Towards Privacy for Social Networks: A Zero-Knowledge Based Definition of Privacy*

Johannes Gehrke
Cornell University
johannes@cs.cornell.edu

Edward Lui
Cornell University
luied@cs.cornell.edu

Rafael Pass
Cornell University
rafael@cs.cornell.edu

May 20, 2011

Abstract

We put forward a zero-knowledge based definition of privacy. Our notion is strictly stronger than the notion of differential privacy and is particularly attractive when modeling privacy in social networks. We furthermore demonstrate that it can be meaningfully achieved for tasks such as computing averages, fractions, histograms, and a variety of graph parameters and properties, such as average degree and distance to connectivity. Our results are obtained by establishing a connection between zero-knowledge privacy and sample complexity, and by leveraging recent sublinear time algorithms.

1 Introduction

Data privacy is a fundamental problem in today's information age. Enormous amounts of data are collected by government agencies, search engines, social networking systems, hospitals, financial institutions, and other organizations, and are stored in databases. There are huge social benefits in analyzing this data; however, it is important that sensitive information about individuals who have contributed to the data is not leaked to users analyzing the data. Thus, one of the main goals is to release statistical information about the population who have contributed to the data without breaching their individual privacy.

Many privacy definitions and schemes have been proposed in the past (see [CKLM09] and [FWCY10] for surveys). However, many of them have been shown to be insufficient by describing realistic attacks on such schemes (e.g., see [Kif09]). The notion of *differential privacy* [DMNS06, Dwo06], 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 of the database access mechanism changes very little so that the output before and after the change are "ϵ-close" (where a specific notion of closeness of distributions is used). This notion has quickly become the standard notion of privacy, and mechanisms for releasing a variety of functions (including histogram queries, principal component analysis, learning, and many more (see [Dwo09] for a recent survey)) have been developed.

As we shall argue, however, although differential privacy provides a strong privacy guarantee, there are realistic social network settings where these guarantees might not be strong enough.

*A preliminary version of this paper appeared in the Eighth Theory of Cryptography Conference (TCC 2011) (see [GLP11]).

Roughly speaking, differential privacy says that whether you're in the database or not is inconsequential for your privacy (i.e., the output of the database mechanism is essentially the same). But this doesn't mean your privacy is protected; the information provided by your *friends* might already breach your privacy.

Alternatively, differential privacy can be rephrased as requiring that an adversary does not learn much more about an individual from the mechanism than what she could learn from knowing everyone else in the database (see the appendix of [DMNS06] for a formalization of this statement). Such a privacy guarantee is not sufficiently strong in the setting of social networks where an individual's *friends* are strongly correlated with the individual; in essence, "If I know your friends, I know you." (Indeed, a recent study [JM09] indicates that an individual's sexual orientation can be accurately predicted just by looking at the person's Facebook friends.) We now give a concrete example to illustrate how a differentially private mechanism can violate the privacy of individuals in a social network setting.

Example (Democrats vs. Republicans)**.** Consider a social network of n people that are grouped into cliques of size 200. In each clique, either at least 80% of the people are Democrats, or at least 80% are Republicans. However, assume that the number of Democrats overall is roughly the same as the number of Republicans. Now, consider a mechanism that computes the proportion (in $[0, 1]$) of Democrats in each clique and adds just enough Laplacian noise to satisfy ϵ-differential privacy for a small ϵ, say $\epsilon = 0.1$. For example, to achieve ϵ-differential privacy, it suffices to add $Lap(\frac{1}{200\epsilon})$ noise[1] to each clique independently, since if a single person changes his or her political preference, the proportion for the person's clique changes by $\frac{1}{200}$ (see Proposition 1 in [DMNS06]).

Since the mechanism satisfies ϵ-differential privacy for a small ϵ, one may think that it is safe to release such information without violating the privacy of any particular person. That is, the released data should not allow us to guess correctly with probability significantly greater than $\frac{1}{2}$ whether a particular person is a Democrat or a Republican. However, this is not the case. With $\epsilon = 0.1$, $Lap(\frac{1}{200\epsilon})$ is a small amount of noise, so with high probability, the data released will tell us the main political preference for any particular clique. An adversary that knows which clique a person is in will be able to correctly guess the political preference of that person with probability close to 80%.

For a more detailed explanation and analysis of the above example, see Appendix A.

Remark. In the above example, we assume that the graph structure of the social network is known and that the adversary can identify which clique an individual is in. Such information is commonly available: Graph structures of (anonymized) social networks are often released; these may include a predefined or natural clustering of the people (nodes) into cliques. Furthermore, an adversary may often also figure out the identity of various nodes in the graph (e.g., see [BDK07, HMJ+08]); in fact, by participating in the social network before the anonymized graph is published, an adversary can even target specific individuals of his or her choice (see [BDK07]).

Differential privacy says that the output of the mechanism does not depend much on any particular individual's data in the database. Thus, in the above example, a person has little reason not to truthfully report his political preference. However, this does not necessarily imply that the mechanism does not violate the person's privacy. In situations where a social network provides auxiliary information about an individual, that person's privacy can be violated even if he decides to not have his information included.

[1] $Lap(\lambda)$ is the Laplace distribution with probability density function $f_\lambda(x) = \frac{1}{2\lambda} e^{\frac{|x|}{\lambda}}$.

It is already known that differential privacy may not provide a strong enough privacy guarantee when an adversary has specific auxiliary information about an individual. For example, it was pointed out in [Dwo06] that if an adversary knows the auxiliary information "person A is two inches shorter than the average American woman", and if a differentially private mechanism accurately releases the average height of American women, then the adversary learns person A's height (which is assumed to be sensitive information in this example). In this example, the adversary has very specific auxiliary information about an individual that is usually hard to obtain. However, in the Democrats vs. Republicans example, the auxiliary information (the graph and clique structure) about individuals is more general and more easily accessible. Since social network settings contain large amounts of auxiliary information and correlation between individuals, differential privacy is usually not strong enough in such settings.

One may argue that there are versions of differential privacy that protect the privacy of groups of individuals, and that the mechanism in the Democrats vs. Republicans example does not satisfy these stronger definitions of privacy. While this is true, the main point here is that differential privacy will not protect the privacy of an individual, even though the definition is designed for individual privacy. Furthermore, even if we had used a differentially private mechanism that ensures privacy for groups of size 200 (i.e., the size of each clique), it might still be possible to deduce information about an individual by looking at the *friends of the friends* of the individual; this includes a significantly larger number of individuals.[2]

1.1 Towards a Zero-Knowledge Definition of Privacy

In 1977, Dalenius [Dal77] stated a privacy goal for statistical databases: anything about an individual that can be learned from the database can also be learned without access to the database. This would be a very desirable notion of privacy. Unfortunately, Dwork and Naor [Dwo06, DN08] demonstrated a general impossibility result showing that a formalization of Dalenius's goal along the lines of semantic security for cryptosystems cannot be achieved, assuming that the database gives any non-trivial utility.

Our aim is to provide a privacy definition along the lines of Dalenius, and more precisely, relying on the notion of *zero-knowledge* from cryptography. In this context, the traditional notion of zero-knowledge says that an adversary gains essentially "zero additional knowledge" by accessing the mechanism. More precisely, whatever an adversary can compute by accessing the mechanism can essentially also be computed without accessing the mechanism. A mechanism satisfying this property would be private but utterly useless, since the mechanism provides essentially no information. The whole point of releasing data is to provide utility; thus, this extreme notion of zero-knowledge, which we now call "complete zero-knowledge", is not very applicable in this setting.

Intuitively, we want the mechanism to not release any additional information beyond some *"aggregate information"* that is considered acceptable to release. To capture this requirement, we use the notion of a "simulator" from zero-knowledge, and we require that a simulator with the acceptable aggregate information can essentially compute whatever an adversary can compute by accessing the mechanism. Our zero-knowledge privacy definition is thus stated relative to some class of algorithms providing acceptable aggregate information.

1.1.1 Aggregate Information

The question is how to define appropriate classes of aggregate information. We focus on the case where the aggregate information is any information that can be obtained from k random

[2]The number of "friends of friends" is usually larger than the square of the number of friends (see [New03]).

samples/rows (each of which corresponds to one individual's data) of the database, where the data of the person the adversary wants to attack has been concealed. The value of k can be carefully chosen so that the aggregate information obtained does not allow one to infer (much) information about the concealed data. The simulator is given this aggregate information and has to compute what the adversary essentially computes, even though the adversary has access to the mechanism. This ensures that the mechanism does not release any additional information beyond this "k random sample" aggregate information given to the simulator.

Differential privacy can be described using our zero-knowledge privacy definition by considering simulators that are given aggregate information consisting of the data of all but one individual in the database; this is the same as aggregate information consisting of "k random samples" with $k = n$, where n is the number of rows in the database (recall that the data of the individual the adversary wants to attack is concealed), which we formally prove later. For k less than n, such as $k = \sqrt{n}$, we obtain notions of privacy that are stronger than differential privacy. For example, we later show that the mechanism in the Democrats vs. Republicans example does not satisfy our zero-knowledge privacy definition when $k = o(n)$ and n is sufficiently large.

We may also consider more general models of aggregate information that are specific to graphs representing social networks; in this context we focus on random samples with some exploration of the neighborhood of each sample.

1.2 Our Results

We consider two different settings for releasing information. In the first setting, we consider statistical (row) databases in a setting where an adversary might have auxiliary information, such as from a social network, and we focus on releasing traditional statistics (e.g., averages, fractions, histograms, etc.) from a database. As explained earlier, differential privacy may not be strong enough in such a setting, so we use our zero-knowledge privacy definition instead. In the second setting, we consider graphs with personal data that represent social networks, and we focus on releasing information directly related to a social network, such as properties of the graph structure.

Setting #1. Computing functions on databases with zero-knowledge privacy: In this setting, we focus on computing functions mapping databases to $\mathbb{R}^m$. We give a characterization of the functions that can be released with zero-knowledge privacy in terms of their *sample complexity*—i.e., how accurate the function can be approximated using random samples from the input database. More precisely, functions with low sample complexity can be computed accurately by a zero-knowledge private mechanism, and vice versa. (It is already known that functions with low sample complexity can be computed with differential privacy (see [DMNS06]), but here we show that the stronger notion of zero-knowledge privacy can be achieved.) In this result, the zero-knowledge private mechanism we construct simply adds Laplacian noise appropriately calibrated to the sample complexity of the function.

Many common queries on statistical databases have low sample complexity, including averages, fraction queries, counting queries, and coarse histogram queries. (In general, it would seem that any "meaningful" query function for statistical databases should have relatively low sample complexity if we think of the rows of the database as random samples from some large underlying population.) We also show that for functions with low sample complexity, we can use differentially private mechanisms to construct zero-knowledge private mechanisms. Using this result, we construct zero-knowledge private mechanisms for such functions while providing decent utility guarantees. All of these results can be found in Section 3.

We also consider mechanisms that answer a class of queries simultaneously, and we generalize the notion of sample complexity to classes of query functions. By showing that a class of fraction queries with low VC dimension has low sample complexity, we are able to use existing differentially private mechanisms for classes of fraction queries to construct zero-knowledge private mechanisms, resulting in improved accuracy for fraction queries. These results can be found in Section 4.

Setting #2. Releasing graph structure information with zero-knowledge privacy: In this setting, we consider a graph representing a social network, and we focus on privately releasing information about the structure of the graph. We use our zero-knowledge privacy definition, since the released information can be combined with auxiliary information such as an adversary's knowledge and/or previously released data (e.g., graph structure information) to breach the privacy of individuals.

The connection between sample complexity and zero-knowledge privacy highlights an interesting connection between *sublinear time algorithms* and privacy. As it turns out, many of the recently developed sublinear algorithms on graphs proceed by picking random samples (vertices) and performing some local exploration; we are able to leverage these algorithms to privately release graph structure information, such as average degree and distance to properties such as connectivity and cycle-freeness. We discuss these results in Section 5.

