and $k \in \mathbb{Z}_{\geq 0}$. Let San be a mechanism that, on input a database D, computes the histogram with respect to the partition P using the database D. Then, San replaces each bin count $i < k$ with $A(i)$, where A is any (randomized) algorithm that satisfies $A(j) \approx_\epsilon A(j-1)$ for every $0 < j < k$ ($A(i)$ is normally a noisy version of i). San then releases the noisy histogram.

Then, San is (k, ϵ)-crowd-blending private. To see this, we note that an individual t in a database D is ϵ-indistinguishable (in fact, 0-indistinguishable) by San from all the individuals in D that belong to the same bin as t. If there are at least k such people, then individual t blends with k people in D, as required. If not, then we have $San(D) \approx_\epsilon San(D \setminus \{t\})$, since the histogram when using the database D is the same as the histogram when using the database $D \setminus \{t\}$ except for individual t's bin, which differs by one; however, San replaces the count i for individual t's bin with $A(i)$, and the algorithm A satisfies $A(i) \approx_\epsilon A(i-1)$, so $San(D) \approx_\epsilon San(D \setminus \{t\})$, as required.

We can choose the algorithm A to be $A(j) = j + Lap(\frac{1}{\epsilon})$, where $Lap(\lambda)$ is (a random variable with) the Laplace distribution with probability density function $f_\lambda(x) = \frac{1}{2\lambda} e^{|x|/\lambda}$. The proof that $A(j) \approx_\epsilon A(j-1)$ for every $0 < j < k$ is simple and can be implicitly found in [4].

The differentially private mechanism in [4] for computing histograms has to add noise to every bin, while our mechanism here only adds noise to the bins that have a count that is $< k$.

Example 3 (Sanitizing a database by generalizing records safely). Many mechanisms for achieving k-anonymity involve "generalizing" the records in the input table by replacing specific values with more general values, such as replacing a specific age with an age range. If this is not done carefully, the privacy of individuals can be breached, as shown by many attacks in the past (e.g., see [16, 17]). Most of these mechanisms do not satisfy crowd-blending privacy. However, if the generalization of records is done carefully, achieving crowd-blending privacy may be possible.

One example is the mechanism of [14]: Let Y be any set, and let $f : X \to Y$ be any function. We think of Y as a set of possible "generalized records", and f is

a function that maps a record to its generalized version. Let San be a mechanism that, on input a database D, applies the function f to each individual in D; let $f(D)$ be the multi-set of images in Y. San then removes each record in $f(D)$ that appears fewer than k times in $f(D)$, and then outputs the result. It is easy to see that San is $(k, 0)$-crowd-blending private. To see this, we note that an individual t in a database D is 0-indistinguishable by San from all the individuals in D that also get mapped to $f(t)$. If there are at least k such people, then individual t blends with k people in D; otherwise, we have $San(D) \approx_0 San(D \setminus \{t\})$ since San removes each record in $f(D)$ that appears fewer than k times in $f(D)$.

Example 4 (Privately Releasing Synthetic Data Points in $\mathbb{R}^d$ for Computing Smooth Functions). Roughly speaking, a function $g : (\mathbb{R}^d)^* \to \mathbb{R}^m$ is *smooth* if the value of g does not change much when we perturb the data points of the input slightly. In the full version of this paper, we show that we can *efficiently* release synthetic data points in $\mathbb{R}^d$ for approximating *all* smooth functions simultaneously while satisfying crowd-blending privacy. On the other hand, we show that there are smooth functions that cannot even be approximated with non-trivial utility from any synthetic data that has been released with differential privacy (even if the differentially private mechanism is *inefficient*). See the full version of this paper for details.

4 Our Main Theorem

In this section, we prove our main theorem that says that when we combine a crowd-blending private mechanism with a natural *pre-sampling* step, the combined algorithm is zero-knowledge private (and thus differentially private as well). The pre-sampling step should be thought of as being part of the data collection process, where individuals in some population are sampled and asked for their data. A crowd-blending private mechanism is then run on the samples to release useful information while preserving privacy.

We first prove our main theorem for the case where the pre-sampling step samples each individual in the population with probability p independently. In reality, the sampling performed during data collection may be slightly *biased* or done slightly incorrectly, and an adversary may know whether certain individuals were sampled or not. Thus, we later extend our main theorem to the case where the sampling probability is not necessarily the same for everybody, but the sampling is still *robust* in the sense that most individuals are sampled independently with probability in between p and p' (this probability can even depend on the individual's data), where p and p' are relatively close to one another, while the remaining individuals are sampled independently with arbitrary probability.

We begin with some necessary terminology and notation. A *population* is a collection of *individuals*, where an individual is simply represented by a data value in the data universe X. Thus, a population is actually a multiset of data values, which is the same as a database. (If we want individuals to have unique data values, we can easily modify X to include personal/unique identifiers.)

Given a population $\mathcal{P}$ and a real number $p \in [0,1]$, let $Sam(\mathcal{P}, p)$ be the outcome of sampling each individual in $\mathcal{P}$ with probability p independently.

Although zero-knowledge privacy was originally defined for mechanisms operating on *databases*, one can also consider mechanisms operating on *populations*, since there is essentially no difference between the way we model populations and databases. (In the definition of zero-knowledge privacy, we simply change "database" to "population" and D to $\mathcal{P}$.) We now describe a class of (randomized) aggregation functions that we will use in the definition of zero-knowledge privacy.

- $iidRS(p)$ = i.i.d. random sampling with probability p : the class of algorithms T such that on input a population $\mathcal{P}$, T chooses each individual in $\mathcal{P}$ with probability p independently, and then performs any computation on the data of the chosen individuals.[3]

We now state and prove the basic version of our main theorem.

Theorem 1 (Sampling + Crowd-Blending Privacy ⇒ Zero-Knowledge Privacy). *Let San be any (k,ϵ)-crowd-blending private mechanism with $k \geq 2$, and let $p \in (0,1)$. Then, the algorithm San_{zk} defined by $San_{zk}(\mathcal{P}) = San(Sam(\mathcal{P}, p))$ for any population $\mathcal{P}$ is $(\epsilon_{zk}, \delta_{zk})$-zero-knowledge private[4] with respect to $iidRS(p)$, where*

$$\epsilon_{zk} = \ln\left(p \cdot \left(\frac{2-p}{1-p}e^{\epsilon}\right) + (1-p)\right) \quad \text{and} \quad \delta_{zk} = e^{-\Omega(k\cdot(1-p)^2)}.$$

The proof of Theorem 1 can be found in the full version of this paper, but the main ideas of the proof will be explained here. To prove Theorem 1, we will first prove two supporting lemmas. The first lemma essentially says that if an individual t blends with (i.e., is indistinguishable by San from) many people in the population, then t's privacy is protected when we sample from the population and run San on the samples:

Lemma 1 (Protection of individuals that blend with many people in the population). *Let San be any mechanism, $\mathcal{P}$ be any population, $p \in (0,1)$, and $\epsilon \geq 0$. Let t be any individual in $\mathcal{P}$, and let A be any non-empty subset of $\mathcal{P} \setminus \{t\}$ such that $t' \approx_{\epsilon, San} t$ for every individual $t' \in A$. Let $n = |A|$. Then, we have*

$$San(Sam(\mathcal{P}, p)) \approx_{\epsilon_{final}, \delta_{final}} San(Sam(\mathcal{P} \setminus \{t\}, p)),$$

where $\epsilon_{final} = \ln(p \cdot (\frac{2-p}{1-p}e^{\epsilon}) + (1-p))$ and $\delta_{final} = e^{-\Omega((n+1)p(1-p)^2)}$.

[3] To make zero-knowledge privacy compose naturally for this type of aggregate information, we can extend $iidRS(p)$ to $iidRS(p, r)$, where T is now allowed to perform r rounds of sampling before performing any computation on the sampled data. It is not hard to see that zero-knowledge privacy with respect to $iidRS(p, r)$ composes in a natural way.