2 Zero-Knowledge Privacy

2.1 Definitions

Let $\mathcal{D}$ be the collection of all databases whose rows are elements (e.g., tuples) from some data universe X. For convenience, we will assume that X contains an element $\perp$, which can be used to conceal the true value of a row. Given a database D, let $|D|$ denote the number of rows in D. For any integer n, let $[n]$ denote the set $\{1, \ldots, n\}$. For any database $D \in \mathcal{D}$, any integer $i \in [|D|]$, and any element $v \in X$, let (D_{-i}, v) denote the database D with row i replaced by the element v.

In this paper, mechanisms, adversaries, and simulators are simply randomized algorithms that play certain roles in our definitions. Let San be a mechanism that operates on databases in $\mathcal{D}$. For any database $D \in \mathcal{D}$, any adversary A, and any $z \in \{0,1\}^*$, let $Out_A(A(z) \leftrightarrow San(D))$ denote the random variable representing the output of A on input z after interacting with the mechanism San operating on the database D. Note that San can be interactive or non-interactive. If San is non-interactive, then $San(D)$ sends information (e.g., a sanitized database) to A and then halts immediately; the adversary A then tries to breach the privacy of some individual in the database D.

Let agg be any class of randomized algorithms that provide aggregate information to simulators, as described in Section 1.1.1. We refer to agg as a *model of aggregate information*.

Definition 1. We say that San is **ϵ-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 $D \in X^n$, every $z \in \{0,1\}^*$, every integer $i \in [n]$, and every $W \subseteq \{0,1\}^*$, the following hold:

- $\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W] \leq e^\epsilon \cdot \Pr[S(z, T(D_{-i}, \perp), i, n) \in W]$
- $\Pr[S(z, T(D_{-i}, \perp), i, n) \in W] \leq e^\epsilon \cdot \Pr[Out_A(A(z) \leftrightarrow San(D)) \in W]$

The probabilities are over the random coins of San and A, and T and S, respectively.

Intuitively, the above definition says that whatever an adversary can compute by accessing the mechanism can essentially also be computed without accessing the mechanism but with certain aggregate information (specified by agg). The adversary in the latter scenario is represented by the simulator S. The definition requires that the adversary's output distribution is close to that of the simulator. This ensures that the mechanism essentially does not release any additional information beyond what is allowed by agg. When the algorithm T provides aggregate information to the simulator S, the data of individual i is concealed so that the aggregate information does not depend directly on individual i's data. However, in the setting of social networks, the aggregate information may still depend on people's data that are correlated with individual i in reality, such as the data of individual i's friends. Thus, the role played by agg is very important in the context of social networks.

To measure the closeness of the adversary's output and the simulator's output, we use the same closeness measure as in differential privacy (as opposed to, say, statistical difference) for the same reasons. As explained in [DMNS06], consider a mechanism that outputs the contents of a randomly chosen row. Suppose agg is defined so that it includes the algorithm that simply outputs its input $(D_{-i}, \perp)$ to the simulator (which is the case of differential privacy; see Section 1.1.1 and 2.2). Then, a simulator can also choose a random row and then simulate the adversary with the chosen row sent to the simulated adversary. The real adversary's output will be very close to the simulator's output in statistical difference ($1/n$ to be precise); however, it is clear that the mechanism always leaks private information about some individual.

Remark. Our ϵ-zero-knowledge privacy definition can be easily extended to (ϵ, δ)-zero-knowledge privacy, where we also allow an additive error of δ on the RHS of the inequalities. We can further extend our definition to (c, ϵ, δ)-zero-knowledge privacy to protect the privacy of any group of c individuals simultaneously. To obtain this more general definition, we would change "$i \in [n]$" to "$I \subseteq [n]$ with $1 \leq |I| \leq c$", and "$S(z, (D_{-i}, \perp), i, n)$" to "$S(z, (D_{-I}, \vec{\perp}), I, n)$", where $(D_{-I}, \vec{\perp})$ denotes the database D with the rows at positions I replaced by $\perp$. We use this more general definition when we consider group privacy.

Remark. In our zero-knowledge privacy definition, we consider computationally unbounded simulators. We can also consider PPT simulators by requiring that the mechanism San and the adversary A are PPT algorithms, and agg is a class of PPT algorithms. All of these algorithms would be PPT in n, the size of the database. With minor modifications, the results of this paper would still hold in this case.

The choice of agg determines the type and amount of aggregate information given to the simulator, and should be decided based on the context in which the zero-knowledge privacy definition is used. The aggregate information should not depend much on data that is highly correlated with the data of a single person, since such aggregate information may be used to breach the privacy of that person. For example, in the context of social networks, such aggregate information should not depend much on any person and the people closely connected to that person, such as his or her friends. By choosing agg carefully, we ensure that the mechanism essentially does not release any additional information beyond what is considered acceptable. We first consider the model of aggregate information where T in the definition of zero-knowledge privacy chooses $k(n)$ random samples. Let $k : \mathbb{N} \to \mathbb{N}$ be any function.

- $RS(k(\cdot)) = k(\cdot)$ random samples: the class of algorithms T such that on input a database $D \in X^n$, T chooses $k(n)$ random samples (rows) from D uniformly without replacement, and then performs any computation on these samples without reading any of the other rows of D.

Note that with such samples, T can emulate choosing $k(n)$ random samples with replacement, or a combination of without replacement and with replacement.

$k(n)$ should be carefully chosen so that the aggregate information obtained does not allow one to infer (much) information about the concealed data. For $k(n) = 0$, the simulator is given no aggregate information at all, which is the case of complete zero-knowledge. For $k(n) = n$, the simulator is given all the rows of the original database except for the target individual i, which is the case of differential privacy (as we prove later). For $k(n)$ strictly in between 0 and n, we obtain notions of privacy that are stronger than differential privacy. For example, one can consider $k(n) = o(n)$, such as $k(n) = \sqrt{n}$.

In the setting of a social network, $k(n)$ can be chosen so that when $k(n)$ random samples are chosen from $(D_{-i}, \perp)$, with very high probability, for (almost) all individuals j, very few of the $k(n)$ chosen samples will be in individual j's local neighborhood in the social network graph. This way, the aggregate information released by the mechanism depends very little on data that is highly correlated with the data of a single individual. The choice of $k(n)$ would depend on various properties of the graph structure, such as clustering coefficient, edge density, and degree distribution. The choice of $k(n)$ would also depend on the amount of correlation between the data of adjacent or close vertices (individuals) in the graph, and the type of information released by the mechanism. In this model of aggregate information, vertices (individuals) in the graph with more adjacent vertices (e.g., representing friends) may have less privacy than those with fewer adjacent vertices. However, this is often the case in social networks, where having more links/connections to other people may result in less privacy.

One can also consider other models of aggregate information, such as the class of algorithms T such that on input a database $D \in X^n$, T reads each row with at most a certain probability, say $\frac{k(n)}{n}$. This class of algorithms, which we call "$k(\cdot)$ adaptive samples" and denote by $AS(k(\cdot))$, is more general and contains $RS(k(\cdot))$. However, there are some "bad" mechanisms that are zero-knowledge private with respect to $AS(k(\cdot))$ but intuitively violate the privacy of individuals. We now give an example of such a mechanism.

Example. Recall the Democrats vs. Republicans example in the introduction. Now, consider a new mechanism that chooses a clique uniformly at random, computes the proportion of Democrats in the chosen clique, adds $Lap(\frac{1}{200\epsilon})$ noise to the computed proportion, and then outputs the clique number/identifier and the noisy proportion. For the same reasons as in the Democrats vs. Republicans example, this mechanism clearly violates the privacy of the individuals in the chosen clique. However, this mechanism is still ϵ-zero-knowledge private with respect to $AS(k(\cdot))$ as long as $k(n)$ isn't too small.

Intuitively, a simulator with $T \in AS(k(\cdot))$ providing aggregate information can simulate the mechanism by doing the same thing the mechanism does, since the mechanism reads each row with probability $\frac{200}{n}$ (each clique is chosen with probability $\frac{200}{n}$, since there are $\frac{n}{200}$ cliques). This works as long as $\frac{200}{n} \leq \frac{k(n)}{n}$, since T is only allowed to read each row with probability at most $\frac{k(n)}{n}$. We assume that T can easily determine which rows belong to a particular clique; for example, the rows of the database can be ordered so that individuals belonging to the same clique appear consecutively in the database, or the nodes in the published social network graph can have distinct labels in $\{1, \ldots, n\}$, and the political preference for node i is stored in row i of the database.

In Section 5, we consider other models of aggregate information that take more into consideration the graph structure of a social network. Note that zero-knowledge privacy does not necessarily guarantee that the privacy of every individual is completely protected. Zero-knowledge privacy is

defined with respect to a model of aggregate information, and such aggregate information may still leak some sensitive information about an individual in certain scenarios.

Composition: Just as for differentially private mechanisms, mechanisms that are ϵ-zero-knowledge private with respect to $RS(k(\cdot))$ also compose nicely.

Proposition 2. *Suppose San_1 is ϵ_1-zero-knowledge private with respect to $RS(k_1(\cdot))$ and San_2 is ϵ_2-zero-knowledge private with respect to $RS(k_2(\cdot))$. Then, the mechanism San obtained by (sequentially) composing San_1 with San_2 is $(\epsilon_1+\epsilon_2)$-zero-knowledge private with respect to $RS((k_1+k_2)(\cdot))$.*

Proof. Let $k(n) = k_1(n) + k_2(n)$, and let $T_1 \in RS(k_1(\cdot))$ and $T_2 \in RS(k_2(\cdot))$ be the aggregate information algorithms guaranteed by the zero-knowledge privacy of San_1 and San_2, respectively. Let T be an algorithm in $RS(k(\cdot))$ that, on input a database $D \in X^n$, chooses $k_1(n)$ random samples as in T_1, chooses $k_2(n)$ random samples as in T_2, runs T_1 and T_2 on D separately using the chosen samples, and then outputs $(T_1(D), T_2(D))$. Let A be any adversary. It is easy to decompose A into two adversaries A_1 and A_2, where A_j is the part of A that interacts with San_j. The output of A_1 contains information describing the state (including the work tape) of A after finishing its interaction with San_1. A_2 expects its input z to be the output of A_1 so that it can start interacting with San_2 with the same information A would have at this point of the interaction. Let S_j be the (guaranteed) simulator for San_j and A_j.

Let S be a simulator that, on input $(z, T(D_{-i}, \perp), i, n) = (z, (T_1(D_{-i}, \perp), T_2(D_{-i}, \perp)), i, n)$, first runs the simulator S_1 on input $(z, T_1(D_{-i}, \perp), i, n)$ to get $z' := S_1(z, T_1(D_{-i}, \perp), i, n)$, and then runs the simulator S_2 on input $(z', T_2(D_{-i}, \perp), i, n)$. Let $D \in X^n$, $z \in \{0,1\}^*$, $i \in [n]$, and $W \subseteq \{0,1\}^*$. Let $Y = Supp(S_1(z, T_1(D_{-i}, \perp), i, n))$. We note that $Y = Supp(Out_{A_1}(A_1(z) \leftrightarrow San_1(D)))$. Now, observe that

$$
\begin{aligned}
&\left|\ln\left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W]}{\Pr[S(z, T(D_{-i}, \perp), i, n) \in W]}\right)\right| \\
\leq\ &\left|\ln\left(\frac{\sum_{z' \in Y} \Pr[Out_{A_2}(A_2(z') \leftrightarrow San_2(D)) \in W] \Pr[Out_{A_1}(A_1(z) \leftrightarrow San_1(D)) = z']}{\sum_{z' \in Y} \Pr[S_2(z', T_2(D_{-i}, \perp), i, n) \in W] \Pr[S_1(z, T_1(D_{-i}, \perp), i, n) = z']}\right)\right| \\
\leq\ &\epsilon_1 + \epsilon_2.
\end{aligned}
$$

□

Group Privacy: A nice feature of differential privacy is that ϵ-differential privacy implies $(c, c\epsilon)$-differential privacy for groups of size c (see [Dwo06] and the appendix in [DMNS06]). We have a similar group privacy guarantee for ϵ-zero-knowledge privacy.