[4] The constant hidden by the $\Omega(\cdot)$ in δ_{zk} can be easily computed; however, we did not try to optimize the constant in any way.

In the lemma, A is any non-empty set of individuals in $\mathcal{P} \setminus \{t\}$ that blend with individual t. (We could set A to be the set of *all* individuals in $\mathcal{P} \setminus \{t\}$ that blend with individual t, but leaving A more general allows us to more easily extend the lemma to the case of "robust" sampling later.) We note that δ_{final} is smaller when $n = |A|$ is larger, i.e., when t blends with more people. Intuitively, if an individual t is indistinguishable by San from many other people in the population, then t's presence or absence in the population does not affect the output of $San(Sam(\cdot, p))$ much, since the people indistinguishable from t can essentially take the place of t in almost any situation (and the output of San would essentially be the same). Since it does not matter much whether individual t is in the population or not, it follows that t's privacy is protected.

The proof of the lemma *roughly* works as follows: Consider two scenarios, one where individual t is in the population (i.e., $San(Sam(\mathcal{P}, p))$ in the lemma), and one where individual t has been removed from the population (i.e., $San(Sam(\mathcal{P} \setminus \{t\}, p))$ in the lemma). Our goal is to show that the output of San is essentially the same in the two scenarios, i.e., $San(Sam(\mathcal{P}, p)) \approx_{\epsilon_{final}, \delta_{final}} San(Sam(\mathcal{P} \setminus \{t\}, p))$. Conditional on individual t not being sampled in the first scenario, the two scenarios are exactly the same, as desired. Thus, we now always condition on individual t being sampled in the first scenario. In the lemma, A is a set of individuals in the population (excluding t) that are indistinguishable from t by San. Let $\widetilde{m}$ denote the number of people in A that are sampled. The proof involves showing the following two properties:

1. $\widetilde{m}$ is relatively smooth near its expectation: For every integer m near the expectation of $\widetilde{m}$, $\Pr[\widetilde{m} = m]$ is relatively close to $\Pr[\widetilde{m} = m + 1]$.
2. For every integer $m \in \{0, \ldots, n-1\}$, the output of San in the first scenario conditioned on $\widetilde{m} = m$ (and t being sampled) is essentially the same as the output of San in the second scenario conditioned on $\widetilde{m} = m + 1$.

For the first property, we note that $\widetilde{m}$ follows a binomial distribution, which can be shown to be relatively smooth near its expectation. To show the second property, we note that when we condition on $\widetilde{m} = m$ (and t being sampled) in the first scenario, m random samples are drawn uniformly from A (one at a time) without replacement, and also $t \notin A$ is sampled for sure (and the remaining individuals are sampled independently with probability p). This is very similar to the second scenario conditioned on $\widetilde{m} = m+1$, where $m+1$ random samples are drawn uniformly from A without replacement, since if we replace the $(m+1)^{th}$ sample by t, we get back the first scenario conditioned on $\widetilde{m} = m$ (and t being sampled). Since the $(m+1)^{th}$ sample is indistinguishable from t by San, the output of San is essentially the same in both scenarios.

Using the two properties above, one can show that when $\widetilde{m}$ is close to its expectation, the output of San is essentially the same in both scenarios. δ_{final} in the lemma captures the probability of the bad event where $\widetilde{m}$ is not close to its expectation, which we bound by essentially using a Chernoff bound. See the full version of this paper for the proof of Lemma 1.

We now show how pre-sampling combined with a crowd-blending private mechanism can protect the privacy of individuals who blend with (i.e., are indistinguishable by San from) few people in the population.

Lemma 2 (Protection of individuals that blend with few people in the population). *Let San be any (k, ϵ)-crowd-blending private mechanism with $k \geq 2$, let $\mathcal{P}$ be any population, and let $p \in (0, 1)$. Let t be any individual in $\mathcal{P}$, and let $n = |\{t' \in \mathcal{P} \setminus \{t\} : t' \approx_{\epsilon, San} t\}|$. Then, if $n \leq \frac{k-1}{p(2-p)}$, we have*

$$San(Sam(\mathcal{P}, p)) \approx_{\epsilon_{final}, \delta_{final}} San(Sam(\mathcal{P} \setminus \{t\}, p)),$$

where $\epsilon_{final} = \ln(pe^{\epsilon} + (1 - p))$ and $\delta_{final} = pe^{-\Omega(k \cdot (1-p)^2)}$.