Proposition 3. *Suppose San is ϵ-zero-knowledge private with respect to agg. Then, for every $c \geq 1$, San is also $(c, (2c-1)\epsilon)$-zero-knowledge private with respect to agg.*

Proof. Let T be the algorithm in agg guaranteed by the ϵ-zero-knowledge privacy of San. Let $c \geq 1$. Consider any adversary A, and let S be the simulator for A and San. Let $D \in X^n$, $z \in \{0,1\}^*$, $I \subseteq [n]$ with $1 \leq |I| \leq c$, and $W \subseteq \{0,1\}^*$. Let i be any integer in I. Then, by the ϵ-zero-knowledge privacy of San, we have

$$
\left|\ln\left(\frac{\Pr[S(z, T(D_{-I}, \vec{\perp}), i, n) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D_{-(I \setminus \{i\})}, \vec{\perp})) \in W]}\right)\right| \leq \epsilon. \tag{1}
$$

We later show that ϵ-zero-knowledge privacy implies 2ϵ-differential privacy (Proposition 7), so San is 2ϵ-differentially private and thus $(c-1, 2(c-1)\epsilon)$-differentially private. As a result, we have

$$\left|\ln\left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D_{-(I\setminus\{i\})}, \vec{\perp})) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W]}\right)\right| \leq 2(c-1)\epsilon. \tag{2}$$

Combining (1) and (2) from above, we get

$$\left|\ln\left(\frac{\Pr[S(z, T(D_{-I}, \vec{\perp}), i, n) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W]}\right)\right| \leq (2c-1)\epsilon.$$

□

It can be easily shown that (ϵ, δ)-differential privacy implies $(0, e^\epsilon - 1 + \delta)$-differential privacy (see [DKM+06] or Section 2.2 for the definition of (ϵ, δ)-differential privacy), which implies $(c, 0, c(e^\epsilon - 1 + \delta))$-differential privacy for groups of size c. We have a similar group privacy guarantee for (ϵ, δ)-zero-knowledge privacy.

Proposition 4. *Suppose San is (ϵ, δ)-zero-knowledge private with respect to agg. Then, for every $c \geq 1$, San is also $(c, 0, (2c-1)(e^\epsilon - 1 + \delta))$-zero-knowledge private with respect to agg.*

Proof. Let T be the algorithm in agg guaranteed by the (ϵ, δ)-zero-knowledge privacy of San. Let $c \geq 1$. Consider any adversary A, and let S be the simulator for A and San. Then, for every database $D \in X^n$, $z \in \{0,1\}^*$, $i \in [n]$, and $W \subseteq \{0,1\}^*$, we have

$$\begin{aligned}
&\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W] \leq e^\epsilon \cdot \Pr[S(z, T(D_{-i}, \perp), i, n) \in W] + \delta \\
\leq &\Pr[S(z, T(D_{-i}, \perp), i, n) \in W] + (e^\epsilon - 1) + \delta, \text{ and}
\end{aligned}$$

$$\begin{aligned}
&\Pr[S(z, T(D_{-i}, \perp), i, n) \in W] \leq e^\epsilon \cdot \Pr[Out_A(A(z) \leftrightarrow San(D)) \in W] + \delta \\
\leq &\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W] + (e^\epsilon - 1) + \delta, \text{ so}
\end{aligned}$$

$$|\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W] - \Pr[S(z, T(D_{-i}, \perp), i, n) \in W]| \leq e^\epsilon - 1 + \delta.$$

Let $D \in X^n$, $z \in \{0,1\}^*$, $I \subseteq [n]$ with $1 \leq |I| \leq c$, and $W \subseteq \{0,1\}^*$. Let i be any integer in I. Then, we have

$$|\Pr[S(z, T(D_{-I}, \vec{\perp}), i, n) \in W] - \Pr[Out_A(A(z) \leftrightarrow San(D_{-(I\setminus\{i\})}, \vec{\perp})) \in W]| \leq e^\epsilon - 1 + \delta. \tag{1}$$

Also, for every pair of databases $D', D'' \in X^n$ differing in one row, say row j, we have

$$\begin{aligned}
&|\Pr[Out_A(A(z) \leftrightarrow San(D')) \in W] - \Pr[Out_A(A(z) \leftrightarrow San(D'')) \in W]| \\
\leq &|\Pr[Out_A(A(z) \leftrightarrow San(D')) \in W] - \Pr[S(z, T(D_{-j}, \perp), j, n) \in W]| \\
&+ |\Pr[S(z, T(D_{-j}, \perp), j, n) \in W] - \Pr[Out_A(A(z) \leftrightarrow San(D'')) \in W]| \\
\leq &\, 2(e^\epsilon - 1 + \delta).
\end{aligned}$$

Now, we note that the database $(D_{-(I\setminus\{i\})}, \vec{\perp})$ differs from the database D in at most $c-1$ rows. By considering a sequence of at most c databases where the first database is $(D_{-(I\setminus\{i\})}, \vec{\perp})$, the last database is D, and adjacent databases differ in only one row (and thus are "$2(e^\epsilon - 1 + \delta)$-close" to one another), we have

$$\begin{aligned}
&|\Pr[Out_A(A(z) \leftrightarrow San(D_{-(I\setminus\{i\})}, \vec{\perp})) \in W] - \Pr[Out_A(A(z) \leftrightarrow San(D)) \in W]| \\
\leq &\, 2(c-1)(e^\epsilon - 1 + \delta).
\end{aligned} \tag{2}$$

Combining (1) and (2) from above yields the result. □

For $agg = RS(k(\cdot))$, we also have the following group privacy guarantee for (ϵ, δ)-zero-knowledge privacy.

Proposition 5. *Suppose San is (ϵ, δ)-zero-knowledge private with respect to $RS(k(\cdot))$. Then, for every $c \geq 1$, San is also $(c, \epsilon, \delta + e^{\epsilon}(c-1)\frac{k(n)}{n})$-zero-knowledge private with respect to $RS(k(\cdot))$.*

Intuitively, for $k(n)$ sufficiently smaller than n, (ϵ, δ)-zero-knowledge privacy with respect to $RS(k(\cdot))$ actually implies some notion of group privacy, since the algorithm T (in the privacy definition) chooses each row with probability $k(n)/n$. Thus, T chooses any row of a fixed group of c rows with probability at most $ck(n)/n$. If this probability is very small, then the output of T and thus the simulator S does not depend much on any group of c rows.

Proof. Fix $c \geq 1$. Since *San* is (ϵ, δ)-zero-knowledge private with respect to $RS(k(\cdot))$, there exists a $T \in RS(k(\cdot))$ such that for every adversary A, there exists a simulator S such that for every $D \in X^n$, $z \in \{0,1\}^*$, $i \in [n]$, and $W \subseteq \{0,1\}^*$, we have

$$\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W] \leq e^{\epsilon} \Pr[S(z, T(D_{-i}, \perp), i, n) \in W] + \delta \quad \text{and}$$

$$\Pr[S(z, T(D_{-i}, \perp), i, n) \in W] \leq e^{\epsilon} \Pr[Out_A(A(z) \leftrightarrow San(D)) \in W] + \delta.$$

Let A be any adversary, and let S be the simulator guaranteed by the zero-knowledge privacy of *San*. Let S' be a simulator that, on input $(z, T(D_{-I}, \vec{\perp}), I, n)$, outputs $S(z, T(D_{-I}, \vec{\perp}), i, n)$, where i is the smallest integer in I. Let $D \in X^n$, $z \in \{0,1\}^*$, $I \subseteq [n]$ with $1 \leq |I| \leq c$, and $W \subseteq \{0,1\}^*$. Let i be the smallest integer in I.

Let E be the event that T reads a row at any of the positions specified by $I \setminus \{i\}$ (the input of T is inferred from context). We note that conditioned on $\overline{E}$, $T(D_{-i}, \perp)$ and $T(D_{-I}, \vec{\perp})$ have the same distribution. Since $T \in RS(k(\cdot))$ and $|I \setminus \{i\}| \leq c-1$, we have $\Pr[E] \leq (c-1) \cdot \frac{k(n)}{n}$ when T is run on any database $D' \in X^n$. Now, observe that

$$\begin{aligned}
&\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W] \\
&\leq e^{\epsilon} \cdot \Pr[S(z, T(D_{-i}, \perp), i, n) \in W] + \delta \\
&= e^{\epsilon} \cdot (\Pr[S(z, T(D_{-i}, \perp), i, n) \in W \mid \overline{E}] \cdot \Pr[\overline{E}] + \Pr[S(z, T(D_{-i}, \perp), i, n) \in W \mid E] \cdot \Pr[E]) + \delta \\
&\leq e^{\epsilon} \cdot (\Pr[S'(z, T(D_{-I}, \vec{\perp}), I, n) \in W \mid \overline{E}] \cdot \Pr[\overline{E}] + \Pr[E]) + \delta \\
&\leq e^{\epsilon} \cdot \left(\Pr[S'(z, T(D_{-I}, \vec{\perp}), I, n) \in W] + (c-1) \cdot \frac{k(n)}{n} \right) + \delta \\
&= e^{\epsilon} \cdot \Pr[S'(z, T(D_{-I}, \vec{\perp}), I, n) \in W] + e^{\epsilon}(c-1) \cdot \frac{k(n)}{n} + \delta.
\end{aligned}$$

We also have

$$\begin{aligned}
&\Pr[S'(z, T(D_{-I}, \vec{\perp}), I, n) \in W] \\
&= \Pr[S'(z, T(D_{-I}, \vec{\perp}), I, n) \in W \mid \overline{E}] \cdot \Pr[\overline{E}] + \Pr[S'(z, T(D_{-I}, \vec{\perp}), I, n) \in W \mid E] \cdot \Pr[E] \\
&\leq \Pr[S(z, T(D_{-i}, \perp), i, n) \in W \mid \overline{E}] \cdot \Pr[\overline{E}] + \Pr[E] \\
&\leq \Pr[S(z, T(D_{-i}, \perp), i, n) \in W] + (c-1) \cdot \frac{k(n)}{n} \\
&\leq e^{\epsilon} \cdot \Pr[Out_A(A(z) \leftrightarrow San(D)) \in W] + \delta + e^{\epsilon}(c-1) \cdot \frac{k(n)}{n}.
\end{aligned}$$

□

2.2 Differential Privacy vs. Zero-Knowledge Privacy

In this section, we compare differential privacy to our zero-knowledge privacy definition. We first state the definition of differential privacy in a form similar to our zero-knowledge privacy definition in order to more easily compare the two. For any pair of databases $D, D' \in X^n$, let $H(D, D')$ denote the number of rows in which D and D' differ, comparing row-wise.

Definition 6. We say that San is **ϵ-differentially private** if for every adversary A, every $z \in \{0,1\}^*$, every pair of databases $D, D' \in X^n$ with $H(D, D') \leq 1$, and every $W \subseteq \{0,1\}^*$, we have

$$\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W] \leq e^\epsilon \cdot \Pr[Out_A(A(z) \leftrightarrow San(D')) \in W],$$

where the probabilities are over the random coins of San and A. For **(c, ϵ)-differential privacy** (for groups of size c), the "$H(D, D') \leq 1$" is changed to "$H(D, D') \leq c$". For **(ϵ, δ)-differential privacy**, we allow an additive error of δ on the RHS of the inequality in the definition.