The proof of the lemma *roughly* works as follows: In the lemma, n is the number of people in the population that individual t blends with, and is assumed to be small. We will show that when we remove individual t from the population, the output of San does not change much.

Consider two scenarios, one where individual t is in the population, and one where individual t has been removed from the population. Conditional on individual t not being sampled in the first scenario, the two scenarios are exactly the same, as desired. Thus, we now always condition on individual t being sampled in the first scenario. Since individual t blends with few people in the population, we have that with very high probability, the database obtained from sampling from the population would contain fewer than k people that blend with individual t; since San is (k, ϵ)-crowd-blending private and individual t does not blend in a crowd of k people in the database, San must essentially ignore individual t's data; thus, the first scenario is essentially the same as the second scenario, since individual t's data is essentially ignored anyway. δ_{final} in the lemma captures the probability of the bad event where the database obtained from sampling actually contains k people that blend with individual t. See the full version of this paper for the proof of Lemma 2.

We are now ready to prove Theorem 1. The proof of the theorem roughly works as follows: By definition of $iidRS(p)$, a simulator in the definition of zero-knowledge privacy is able to obtain the aggregate information $Sam(\mathcal{P} \setminus \{t\}, p)$. With $Sam(\mathcal{P} \setminus \{t\}, p)$, the simulator can easily compute $San(Sam(\mathcal{P} \setminus \{t\}, p))$, which it can then use to simulate the computation of the given adversary. It is not hard to see that the simulation works if $San(Sam(\mathcal{P}, p)) \approx_{\epsilon_{zk}, \delta_{zk}} San(Sam(\mathcal{P} \setminus \{t\}, p))$ holds. Thus, consider any population $\mathcal{P}$ and any individual $t \in \mathcal{P}$. Recall that Lemma 1 protects the privacy of individuals that blend with many people in $\mathcal{P}$, while Lemma 2 protects the privacy of individuals that blend with few people in $\mathcal{P}$. Thus, if individual t blends with many people in $\mathcal{P}$, we use Lemma 1; otherwise, we use Lemma 2. It then follows that $San(Sam(\mathcal{P}, p)) \approx_{\epsilon_{zk}, \delta_{zk}} San(Sam(\mathcal{P} \setminus \{t\}, p))$, as required. See the full version of this paper for the proof of Theorem 1.

4.1 Our Main Theorem Extended to Robust Sampling

We now extend our main theorem to the case where the sampling probability is not necessarily the same for everybody, but the sampling is still "robust" in the sense that most individuals are sampled independently with probability in between p and p' (this probability can even depend on the individual's data), where p and p' are relatively close to one another (i.e., $\frac{p'}{p}$ is not too large), while the remaining individuals are sampled independently with arbitrary probability.

We begin with some more notation. Given a population $\mathcal{P}$ and a function $\pi : X \to [0,1]$, let $Sam(\mathcal{P}, \pi)$ be the outcome of sampling each individual t in $\mathcal{P}$ with probability $\pi(t)$ independently. We note that for $Sam(\mathcal{P}, \pi)$, two individuals in $\mathcal{P}$ with the same data value in X will have the same probability of being sampled. However, we can easily modify the data universe X to include personal/unique identifiers so that we can represent an individual by a unique data value in X. Thus, for convenience, we now define a population to be a subset of the data universe X instead of being a multiset of data values in X. Then, each individual in a population would have a unique data value in X, so π does not have to assign the same sampling probability to two different individuals. We now describe a class of aggregation functions that we will use in the definition of zero-knowledge privacy.