Proposition 7. *Suppose San is ϵ-zero-knowledge private with respect to any class agg. Then, San is 2ϵ-differentially private.*

Proof. Let A be any adversary, let $z \in \{0,1\}^*$, let $D', D'' \in X^n$ with $H(D', D'') \leq 1$, and let $W \subseteq \{0,1\}^*$. Since $H(D', D'') \leq 1$, there exists an integer $i \in [n]$ such that $D'_{-i} = D''_{-i}$. Since San is ϵ-zero-knowledge private with respect to agg, there exists a $T \in agg$ and a simulator S such that for every database $D \in X^n$, we have

$$\left| \ln \left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W]}{\Pr[S(z, T(D_{-i}, \perp), i, n) \in W]} \right) \right| \leq \epsilon.$$

Now, observe that

$$\begin{aligned}
&\left| \ln \left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D')) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D'')) \in W]} \right) \right| \\
\leq\ &\left| \ln \left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D')) \in W]}{\Pr[S(z, T(D'_{-i}, \perp), i, n) \in W]} \right) \right| + \left| \ln \left(\frac{\Pr[S(z, T(D'_{-i}, \perp), i, n) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D'')) \in W]} \right) \right| \\
\leq\ &\epsilon + \left| \ln \left(\frac{\Pr[S(z, T(D''_{-i}, \perp), i, n) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D'')) \in W]} \right) \right| \leq 2\epsilon.
\end{aligned}$$

□

Proposition 8. *Suppose San is ϵ-differentially private. Then, San is ϵ-zero-knowledge private with respect to $RS(n)$.*

Proof. Let T be an algorithm in $RS(n)$ that, on input a database $D' \in X^n$, chooses n "random" samples from D' without replacement (i.e., chooses all the rows of the database), and then outputs the whole database D'. Let A be any adversary. Let S be the simulator that, on input $(z, (D_{-i}, \perp), i, n)$, simulates the interaction between $A(z)$ and $San(D_{-i}, \perp)$, and outputs whatever A outputs in the simulated interaction. Thus, we have $S(z, T(D_{-i}, \perp), i, n) = Out_A(A(z) \leftrightarrow San(D_{-i}, \perp))$. Let $D \in X^n$, $z \in \{0,1\}^*$, $i \in [n]$, and $W \subseteq \{0,1\}^*$. Since San is ϵ-differentially private and $H(D, (D_{-i}, \perp)) \leq 1$, we have

$$\left| \ln \left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W]}{\Pr[S(z, T(D_{-i}, \perp), i, n) \in W]} \right) \right| = \left| \ln \left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D_{-i}, \perp)) \in W]} \right) \right| \leq \epsilon.$$

□

Remark. If we consider PPT simulators in the definition of zero-knowledge privacy instead of computationally unbounded simulators, then we require *San* in Proposition 8 to be PPT as well.

Combining Propositions 7 and 8, we see that our zero-knowledge privacy definition includes differential privacy as a special case (up to a factor of 2 for ϵ).

2.3 Revisiting the Democrats vs. Republicans Example

Recall the Democrats vs. Republicans example in the introduction. The mechanism in the example is ϵ-differentially private for some small ϵ, even though the privacy of individuals is clearly violated. However, the mechanism is not zero-knowledge private in general. Suppose that the people's political preferences are stored in a database $D \in X^n$.

Proposition 9. *Fix $\epsilon > 0$, $c \geq 1$, and any function $k(\cdot)$ such that $k(n) = o(n)$. Let San be a mechanism that on input $D \in X^n$ computes the proportion of Democrats in each clique and adds $Lap(\frac{c}{200\epsilon})$ noise to each proportion independently. Then, San is (c, ϵ)-differentially private, but for every constant $\epsilon' > 0$ and every sufficiently large n, San is not ϵ'-zero-knowledge private with respect to $RS(k(\cdot))$.*

Intuitively, *San* is not ϵ'-zero-knowledge private with respect to $RS(k(\cdot))$ because for sufficiently large n, an adversary having only $k(n) = o(n)$ random samples would not have any samples in many of the cliques, so the adversary would know nothing about many of the cliques. Therefore, the adversary does gain knowledge by accessing the mechanism, which gives some information about every clique since the amount of noise added to each clique is constant.

Proof. We note that when a single person changes his or her political preference, the vector of proportions of Democrats changes by $\frac{1}{200}$ in L_1 distance. Thus, by Proposition 1 in [DMNS06], *San* is (ϵ/c)-differentially private, which implies that *San* is (c, ϵ)-differential private (see [Dwo06] and the appendix in [DMNS06]), as required.

Let $\epsilon' > 0$. Let A be the adversary that simply outputs whatever the mechanism releases. To obtain a contradiction, suppose there exists a $T \in RS(k(\cdot))$ and a simulator S for A satisfying the required condition in the definition of ϵ'-zero-knowledge privacy. Recall that there are 200 people in each clique. Let $\lambda = \frac{c}{200\epsilon}$, let $K \geq 600\epsilon'\lambda$ be a constant such that 200 divides K, and let $n \geq K$ such that 200 divides n. Let $W = (\mathbb{R}_{\leq 0})^{K/200} \times \mathbb{R}^{n/200 - K/200}$, and let $z \in \{0,1\}^*$. Then, for every $D \in X^n$, we have

$$\left| \ln \left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D)) \in W]}{\Pr[S(z, T(D_{-1}, \perp), 1, n) \in W]} \right) \right| \leq \epsilon'.$$

Without loss of generality, suppose that the rows of a database in X^n are ordered so that the first 200 rows correspond to 200 people in the same clique, and the next 200 rows correspond to 200 people in the same clique, and so on. Let D_1 be the database $(0, \ldots, 0)$ of size n, and let D_2 be the database $(1^K, 0, \ldots, 0)$ of size n, where $1^K = (1, \ldots, 1)$ is of size K. Let $X_1, \ldots, X_{n/200} \sim Lap(\lambda)$ (independently), and let $X = (X_1, \ldots, X_{n/200})$. Now, observe the following:

$$\begin{aligned}
&\ln \left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D_1)) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W]} \right) = \ln \left(\frac{\Pr[(0^{n/200}) + X \in W]}{\Pr[(1^{K/200}, 0, \ldots, 0) + X \in W]} \right) \\
&= \ln \left(\frac{\prod_{j=1}^{K/200} \Pr[X_j \in \mathbb{R}_{\leq 0}]}{\prod_{j=1}^{K/200} \Pr[X_j \in (-\infty, -1]]} \right) = \ln \left(\frac{(\frac{1}{2})^{K/200}}{\prod_{j=1}^{K/200} F_\lambda(-1)} \right) = \ln \left(\frac{(\frac{1}{2})^{K/200}}{\prod_{j=1}^{K/200} (\frac{1}{2} e^{-1/\lambda})} \right) \\
&= \ln(e^{\frac{K}{200 \cdot \lambda}}) = \frac{K}{200 \cdot \lambda} \geq \frac{600\epsilon'\lambda}{200 \cdot \lambda} \geq 3\epsilon', \qquad (1)
\end{aligned}$$

where $F_\lambda(x) = \frac{1}{2}e^{x/\lambda}$ is the cumulative distribution function of $Lap(\lambda)$ for $x \le 0$. Let E_K denote the event that T does not read any of the first K rows of its input (the input of T is inferred from context). Now, observe that

$$\begin{aligned}
&\ln\left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D_1)) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W]}\right) \\
&= \ln\left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D_1)) \in W]}{\Pr[S(z, T((D_1)_{-1}, \perp), 1, n) \in W]}\right) + \ln\left(\frac{\Pr[S(z, T((D_1)_{-1}, \perp), 1, n) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W]}\right) \\
&\le \epsilon' + \ln\left(\frac{\Pr[S(z, T((D_1)_{-1}, \perp), 1, n) \in W \mid E_K]\Pr[E_K] + \Pr[S(z, T((D_1)_{-1}, \perp), 1, n) \in W \mid \overline{E_K}]\Pr[\overline{E_K}]}{\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W]}\right) \\
&\le \epsilon' + \ln\left(\frac{\Pr[S(z, T((D_2)_{-1}, \perp), 1, n) \in W \mid E_K] \cdot \Pr[E_K]}{\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W]} + \frac{\Pr[\overline{E_K}]}{\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W]}\right) \\
&\le \epsilon' + \ln\left(\frac{\Pr[S(z, T((D_2)_{-1}, \perp), 1, n) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W]} + \frac{\Pr[\overline{E_K}]}{\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W]}\right)
\end{aligned}$$

Since $T \in RS(k(\cdot))$ and $k(n) = o(n)$, we have the numerator $\Pr[\overline{E_K}] \to 0$ as $n \to \infty$. However, the denominator $\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W] = (\frac{1}{2})^{K/200}e^{-K/(200\lambda)}$ (partly computed earlier) is a constant. Since ln is continuous and $\frac{\Pr[S(z,T((D_2)_{-1},\perp),1,n)\in W]}{\Pr[Out_A(A(z)\leftrightarrow San(D_2))\in W]} \in [e^{-\epsilon'}, e^{\epsilon'}]$ for all n, we have that for sufficiently large n,

$$\begin{aligned}
&\epsilon' + \ln\left(\frac{\Pr[S(z, T((D_2)_{-1}, \perp), 1, n) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W]} + \frac{\Pr[\overline{E_K}]}{\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W]}\right) \\
&\le \epsilon' + \ln\left(\frac{\Pr[S(z, T((D_2)_{-1}, \perp), 1, n) \in W]}{\Pr[Out_A(A(z) \leftrightarrow San(D_2)) \in W]}\right) + \frac{\epsilon'}{2} \;\le\; \epsilon' + \epsilon' + \frac{\epsilon'}{2} \;\le\; \frac{5\epsilon'}{2}.
\end{aligned}$$

Thus, for sufficiently large n, we have $\ln\left(\frac{\Pr[Out_A(A(z)\leftrightarrow San(D_1))\in W]}{\Pr[Out_A(A(z)\leftrightarrow San(D_2))\in W]}\right) \le \frac{5\epsilon'}{2}$, which contradicts (1) above. □

Remark. In the Democrats vs. Republicans example, even if *San* adds $Lap(\frac{1}{\epsilon})$ noise to achieve $(200, \epsilon)$-differential privacy so that the privacy of each clique (and thus each person) is protected, the mechanism would still fail to be ϵ'-zero-knowledge private with respect to $RS(k(\cdot))$ for any constant $\epsilon' > 0$ when n is sufficiently large (see Proposition 9). Thus, zero-knowledge privacy with respect to $RS(k(\cdot))$ with $k(n) = o(n)$ seems to provide an unnecessarily strong privacy guarantee in this particular example. However, this is mainly because the clique size is fixed and known to be 200, and we have assumed that the only correlation between people's political preferences that exists is within a clique. In a more realistic social network, there would be cliques of various sizes, and the correlation between people's data would be more complicated. For example, an adversary knowing your friends' friends may still be able to infer a lot of information about you.

3 Characterizing Zero-Knowledge Privacy

In this section, we focus on constructing zero-knowledge private mechanisms that compute a function mapping databases in X^n to $\mathbb{R}^m$, and we characterize the set of functions that can be computed with zero-knowledge privacy. These are precisely the functions with low sample complexity, i.e., can be approximated (accurately) using only limited information from the database, such as k random samples.

We quantify the error in approximating a function $g : X^n \to \mathbb{R}^m$ using L_1 distance. Let the L_1-sensitivity of g be defined by $\Delta(g) = \max\{||g(D') - g(D'')||_1 : D', D'' \in X^n \text{ s.t. } H(D', D'') \leq 1\}$. Let $\mathcal{C}$ be any class of randomized algorithms.

Definition 10. A function $g : X^n \to \mathbb{R}^m$ is said to have (δ, β)**-sample complexity with respect to** $\mathcal{C}$ if there exists an algorithm $T \in \mathcal{C}$ such that for every database $D \in X^n$, we have $T(D) \in \mathbb{R}^m$ and

$$\Pr[||T(D) - g(D)||_1 \leq \delta] \geq 1 - \beta.$$

T is said to be a (δ, β)**-sampler for** g **with respect to** $\mathcal{C}$.

Remark. If we consider PPT simulators in the definition of zero-knowledge privacy instead of computationally unbounded simulators, then we would require here that $\mathcal{C}$ is a class of PPT algorithms (PPT in n, the size of the database). Thus, in the definition of (δ, β)-sample complexity, we would consider a family of functions (one for each value of n) that can be computed in PPT, and the sampler T would be PPT in n.

It was shown in [DMNS06] that functions with low sample complexity with respect to $RS(k(\cdot))$ have low sensitivity as well.

Lemma 11 ([DMNS06])**.** *Suppose $g : X^n \to \mathbb{R}^m$ has (δ, β)-sample complexity with respect to $RS(k(\cdot))$ for some $\beta < \frac{1-k(n)/n}{2}$. Then, $\Delta(g) \leq 2\delta$.*

As mentioned in [DMNS06], the converse of the above lemma is not true, i.e., not all functions with low sensitivity have low sample complexity (see [DMNS06] for an example). This should be no surprise, since functions with low sensitivity have accurate differentially private mechanisms, while functions with low sample complexity have accurate zero-knowledge private mechanisms. We already know that zero-knowledge privacy is stronger than differential privacy, as illustrated by the Democrats vs. Republicans example.