- $iRS(p, p', \ell)$ = independent random sampling with probability in between p and p' except for ℓ individuals: the class of algorithms T such that on input a population $\mathcal{P}$, T independently chooses each individual $t \in \mathcal{P}$ with some probability $p_t \in [0,1]$ (possibly dependent on t's data), but all except for at most ℓ individuals in $\mathcal{P}$ must be chosen with probability in $\{0\} \cup [p, p']$; T then performs any computation on the chosen individuals' data.

We now state the extended version of our main theorem.

Theorem 2 (Robust Sampling + Crowd-Blending Privacy ⇒ Zero-Knowledge Privacy). *Let San be any (k, ϵ)-crowd-blending private mechanism with $k \geq 2$, let $0 < p \leq p' < 1$, let $\pi : X \to [0,1]$ be any function, let $\ell = |\{x \in X : \pi(x) \notin \{0\} \cup [p, p']\}|$, and let $p_{\max} = \sup_{x \in X} \pi(x)$. Suppose $\ell < k - 1$.*

Then, the algorithm San_{zk} defined by $San_{zk}(\mathcal{P}) = San(Sam(\mathcal{P}, \pi))$ for any population $\mathcal{P}$ is $(\epsilon_{zk}, \delta_{zk})$-zero-knowledge private with respect to $iRS(p, p', \ell)$, where

$$\epsilon_{zk} = \ln\left(p_{\max} \cdot \left(\frac{p'}{p}\frac{(1-p)(2-p)}{(1-p')^2}e^{\epsilon}\right) + (1 - p_{\max})\right) \text{ and}$$

$$\delta_{zk} = \max\left\{\frac{p_{\max}}{p}, \frac{p_{\max}}{1-p'}\right\} e^{-\Omega((k-\ell)\cdot(1-p')^2)}.$$

In the theorem, ℓ represents the number of individuals that are sampled with probability outside of $\{0\} \cup [p, p']$. We prove the theorem by extending Lemmas 1 and 2 to the case of "robust" sampling. We now describe *some* (but not all)

of the main changes to the lemmas and their proofs (see the full version of this paper for the proof of Theorem 2 and the extended lemmas).

Let us first consider Lemma 1, which protects the privacy of individuals that blend with many people in the population. Like before, consider two scenarios, one where individual t is in the population, and one where individual t has been removed. Let $\widetilde{m}$ denote the number of people in A that are sampled (recall that A is a set of individuals that blend with individual t). Recall that in the proof of Lemma 1, we had to show two properties: (1) $\widetilde{m}$ is relatively smooth near its expectation, and (2) the output of San in the first scenario conditioned on $\widetilde{m} = m$ (and t being sampled) is essentially the same as the output of San in the second scenario conditioned on $\widetilde{m} = m + 1$.

For the first property, we used the fact that the binomial distribution is relatively smooth near its expectation. Here, since the sampling is no longer i.i.d. but is still robust, we need the Poisson binomial distribution (the sum of independent Bernoulli trials, where the success probabilities are not necessarily the same) to be relatively smooth near its expectation. This can be shown as long as the success probabilities are all relatively close to one another; this is ensured by changing the lemma so that everyone in the set A is required to have a sampling probability in $[p, p']$.

For the second property, we used the fact that when we condition on $\widetilde{m} = m+1$ in the second scenario, we are drawing $m+1$ random samples from A (one at a time) uniformly without replacement, and if we replace the $(m+1)^{th}$ sample by t, we get the first scenario conditioned on $\widetilde{m} = m$ and t being sampled. This idea still works in the new setting where the sampling probabilities are no longer the same, since there is still a "draw-by-draw" selection procedure for drawing samples from A (one at a time) in a way so that right after drawing the j^{th} sample, the distribution of samples we currently have is the same as if we have conditioned on $\widetilde{m} = j$ (e.g., see Section 3 in [21]).

We now consider Lemma 2, which protects the privacy of individuals that blend with few people in the population. The extension of Lemma 2 to robust sampling redefines what is meant by "few people", since even if an individual blends with few people, many of them could be sampled with probability 1. With this modification, the proof of the extended lemma is similar to the proof of the original lemma.