We now state how the sample complexity of a function is related to the amount of noise a mechanism needs to add to the function value in order to achieve a certain level of zero-knowledge privacy.

Proposition 12. *Suppose $g : X^n \to [a, b]^m$ has (δ, β)-sample complexity with respect to some $\mathcal{C}$. Then, the mechanism $San(D) = g(D) + (X_1, \ldots, X_m)$, where $X_j \sim Lap(\lambda)$ for $j = 1, \ldots, m$ independently, is $\ln((1-\beta)e^{\frac{\Delta(g)+\delta}{\lambda}} + \beta e^{\frac{(b-a)m}{\lambda}})$-zero-knowledge private with respect to $\mathcal{C}$.*

Intuitively, *San* should be zero-knowledge private because a simulator can simulate *San* by first approximating $g(D)$ by running a sampler $T \in \mathcal{C}$ for g, and then adding the same amount of noise as *San*; the error in approximating $g(D)$ is blurred by the added noise so that the simulator's output distribution is close to *San*'s output distribution.

Proof. Let T be a (δ, β)-sampler for g with respect to $\mathcal{C}$. Let A be any adversary. Let S be a simulator that, on input $(z, T(D_{-i}, \perp), i, n)$, first checks whether $T(D_{-i}, \perp)$ is in $[a, b]^m$; if not, S projects $T(D_{-i}, \perp)$ onto the set $[a, b]^m$ (with respect to L_1 distance) so that the accuracy of $T(D_{-i}, \perp)$ is improved and $||g(D) - T(D_{-i}, \perp)||_1 \leq (b-a)m$ always holds, which we use later. From here on, $T(D_{-i}, \perp)$ is treated as a random variable that reflects the possible modification S may perform. The simulator S computes $T(D_{-i}, \perp) + (X_1, \ldots, X_m)$, which we will denote using the random variable $S'(z, T(D_{-i}, \perp), i, n)$. S then simulates the computation of $A(z)$ with $S'(z, T(D_{-i}, \perp), i, n)$ sent to A as a message, and outputs whatever A outputs.

Let $D \in X^n$, $z \in \{0,1\}^*$, $i \in [n]$. Fix $x \in T(D_{-i}, \perp)$ and $s \in \mathbb{R}^m$. Then, we have

$$\max\left\{\frac{f_\lambda(s-g(D))}{f_\lambda(s-x)}, \frac{f_\lambda(s-x)}{f_\lambda(s-g(D))}\right\} = \max\left\{e^{(\frac{1}{\lambda}\cdot(||s-x||_1 - ||s-g(D)||_1))}, e^{(\frac{1}{\lambda}\cdot(||s-g(D)||_1 - ||s-x||_1))}\right\}$$
$$\leq e^{(\frac{1}{\lambda}\cdot||g(D)-x||_1)} \leq e^{(\frac{1}{\lambda}\cdot(||g(D)-g(D_{-i},\perp)||_1 + ||g(D_{-i},\perp)-x||_1))} \leq e^{(\frac{1}{\lambda}\cdot(\Delta(g)+||g(D_{-i},\perp)-x||_1))}. \quad (1)$$

Since $||g(D)-x||_1 \leq (b-a)m$ always holds, we also have

$$\max\left\{\frac{f_\lambda(s-g(D))}{f_\lambda(s-x)}, \frac{f_\lambda(s-x)}{f_\lambda(s-g(D))}\right\} \leq e^{(\frac{1}{\lambda}\cdot||g(D)-x||_1)} \leq e^{\frac{(b-a)m}{\lambda}}. \quad (2)$$

Since T is a (δ,β)-sampler for g, we have $\Pr[||g(D_{-i},\perp) - T(D_{-i},\perp)||_1 \leq \delta] \geq 1-\beta$. Thus, using (1) and (2) above, we have

$$\ln\left(\frac{\sum_{x\in T(D_{-i},\perp)} f_\lambda(s-x)\cdot\Pr[T(D_{-i},\perp)=x]}{f_\lambda(s-g(D))}\right) \leq \ln((1-\beta)e^{\frac{\Delta(g)+\delta}{\lambda}} + \beta e^{\frac{(b-a)m}{\lambda}}).$$

Now, using (1) and (2) again, we also have

$$\begin{aligned}
&\ln\left(\frac{f_\lambda(s-g(D))}{\sum_{x\in T(D_{-i},\perp)} f_\lambda(s-x)\cdot\Pr[T(D_{-i},\perp)=x]}\right)\\
=\ &-\ln\left(\frac{\sum_{x\in T(D_{-i},\perp)} f_\lambda(s-x)\cdot\Pr[T(D_{-i},\perp)=x]}{f_\lambda(s-g(D))}\right)\\
\leq\ &-\ln((1-\beta)e^{-\frac{\Delta(g)+\delta}{\lambda}} + \beta e^{-\frac{(b-a)m}{\lambda}}) = \ln(((1-\beta)e^{-\frac{\Delta(g)+\delta}{\lambda}} + \beta e^{-\frac{(b-a)m}{\lambda}})^{-1})\\
\leq\ &\ln((1-\beta)e^{\frac{\Delta(g)+\delta}{\lambda}} + \beta e^{\frac{(b-a)m}{\lambda}}),
\end{aligned}$$

where the last inequality follows from the fact that the function $f(x) = x^{-1}$ is convex for $x > 0$. Then, for every $s \in \mathbb{R}^n$, we have

$$\left|\ln\left(\frac{\Pr[San(D)=s]}{\Pr[S'(z,T(D_{-i},\perp),i,n)=s]}\right)\right| = \left|\ln\left(\frac{f_\lambda(s-g(D))}{\sum_{x\in T(D_{-i},\perp)} f_\lambda(s-x)\cdot\Pr[T(D_{-i},\perp)=x]}\right)\right|$$
$$\leq \ln((1-\beta)e^{\frac{\Delta(g)+\delta}{\lambda}} + \beta e^{\frac{(b-a)m}{\lambda}}).$$

Thus, for every $W \subseteq \{0,1\}^*$, we have $\left|\ln\left(\frac{\Pr[Out_A(A(z)\leftrightarrow San(D))\in W]}{\Pr[S(z,T(D_{-i},\perp),i,n)\in W]}\right)\right| \leq \ln((1-\beta)e^{\frac{\Delta(g)+\delta}{\lambda}} + \beta e^{\frac{(b-a)m}{\lambda}})$. □

Corollary 13. *Suppose $g : X^n \to [a,b]^m$ has (δ,β)-sample complexity with respect to $RS(k(\cdot))$ for some $\beta < \frac{1-k(n)/n}{2}$. Then, the mechanism $San(D) = g(D) + (X_1,\ldots,X_m)$, where $X_j \sim Lap(\lambda)$ for $j = 1,\ldots,m$ independently, is $\ln((1-\beta)e^{\frac{3\delta}{\lambda}} + \beta e^{\frac{(b-a)m}{\lambda}})$-zero-knowledge private with respect to $RS(k(\cdot))$.*

Proof. This follows from combining Proposition 12 and Lemma 11. □

Using Proposition 12, we can recover the basic mechanism in [DMNS06] that is ϵ-differentially private.

Corollary 14. *Let $g : X^n \to [a,b]^m$ and $\epsilon > 0$. A mechanism San for g that adds $Lap(\frac{\Delta(g)}{\epsilon})$ noise to $g(D)$ is ϵ-zero-knowledge private with respect to $RS(n)$.*

Proof. We note that every function $g : X^n \to \mathbb{R}^m$ has $(0, 0)$-sample complexity with respect to $RS(n)$. The corollary follows by applying Proposition 12. □

We now show how the zero-knowledge privacy and utility properties of a mechanism computing a function is related to the sample complexity of the function. A class of algorithms *agg* is said to be *closed under postprocessing* if for any $T \in agg$ and any algorithm M, the composition of M and T (i.e., the algorithm that first runs T on the input and then runs M on the output of T) is also in *agg*. We note that $RS(k(\cdot))$ is closed under postprocessing.

Proposition 15. *Let agg be any class of algorithms that is closed under postprocessing, and suppose a function $g : X^n \to \mathbb{R}^m$ has a mechanism San such that the following hold:*

- *Utility:* $\Pr[||San(D) - g(D)||_1 \leq \delta] \geq 1 - \beta$ *for every* $D \in X^n$
- *Privacy: San is ϵ-zero-knowledge private with respect to agg.*

Then, g has $(\delta, \frac{\beta+(e^\epsilon-1)}{e^\epsilon})$-sample complexity with respect to agg.

The intuition is that the zero-knowledge privacy of *San* guarantees that *San* can be simulated by a simulator S that is given aggregate information provided by some algorithm $T \in agg$. Thus, an algorithm that runs T and then S will be able to approximate g with accuracy similar to that of *San*.

Proof. Let A be an adversary that simply outputs whatever *San* releases. Since *San* is ϵ-zero-knowledge private with respect to *agg*, there exists a $B \in agg$ and a simulator S such that for every $D \in X^n$, $z \in \{0,1\}^*$, and $t \in \{0,1\}^*$, we have

$$\left|\ln\left(\frac{\Pr[Out_A(A(z) \leftrightarrow San(D)) = t]}{\Pr[S(z, B(D_{-1}, \perp), 1, n) = t]}\right)\right| \leq \epsilon. \tag{1}$$

Fix $z \in \{0,1\}^*$. Let T be an algorithm that, on input $D \in X^n$, first runs B on $(D_{-1}, \perp)$, then runs S on $(z, B(D_{-1}, \perp), 1, n)$, and then outputs $S(z, B(D_{-1}, \perp), 1, n)$. Since $B \in agg$, S is an algorithm, and *agg* is closed under postprocessing, we have that T is in *agg*. Let $D \in X^n$. We have

$$\begin{aligned}
&Pr[||T(D) - g(D)||_1 \leq \delta] \;=\; \Pr[||S(z, B(D_{-1}, \perp), 1, n) - g(D)||_1 \leq \delta] \\
&= \sum_{t \in Supp(S(z, B(D_{-1}, \perp), 1, n))} \Pr[||t - g(D)||_1 \leq \delta] \cdot \Pr[S(z, B(D_{-1}, \perp), 1, n) = t] \\
&\geq \sum_{t \in Supp(Out_A(A(z) \leftrightarrow San(D)))} \Pr[||t - g(D)||_1 \leq \delta] \cdot \frac{1}{e^\epsilon} \Pr[Out_A(A(z) \leftrightarrow San(D)) = t] \\
&= \frac{1}{e^\epsilon} \Pr[||Out_A(A(z) \leftrightarrow San(D)) - g(D)||_1 \leq \delta] \;=\; \frac{1}{e^\epsilon} \Pr[||San(D) - g(D)||_1 \leq \delta] \\
&\geq \frac{1}{e^\epsilon}(1 - \beta) \;=\; 1 - \frac{\beta + (e^\epsilon - 1)}{e^\epsilon},
\end{aligned}$$

where the first inequality is due to (1). Thus, T is a $(\delta, \frac{\beta+(e^\epsilon-1)}{e^\epsilon})$-sampler for g with respect to *agg*. □

3.1 Simple Examples of Zero-Knowledge Private Mechanisms

In this section, we show how to construct some simple examples of zero-knowledge private mechanisms with respect to $RS(k(\cdot))$.

Example (Averages). Let $n \geq 1$, $k = k(n)$. Let $avg : [0,1]^n \to [0,1]$ be defined by $avg(D) = \frac{\sum_{i=1}^{n} D_i}{n}$, and let $San(D) = avg(D) + Lap(\lambda)$, where $\lambda > 0$. Let T be an algorithm that, on input a database $D \in [0,1]^n$, chooses k random samples from D uniformly, and then outputs the average of the k random samples. By Hoeffding's inequality, we have $\Pr[|T(D) - avg(D)| \leq \delta] \geq 1 - 2e^{-2k\delta^2}$. Thus, avg has $(\delta, 2e^{-2k\delta^2})$-sample complexity with respect to $RS(k(\cdot))$. By Proposition 12, San is $\ln(e^{\frac{1}{\lambda}(\frac{1}{n}+\delta)} + 2e^{\frac{1}{\lambda}-2k\delta^2})$-zero-knowledge private with respect to $RS(k(\cdot))$.

Let $\epsilon \in (0,1]$. We choose $\delta = \frac{1}{k^{1/3}}$ and $\lambda = \frac{1}{\epsilon}(\frac{1}{n}+\delta) = \frac{1}{\epsilon}(\frac{1}{n} + \frac{1}{k^{1/3}})$ so that $\ln(e^{\frac{1}{\lambda}(\frac{1}{n}+\delta)} + 2e^{\frac{1}{\lambda}-2k\delta^2})$ $= \ln(e^{\epsilon} + 2e^{\frac{\epsilon}{1/n+k^{-1/3}}-2k^{1/3}}) \leq \ln(e^{\epsilon} + 2e^{-k^{1/3}}) \leq \epsilon + 2e^{-k^{1/3}}$. Thus, we have the following result:

- By adding $Lap(\frac{1}{\epsilon}(\frac{1}{n} + \frac{1}{k^{1/3}})) = Lap(O(\frac{1}{\epsilon k^{1/3}}))$ noise to $avg(D)$, San is $(\epsilon + 2e^{-k^{1/3}})$-zero-knowledge private with respect to $RS(k(\cdot))$.

Our example mechanism for computing averages comes from the general connection between sample complexity and zero-knowledge privacy (Proposition 12), which holds for any model agg of aggregate information. For computing averages, we can actually construct a mechanism with better utility by choosing $k(n)$ random samples without replacement from the input database $D \in X^n$ and then running a differentially private mechanism on the chosen samples. We will show that such a mechanism is zero-knowledge private with respect to $RS(k(\cdot))$ and has even better privacy parameters than the differentially private mechanism, due to the initial sampling step.

In general, this "Sample and DP-Sanitize" method works for query functions that can be approximated using random samples (e.g., averages, fractions, and histograms), and allows us to convert differentially private mechanisms to zero-knowledge private mechanisms with respect to $RS(k(\cdot))$. We now show what privacy guarantees are obtained by the Sample and DP-Sanitize method.

Proposition 16 (Sample and DP-Sanitize)**.** *Let San_{DP} be any (ϵ, δ)-differentially private mechanism. Let San be any mechanism that, on input $D \in X^n$, chooses $k = k(n)$ random samples* without replacement*, runs San_{DP} on the chosen samples, and then performs any computation on the output of San_{DP} without reading the input database D again. Then, the following hold:*

- *San is (ϵ, δ)-zero-knowledge private with respect to $RS(k(\cdot))$.*
- *San is $(2\ln(1+\frac{k}{n}(e^{\epsilon}-1)), (2+\frac{k}{n}(e^{\epsilon}-1))\frac{k}{n}\delta)$-zero-knowledge private with respect to $RS(k(\cdot))$.*
- *If $\epsilon \leq 1$, then San is $(\frac{4k}{n}\epsilon, \frac{4k}{n}\delta)$-zero-knowledge private with respect to $RS(k(\cdot))$.*

Intuitively, San is zero-knowledge private with respect to $RS(k(\cdot))$ because San_{DP} is differentially private and is only run on k random samples; also, San has better privacy parameters than those of San_{DP} because of the extra noise added from choosing only k random samples.

Proof. We observe that the mechanism San itself is in $RS(k(\cdot))$. Thus, let $T = San$. Let $n \geq 1$, $D \in X^n$, and $i \in [n]$.

We first show that San is (ϵ, δ)-zero-knowledge private with respect to $RS(k(\cdot))$. Consider $San(D)$ and $T(D_{-i}, \perp) = San(D_{-i}, \perp)$. We note that $San(D)$ and $San(D_{-i}, \perp)$ have the same output distribution when both San's choose the same k random samples and the samples do not

contain row i. When both San's choose the same k random samples and the samples do contain row i, the two databases formed by the chosen samples of the two San's (respectively) will differ in exactly one row. Since both San's run an (ϵ, δ)-differentially private mechanism, namely San_{DP}, on the chosen samples, and since San does not use the original input database in its computation afterwards, it is easy to see that the output distribution of the two San's satisfy the closeness condition in the (ϵ, δ)-zero-knowledge privacy definition. Since the simulator S in the privacy definition gets $T(D_{-i}, \perp) = San(D_{-i}, \perp)$ as one of its inputs, it is easy to show that San is (ϵ, δ)-zero-knowledge private with respect to $RS(k(\cdot))$.

We now show that San is $(2\ln(1 + \frac{k}{n}(e^\epsilon - 1)), (2 + \frac{k}{n}(e^\epsilon - 1))\frac{k}{n}\delta)$-zero-knowledge private with respect to $RS(k(\cdot))$. If $k = n$, then this follows from the fact that San is (ϵ, δ)-zero-knowledge private with respect to $RS(k(\cdot))$. Thus, we now assume that $k \leq n-1$. We will show that $San(D)$ and $T(D_{-i}, \perp) = San(D_{-i}, \perp)$ are "$(2\ln(1+\frac{k}{n}(e^\epsilon-1)), (2+\frac{k}{n}(e^\epsilon-1))\frac{k}{n}\delta)$"-close. Abusing notation, let $San(D_{-i})$ denote the output of San on input D_{-i} but San chooses $k = k(n)$ random samples (without replacement) instead of $k(|D_{-i}|) = k(n-1)$ random samples. Our strategy is to show that $San(D)$ is close to $San(D_{-i})$ and $San(D_{-i})$ is close to $San(D_{-i}, \perp)$. Let $W \subseteq \{0,1\}^*$, and let E be the event that row i is chosen when San chooses k random samples. Observe that

$$\begin{aligned}
&\Pr[San(D) \in W] = \Pr[San(D) \in W \mid E]\Pr[E] + \Pr[San(D) \in W \mid \overline{E}]\Pr[\overline{E}] \\
&\leq (e^\epsilon \Pr[San(D_{-i}) \in W] + \delta)\Pr[E] + \Pr[San(D_{-i}) \in W](1 - \Pr[E]) \\
&= (1 + \Pr[E](e^\epsilon - 1))\Pr[San(D_{-i}) \in W] + \Pr[E]\delta.
\end{aligned}$$

We also have

$$\begin{aligned}
&\Pr[San(D) \in W] = \Pr[San(D) \in W \mid E]\Pr[E] + \Pr[San(D) \in W \mid \overline{E}]\Pr[\overline{E}] \\
&\geq e^{-\epsilon}(\Pr[San(D_{-i}) \in W] - \delta)\Pr[E] + \Pr[San(D_{-i}) \in W](1 - \Pr[E]) \\
&= \Pr[San(D_{-i}) \in W](\Pr[E]e^{-\epsilon} + (1 - \Pr[E])) - e^{-\epsilon}\Pr[E]\delta \\
\implies &\Pr[San(D_{-i}) \in W] \leq \frac{1}{\Pr[E]e^{-\epsilon} + (1 - \Pr[E])}\Pr[San(D) \in W] + \frac{1}{\Pr[E] + (1 - \Pr[E])e^\epsilon}\Pr[E]\delta \\
&\leq (\Pr[E]e^\epsilon + (1 - \Pr[E]))\Pr[San(D) \in W] + \Pr[E]\delta \\
&\leq (1 + \Pr[E](e^\epsilon - 1))\Pr[San(D) \in W] + \Pr[E]\delta,
\end{aligned}$$

where the second last inequality follows from the fact that the function $f(x) = \frac{1}{x}$ is convex for $x > 0$. Thus, we have the following:

- $\Pr[San(D) \in W] \leq (1 + \Pr[E](e^\epsilon - 1))\Pr[San(D_{-i}) \in W] + \Pr[E]\delta$
- $\Pr[San(D_{-i}) \in W] \leq (1 + \Pr[E](e^\epsilon - 1))\Pr[San(D) \in W] + \Pr[E]\delta$

Using the same argument as above but with $(D_{-i}, \perp)$ in place of D, we get the following:

- $\Pr[San(D_{-i}, \perp) \in W] \leq (1 + \Pr[E](e^\epsilon - 1))\Pr[San(D_{-i}) \in W] + \Pr[E]\delta$
- $\Pr[San(D_{-i}) \in W] \leq (1 + \Pr[E](e^\epsilon - 1))\Pr[San(D_{-i}, \perp) \in W] + \Pr[E]\delta$

Combining the results above and noting that $\Pr[E] = \frac{k}{n}$, we have the following:

- $\Pr[San(D) \in W] \leq (1 + \frac{k}{n}(e^\epsilon - 1))^2 \Pr[San(D_{-i}, \perp) \in W] + (2 + \frac{k}{n}(e^\epsilon - 1))\frac{k}{n}\delta$
- $\Pr[San(D_{-i}, \perp) \in W] \leq (1 + \frac{k}{n}(e^\epsilon - 1))^2 \Pr[San(D) \in W] + (2 + \frac{k}{n}(e^\epsilon - 1))\frac{k}{n}\delta$

It easily follows that San is $(2\ln(1+\frac{k}{n}(e^\epsilon-1)), (2+\frac{k}{n}(e^\epsilon-1))\frac{k}{n}\delta)$-zero-knowledge private with respect to $RS(k(\cdot))$. Now, suppose that $\epsilon \leq 1$. Then, one can easily verify that $e^\epsilon - 1 \leq 2\epsilon$, so $1+\frac{k}{n}(e^\epsilon-1) \leq e^{\frac{2k}{n}\epsilon}$ and $2+\frac{k}{n}(e^\epsilon-1) \leq 4$. Thus, San is $(\frac{4k}{n}\epsilon, \frac{4k}{n}\delta)$-zero-knowledge private with respect to $RS(k(\cdot))$. □

We now use the Sample and DP-Sanitize method (Proposition 16) to construct some zero-knowledge private mechanisms.

In the examples below, let $n \geq 1$, $k = k(n)$, and $\epsilon, \beta \in (0,1]$. If $D \in X^n$ is a database, let $\widehat{D}$ be a random variable representing a database formed by choosing k random samples from D uniformly without replacement.

Example (Improved Accuracy for Averages). Let $X = [0,1]$, and let $avg(D) = \frac{\sum_{i=1}^{|D|} D_i}{|D|}$. Let $San(D) = avg(\widehat{D}) + Lap(\frac{1}{\epsilon k})$ for $D \in X^n$. Then, by Proposition 16, San is $\frac{4k}{n}\epsilon$-zero-knowledge private with respect to $RS(k(\cdot))$.

Also, by Hoeffding's inequality (which still holds when the sampling is done without replacement as opposed to with replacement (e.g., see [Hoe63])), we have $\Pr[|avg(\widehat{D}) - avg(D)| \geq \alpha] \leq 2e^{-2k\alpha^2}$, and the RHS is $\leq \frac{\beta}{2}$ if $\alpha \geq \frac{1}{\sqrt{k}}\sqrt{\frac{1}{2}\ln(\frac{4}{\beta})}$; thus, we have $\Pr[|avg(\widehat{D}) - avg(D)| \geq \frac{1}{\sqrt{k}}\sqrt{\frac{1}{2}\ln(\frac{4}{\beta})}] \leq \frac{\beta}{2}$. One can also easily verify that for $Y \sim Lap(\frac{1}{\epsilon k})$, we have $\Pr[|Y| \geq \alpha] = e^{-\epsilon k \alpha}$, and the RHS is $\leq \frac{\beta}{2}$ if $\alpha \geq \frac{1}{\epsilon k}\ln(\frac{2}{\beta})$; thus, we have $\Pr[|Y| \geq \frac{1}{\epsilon k}\ln(\frac{2}{\beta})] \leq \frac{\beta}{2}$. Thus, by the union bound, we have the following result:

- For $D \in X^n$, $San(D) = avg(\widehat{D}) + Lap(\frac{1}{\epsilon k})$ approximates $avg(D)$ to within an additive error of $\frac{1}{\sqrt{k}}\sqrt{\frac{1}{2}\ln(\frac{4}{\beta})} + \frac{1}{\epsilon k}\ln(\frac{2}{\beta})$ with probability at least $1-\beta$, and is $\frac{4k}{n}\epsilon$-zero-knowledge private with respect to $RS(k(\cdot))$.