When we prove the extended theorem using the extended lemmas, when we are trying to show that privacy holds for individual t, we look at how many people blend with t that are sampled with probability in $[p, p']$ (in particular, we exclude the ℓ people that are sampled with probability outside of $\{0\} \cup [p, p']$); similar to before, if this number is large, we use the extended version of Lemma 1; otherwise, we use the extended version of Lemma 2. See the full version of this paper for the proof of Theorem 2.

References

1. Chen, B.C., Kifer, D., LeFevre, K., Machanavajjhala, A.: Privacy-preserving data publishing. Foundations and Trends in Databases **2**(1-2) (2009) 1–167

2. Fung, B.C.M., Wang, K., Chen, R., Yu, P.S.: Privacy-preserving data publishing: A survey of recent developments. ACM Comput. Surv. **42**(4) (2010) 1–53
3. Kifer, D.: Attacks on privacy and definetti's theorem. In: SIGMOD Conference. (2009) 127–138
4. Dwork, C., Mcsherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Proc. of the 3rd Theory of Cryptography Conference. (2006) 265–284
5. Dwork, C.: Differential privacy. In: ICALP. (2006) 1–12
6. Dwork, C.: The differential privacy frontier. In: Proc. of the 6th Theory of Cryptography Conference (TCC). (2009)
7. Dwork, C.: Differential privacy: A survey of results. In: Theory and Applications of Models of Computation. Volume 4978 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2008) 1–19
8. Gehrke, J., Lui, E., Pass, R.: Towards privacy for social networks: a zero-knowledge based definition of privacy. In: Proceedings of the 8th conference on Theory of cryptography. TCC'11 (2011) 432–449
9. Dwork, C., Rothblum, G., Vadhan, S.: Boosting and differential privacy. In: Proc. of the 51st Annual IEEE Symposium on Foundations of Computer Science. (2010)
10. Blum, A., Ligett, K., Roth, A.: A learning theory approach to non-interactive database privacy. In: STOC '08: Proc. of the 40th annual ACM symposium on Theory of computing. (2008) 609–618
11. Chaudhuri, K., Mishra, N.: When random sampling preserves privacy. In: CRYPTO'06. (2006) 198–213
12. Nissim, K., Raskhodnikova, S., Smith, A.: Smooth sensitivity and sampling in private data analysis. In: STOC 2007. (2007) 75–84
13. Kasiviswanathan, S., Lee, H., Nissim, K., Raskhodnikova, S., Smith, A.: What can we learn privately? In: Foundations of Computer Science, 2008. (2008) 531–540
14. Li, N., Qardaji, W.H., Su, D.: Provably private data anonymization: Or, k-anonymity meets differential privacy. Manuscript (2011)
15. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertain. Fuzziness Knowl.-Based Syst. **10** (October 2002) 557–570
16. Wong, R.C.W., Fu, A.W.C., Wang, K., Pei, J.: Minimality attack in privacy preserving data publishing. In: Proceedings of the 33rd international conference on Very large data bases. VLDB '07, VLDB Endowment (2007) 543–554
17. Zhang, L., Jajodia, S., Brodsky, A.: Information disclosure under realistic assumptions: privacy versus optimality. In: Proceedings of the 14th ACM conference on Computer and communications security. CCS '07, ACM (2007) 573–583
18. Chawla, S., Dwork, C., McSherry, F., Smith, A., Wee, H.: Toward privacy in public databases. In: Second Theory of Cryptography Conference (TCC 2005). (2005) 363–385
19. Ullman, J., Vadhan, S.: Pcps and the hardness of generating private synthetic data. In: Proceedings of the 8th conference on Theory of cryptography. TCC'11 (2011) 400–416
20. Dwork, C., Naor, M., Reingold, O., Rothblum, G.N., Vadhan, S.: On the complexity of differentially private data release: efficient algorithms and hardness results. In: Proceedings of the 41st annual ACM symposium on Theory of computing. STOC '09 (2009) 381–390
21. Chen, X.H., Dempster, A.P., Liu, J.S.: Weighted finite population sampling to maximize entropy. Biometrika **81**(3) (1994) pp. 457–469