This mechanism is usually more accurate than the mechanism in the earlier example for averages, which adds at least $Lap(\frac{1}{\epsilon k^{1/3}})$ noise and thus is accurate to within an additive error of $\frac{1}{\epsilon k^{1/3}}\ln(\frac{2}{\beta})$ with probability at most $1-\beta$.

Example (Fraction Queries: Fraction of rows satisfying some property P). Let $P : X \to \{0,1\}$ be the predicate representing some property of a row. Let $frac_P(D) = \frac{\sum_{i=1}^{|D|} P(D_i)}{|D|}$, which is the fraction of rows satisfying property P. Since $frac_P(D)$ can be viewed as the average of the numbers $\{P(D_i)\}_{i=1}^n$, we can get the same result as in the example for averages:

- For $D \in X^n$, $San(D) = frac(\widehat{D}) + Lap(\frac{1}{\epsilon k})$ approximates $frac(D)$ to within an additive error of $\frac{1}{\sqrt{k}}\sqrt{\frac{1}{2}\ln(\frac{4}{\beta})} + \frac{1}{\epsilon k}\ln(\frac{2}{\beta})$ with probability at least $1-\beta$, and is $\frac{4k}{n}\epsilon$-zero-knowledge private with respect to $RS(k(\cdot))$.

Example (Counting Queries: Number of rows satisfying some property P). Let $P : X \to \{0,1\}$ be the predicate representing some property of a row. Let $count(D) = \sum_{i=1}^n P(D_i)$, which is the number of rows satisfying property P. Since $g(D)$ is simply a fraction query but scaled by a factor of n, we can get the same result as in the example for fraction queries except that the error is scaled by a factor of n:

- For $D \in X^n$, $San(D) = n \cdot (frac(\widehat{D}) + Lap(\frac{1}{\epsilon k}))$ approximates $count(D) = n \cdot frac_P(D)$ to within an additive error of $\frac{n}{\sqrt{k}}\sqrt{\frac{1}{2}\ln(\frac{4}{\beta})} + \frac{n}{\epsilon k}\ln(\frac{2}{\beta})$ with probability at least $1-\beta$, and is $\frac{4k}{n}\epsilon$-zero-knowledge private with respect to $RS(k(\cdot))$.

Example (Histograms). Let $B_1, \ldots, B_m$ be any partition of X with m blocks. We refer to each B_i as a bin. Let $hist(D) = (b_1, \ldots, b_n)$, where $b_i = |\{j \in [n] : D_j \in B_i\}|$ is the number of rows of D that belong to bin B_i. Given a database D, let $\widetilde{D}_1, \ldots, \widetilde{D}_m$ be independent random variables representing databases formed by choosing $\frac{k}{m}$ random samples from the database D uniformly without replacement.

We can construct a zero-knowledge private mechanism (with respect to $RS(k(\cdot))$) that computes the histogram with respect to the bins $B_1, \ldots, B_m$, by composing San_i for $i = 1, \ldots, m$, where San_i is any zero-knowledge private mechanism (with respect to $RS(\frac{1}{m}k(\cdot))$) for estimating the number of rows in the i^{th} bin, and then applying our composition result (Propsition 2). Using our mechanism for counting queries, we can define San_i so that it approximates the number of rows in the i^{th} bin to within an additive error of $\frac{n\sqrt{m}}{\sqrt{k}}\sqrt{\frac{1}{2}\ln(\frac{4m}{\beta})} + \frac{nm}{\epsilon k}\ln(\frac{2m}{\beta})$ with probability at least $1 - \frac{\beta}{m}$, and is $\frac{4k}{nm}\epsilon$-zero-knowledge private with respect to $RS(\frac{1}{m}k(\cdot))$. Then, applying the union bound and our composition result (Propsition 2), we get the following result:

- For $D \in X^n$, $San(D) = (San_1(\widetilde{D}_1), \ldots, San_m(\widetilde{D}_m))$ approximates $hist(D)$ to within an error (with respect to L_1 distance) of $\frac{nm^{3/2}}{\sqrt{k}}\sqrt{\frac{1}{2}\ln(\frac{4m}{\beta})} + \frac{nm^2}{\epsilon k}\ln(\frac{2m}{\beta})$ with probability at least $1-\beta$, and is $\frac{4k}{n}\epsilon$-zero-knowledge private with respect to $RS(k(\cdot))$.

4 Answering a Class of Queries Simultaneously

In this section, we consider mechanisms that answer a class of query functions simultaneously. We generalize the notion of sample complexity (with respect to $RS(k(\cdot))$) to classes of query functions and show a connection between differential privacy and zero-knowledge privacy for any class of query functions with low sample complexity. In particular, we show that for any class $\mathcal{Q}$ of query functions that can be approximated simultaneously using random samples, any differentially private mechanism that is "useful" for $\mathcal{Q}$ can be converted to a zero-knowledge private mechanism that is useful for $\mathcal{Q}$, similar to the Sample and DP-Sanitize method. We also show that any class of fraction queries with low VC dimension can be approximated simultaneously using random samples, so we can use existing differentially private mechanisms (e.g., the ones in [BLR08] and [DRV10]) to obtain zero-knowledge private mechanisms for any class of fraction queries with low VC dimension.

Let X^* denote the set of all databases whose rows are elements from the data universe X. In this section, a query is a function from X^* to $\mathbb{R}^m$ for some m.

In this section, we consider mechanisms that answer a class $\mathcal{Q}$ of queries simulataneously by outputting a "synopsis" (e.g., a synthetic database) that allows us to answer all the queries in $\mathcal{Q}$. A *synopsis* is a pair $(\widetilde{D}, R)$, where $\widetilde{D}$ is any data structure (containing data), and R is a description of any deterministic "query-answering" algorithm that, on input a data structure $\widetilde{D}$ and a query $q : X^* \to \mathbb{R}^m$, answers the query by reading $\widetilde{D}$ and outputting some vector in $\mathbb{R}^m$.

Let R_{DB} be the usual query-answering algorithm for databases that, on input a database $D \in X^*$ and a query $q : X^* \to \mathbb{R}^m$, answers with $q(D)$. If D is a database, then (D, R_{DB}) is an example of a synopsis. If $\widehat{D}$ is a database obtained by choosing k random samples from D, then another example of a synopsis is $(\widehat{D}, R_{\widehat{D}})$, where $R_{\widehat{D}}$ is a query-answering algorithm that approximates a given counting query q on the larger database D by computing $q(\widehat{D})$ and then scaling the answer by $\frac{|D|}{k}$ (to compensate for the fact that $\widehat{D}$ contained only k random samples from D).

Let $\mathcal{Q}$ be any class of queries that map databases in X^* to vectors in $\mathbb{R}^m$ for some m. We now define what it means for two synopses to be close to one another with respect to $\mathcal{Q}$.

Definition 17. Two synopses $(\widetilde{D}, R)$ and $(\widetilde{D}', R')$ are said to be α**-close with respect to** $\mathcal{Q}$ if $\sup_{q \in \mathcal{Q}} ||R(\widetilde{D}, q) - R'(\widetilde{D}', q)||_1 \leq \alpha$.

Intuitively, two synopses are α-close to one another with respect to $\mathcal{Q}$ if they are "α-indistinguishable" by $\mathcal{Q}$, i.e., no query in $\mathcal{Q}$ can be used to distinguish the two synopses by more than α. Thus, if two synopses are close to one another with respect to $\mathcal{Q}$, then we can use one synopsis to approximate the other synopsis's answers to queries in $\mathcal{Q}$. We want to construct mechanisms that, on input a database D, outputs a synopsis that is close to the synopsis (D, R_{DB}), so that we can use the synopsis to accurately answer queries in $\mathcal{Q}$ on the database D. We define the usefulness/utility of a mechanism from this perspective of closeness of synopses.

Definition 18. A mechanism San is (α, β)**-useful with respect to** $\mathcal{Q}$ **for databases of size** n if for every input database $D \in X^n$, with probability at least $1 - \beta$ (over the random coins of San), $San(D)$ outputs a synopsis $(\widetilde{D}, R)$ that is α-close to (D, R_{DB}) with respect to $\mathcal{Q}$.

We now generalize the notion of sample complexity with respect to $RS(k(\cdot))$ to classes of query functions. Intuitively, a class $\mathcal{Q}$ of queries has low sample complexity if the queries can be approximated *simultaneously* using k random samples from the input database.

Definition 19. A class $\mathcal{Q}$ of queries is said to have (k, α, β)**-sample complexity for databases of size** n **with converter** $f : \mathcal{Q} \times \mathbb{R}^m \to \mathbb{R}^m$ if for every database $D \in X^n$, if we choose k random samples from the database D *without replacement* and form a database $\widehat{D}$ consisting of the chosen samples, then with probability at least $1 - \beta$, we have $\sup_{q \in \mathcal{Q}} ||q(D) - f(q, q(\widehat{D}))||_1 \leq \alpha$.

The converter f in the above definition is used to convert the answer to a query q on the database $\widehat{D}$ to an answer to the same query q on the original database D. When $\mathcal{Q}$ is a class of queries computing averages or the fraction of rows satisfying some predicate, f would normally be the function $f(q, \vec{x}) = \vec{x}$. When $\mathcal{Q}$ is a class of queries computing sums, f would normally be the function $f(q, \vec{x}) = \frac{n}{k}\vec{x}$, since the database $\widehat{D}$ consists of only k random samples from the original database D, which has n rows.

We now show that for any class $\mathcal{Q}$ of queries with low sample complexity, we can convert any useful differentially private mechanism to a useful zero-knowledge private mechanism (with respect to $RS(k(\cdot))$).

Proposition 20. *Let $n \geq 1$, $k = k(n)$. Suppose a class $\mathcal{Q}$ of queries has (k, α_1, β_1)-sample complexity for databases of size n with a converter $f : \mathcal{Q} \times X^m \to X^m$ such that $||f(q, x) - f(q, y)||_1 \leq L||x - y||_1$ for every $x, y \in \mathbb{R}^m, q \in \mathcal{Q}$, where L is a non-negative real constant. Let $\epsilon \in (0, 1]$, and let San_{DP} be any (ϵ, δ)-differentially private mechanism that is (α_2, β_2)-useful with respect to $\mathcal{Q}$ for databases of size k.*

Then, using San_{DP}, we can construct a mechanism San_{ZK} that is $(\frac{4k}{n}\epsilon, \frac{4k}{n}\delta)$-zero-knowledge private with respect to $RS(k(\cdot))$ and is $(\alpha_1 + L\alpha_2, \beta_1 + \beta_2)$-useful with respect to $\mathcal{Q}$ for databases of size n.

Proof. Let San_{ZK} be the mechanism that, on input a database $D \in X^n$, first chooses k random samples without replacement from D, and then forms the database $\widehat{D}$ using the samples. Then, San_{ZK} runs the mechanism San_{DP} on $\widehat{D}$ to get a synopsis $(\widetilde{D}, R)$ that is α-close to $(\widehat{D}, R_{DB})$ with respect to $\mathcal{Q}$. Then, San_{ZK} outputs the synopsis $(\widetilde{D}, R')$, where R' is the algorithm that, on input the data structure $\widetilde{D}$ and a query q, first runs R on $(\widetilde{D}, q)$, then converts the answer $R(\widetilde{D}, q)$ to an answer on the original database D by computing $f(q, R(\widetilde{D}, q))$, and then outputs $f(q, R(\widetilde{D}, q))$.

By Proposition 16, San_{ZK} is $(\frac{4k}{n}\epsilon, \frac{4k}{n}\delta)$-zero-knowledge private with respect to $RS(k(\cdot))$.

We now show that San_{ZK} is $(\alpha_1 + L\alpha_2, \beta_1 + \beta_2)$-useful with respect to $\mathcal{Q}$ for databases of size n. Let $D \in X^n$. Since $\mathcal{Q}$ has (k, α_1, β_1)-sample complexity for databases of size n with the converter f, we have that with probability at least $1 - \beta_1$, $San_{ZK}(D)$ forms a database $\widehat{D} \in X^k$ such that $\sup_{q \in \mathcal{Q}} ||q(D) - f(q, q(\widehat{D}))||_1 \leq \alpha_1$. Since San_{DP} is (α_2, β_2)-useful with respect to $\mathcal{Q}$ for databases of size k, we also have that for every database $\widehat{D} \in X^k$, with probability at least $1 - \beta_2$, the mechanism $San_{DP}(\widehat{D})$ run by $San_{ZK}(D)$ outputs a synopsis $(\widetilde{D}, R)$ such that $\sup_{q \in \mathcal{Q}} ||q(\widehat{D}) - R(\widetilde{D}, q)||_1 \leq \alpha_2$.

Thus, with probability at least $1 - (\beta_1 + \beta_2)$, the synopsis $(\widetilde{D}, R')$ that $San_{ZK}(D)$ outputs satisfies $\sup_{q \in \mathcal{Q}} ||q(D) - R'(\widetilde{D}, q)||_1 = \sup_{q \in \mathcal{Q}} ||q(D) - f(q, R(\widetilde{D}, q))||_1 \leq \sup_{q \in \mathcal{Q}}(||q(D) - f(q, q(\widehat{D}))||_1 + ||f(q, q(\widehat{D})) - f(q, R(\widetilde{D}, q))||_1) \leq \alpha_1 + \sup_{q \in \mathcal{Q}}(L||q(\widehat{D}) - R(\widetilde{D}, q)||_1) \leq \alpha_1 + L\alpha_2$. Thus, with probability at least $1 - (\beta_1 + \beta_2)$, San_{ZK} outputs a synopsis $(\widetilde{D}, R')$ that is $(\alpha_1 + L\alpha_2)$-close to (D, R_{DB}) with respect to $\mathcal{Q}$, as required. □

4.1 Sample Complexity of a Class of Fraction Queries

There already exist differentially private mechanisms for classes of fraction queries (e.g., the ones in [BLR08] and [DRV10]). To use these mechanisms in Proposition 20, we will show that any class of fraction queries with low VC dimension has low sample complexity.

If the sampling in the definition of sample complexity were done *with* replacement as opposed to *without* replacement, then we could use known learning theory results to show that any class of fraction queries with low VC dimension has low sample complexity. However, the privacy guarantees of the above proposition rely on the fact that the sampling is done without replacement, since the proof uses Proposition 16, which needs this requirement. If the sampling is done with replacement, we are unable to achieve as good privacy parameters.

Our strategy is still to use known learning theory results, but we will adapt known proofs of the results as necessary so that we can use the results to show that any class of fraction queries with low VC dimension has low sample complexity, where the sampling is done without replacement.

A fraction query is a query q of the form $q(D) = \frac{|\{D_i \,:\, i \in [|D|],\ \phi(D_i) = 1\}|}{|D|}$, where D_i is the i^{th} row of the database D, and $\phi : X \to \{0, 1\}$ is some predicate. Thus, a fraction query corresponds to some predicate, and for any class $\mathcal{Q}$ of fraction queries, we can consider the class $\widehat{\mathcal{Q}}$ of predicates that correspond to the fraction queries in $\mathcal{Q}$.

We now review some terminology from learning theory. Let $\widehat{\mathcal{Q}}$ be any class of predicates, and let S be any finite subset of X. The restriction of $\widehat{\mathcal{Q}}$ to S, denoted $\widehat{\mathcal{Q}}|_S$, is the set $\{\phi|_S : S \to \{0, 1\} \mid \phi \in \widehat{\mathcal{Q}}\}$, i.e., the set of restrictions to S of all predicates in $\widehat{\mathcal{Q}}$. The growth function $\Pi_{\widehat{\mathcal{Q}}} : \mathbb{N} \to \mathbb{N}$ of $\widehat{\mathcal{Q}}$ is defined by $\Pi_{\widehat{\mathcal{Q}}}(m) = \max_{S' \subseteq X, |S'| = m} |\widehat{\mathcal{Q}}|_{S'}|$. We note that $\Pi_{\widehat{\mathcal{Q}}}(m) \leq 2^m$ for every $m \in \mathbb{N}$, since for any finite $S' \subseteq X$, there are only $2^{|S'|}$ functions from S' to $\{0, 1\}$.

We say that $\widehat{\mathcal{Q}}$ *shatters* S if $|\widehat{\mathcal{Q}}|_S| = 2^{|S|}$, i.e., for every predicate $\phi : S \to \{0, 1\}$, there exists a predicate $\phi' \in \widehat{\mathcal{Q}}$ such that $\phi'|_S = \phi$. The Vapnik-Chervonenkis dimension (VC dimension) of $\widehat{\mathcal{Q}}$ is the size of the largest finite set $S \subseteq X$ shattered by $\widehat{\mathcal{Q}}$, or ∞ if the largest doesn't exist. Equivalently, the VC dimension of $\widehat{\mathcal{Q}}$ is the largest non-negative integer m such that $\Pi_{\widehat{\mathcal{Q}}}(m) = 2^m$, or ∞ if the largest doesn't exist. We note that if $\widehat{\mathcal{Q}}$ is finite, then the VC dimension of $\widehat{\mathcal{Q}}$ is at most $\log_2 |\widehat{\mathcal{Q}}|$, since if a finite set $S \subseteq X$ is shattered by $\widehat{\mathcal{Q}}$, then $|\widehat{\mathcal{Q}}|_S| = 2^{|S|}$, so $\widehat{\mathcal{Q}}$ must contain at least $2^{|S|}$ predicates.

For convenience, when we refer to the VC dimension of a class $\mathcal{Q}$ of fraction queries, we are actually referring to the VC dimension of the class $\widehat{\mathcal{Q}}$ of predicates that corresponds to $\mathcal{Q}$. We now prove a lemma that describes how well k random samples chosen *without* replacement can

simultaneously approximate a class of fraction queries. This lemma is similar to a known result in learning theory (e.g., see Theorem 4.3 in [AB99]).

Lemma 21. *Let $\mathcal{Q}$ be any class of fraction queries, and let $\widehat{\mathcal{Q}}$ be the corresponding class of predicates. Then, for every database $D \in X^n$, $\alpha > 0$, and $k \geq 0$, we have*

$$\Pr[|q(D) - q(\widehat{D})| \geq \alpha \text{ for some } q \in \mathcal{Q}] \leq 4\Pi_{\widehat{\mathcal{Q}}}(2k)e^{-\frac{k\alpha^2}{8}},$$

where the probability is over the choice of the database $\widehat{D}$ formed by choosing k random samples without replacement *from the database D.*

Proof. Fix $D \in X^n$, $\alpha > 0$, $k \geq 0$. Let B be the event that $|q(D) - q(\widehat{D})| \geq \alpha$ for some $q \in \mathcal{Q}$, where $\widehat{D}$ is formed by choosing k random samples without replacement from the database D. Now, consider choosing another set of k random samples without replacement from the database D, and denote the samples by $\widetilde{D}$. $\widetilde{D}$ is chosen independently of $\widehat{D}$, so $\widetilde{D}$ and $\widehat{D}$ may contain overlapping samples.

Let B' be the event that $|q(\widehat{D}) - q(\widetilde{D})| \geq \frac{\alpha}{2}$ for some $q \in \mathcal{Q}$, where $\widehat{D}$ and $\widetilde{D}$ are chosen as above. Our goal is to bound $\Pr[B]$, and we do so by showing $\Pr[B] \leq 2\Pr[B']$ and bounding $\Pr[B']$ instead. We first note that if $k < \frac{4}{\alpha^2}$, then the RHS of the inequality in the lemma is ≥ 1, and so the lemma holds trivially. Thus, we can assume that $k \geq \frac{4}{\alpha^2}$.

We now show that $\Pr[B] \leq 2\Pr[B']$. To do this, we first show that $\Pr[B' \mid B] \geq \frac{1}{2}$. Suppose event B occurs so that $|q(D) - q(\widehat{D})| \geq \alpha$ for some fixed $\widehat{D}$ and $q \in \mathcal{Q}$. We will show that $\Pr[|q(\widetilde{D}) - q(D)| \leq \frac{\alpha}{2}] \geq \frac{1}{2}$, so with probability $\geq \frac{1}{2}$, the event B' also occurs, since $|q(\widetilde{D}) - q(D)| \leq \frac{\alpha}{2}$ and $|q(D) - q(\widehat{D})| \geq \alpha$ imply that $|q(\widehat{D}) - q(\widetilde{D})| \geq \frac{\alpha}{2}$. Now, by Hoeffding's inequality, we have $\Pr[|q(\widetilde{D}) - q(D)| \leq \frac{\alpha}{2}] \geq 1 - 2e^{-2k(\frac{\alpha}{2})^2}$, and the RHS is $\geq \frac{1}{2}$ if and only if $k \geq \frac{2\ln 4}{\alpha^2}$, which holds since $k \geq \frac{4}{\alpha^2}$. We have shown that $\Pr[B' \mid B] \geq \frac{1}{2}$. Thus, $\frac{\Pr[B']}{\Pr[B]} \geq \frac{\Pr[B' \text{ and } B]}{\Pr[B]} = \Pr[B' \mid B] \geq \frac{1}{2}$, so $\Pr[B] \leq 2\Pr[B']$.

We will now bound $\Pr[B']$. Consider the following process. Choose $\widehat{D}$ and $\widetilde{D}$ as before, and then perform the following swapping process. Let Y be the set of samples of D that were chosen to be in *both* $\widehat{D}$ and $\widetilde{D}$. Regarding $\widehat{D}$ and $\widetilde{D}$ as sets as samples, we arbitrarily pair (using any fixed deterministic algorithm) each sample in $\widehat{D} \setminus Y$ with a sample in $\widetilde{D} \setminus Y$ so that we have a (perfect) matching between $\widehat{D} \setminus Y$ and $\widetilde{D} \setminus Y$; we also pair each sample in $\widehat{D} \cap Y$ with the corresponding (equal) sample in $\widetilde{D} \cap Y$. Then, for each matched pair x, y, we swap x and y with probability $\frac{1}{2}$. Let $\widehat{D}'$ and $\widetilde{D}'$ denote the resulting $\widehat{D}$ and $\widetilde{D}$. It is easy to see that the sets $\widehat{D}'$ and $\widetilde{D}'$ are identically distributed to $\widehat{D}$ and $\widetilde{D}$. (The main difference between this proof and classic proofs (e.g., see [AB99]) of the corresponding learning theory result is in this swapping procedure, where we do the swapping in a particular way to ensure that $\widehat{D}'$ and $\widetilde{D}'$ are identically distributed to $\widehat{D}$ and $\widetilde{D}$ even though our sampling is done without replacement.)

Let B'' be the event that $|q(\widehat{D}') - q(\widetilde{D}')| \geq \frac{\alpha}{2}$ for some $q \in \mathcal{Q}$. Then $\Pr[B'] = \Pr[B'']$, so it suffices to bound $\Pr[B'']$. We will show that $\Pr[B''] \leq 2\Pi_Q(2k)e^{-\frac{k\alpha^2}{8}}$ by showing that $\Pr[B'' \mid \widehat{D}, \widetilde{D}] \leq 2\Pi_{\widehat{\mathcal{Q}}}(2k)e^{-\frac{k\alpha^2}{8}}$ for every fixed $\widehat{D}$ and $\widetilde{D}$ that can be sampled while generating $\widehat{D}'$ and $\widetilde{D}'$.

To this end, fix $\widehat{D}$ and $\widetilde{D}$. Let $t = |(\widehat{\mathcal{Q}}|_{\widehat{D}' \cup \widetilde{D}'})|$, where $\widehat{D}' \cup \widetilde{D}'$ is regarded as a set of elements in X. We note that $t \leq \Pi_{\widehat{\mathcal{Q}}}(2k)$, since $|\widehat{D}' \cup \widetilde{D}'| \leq 2k$. Since $|(\widehat{\mathcal{Q}}|_{\widehat{D}' \cup \widetilde{D}'})| = t$, we can choose t predicates $\phi_1, \ldots, \phi_t \in \widehat{\mathcal{Q}}$ such that for any predicate $\phi \in \widehat{\mathcal{Q}}$, there exists an $i \in \{1, \ldots, t\}$ such that $\phi|_{\widehat{D}' \cup \widetilde{D}'} = \phi_i|_{\widehat{D}' \cup \widetilde{D}'}$, i.e., $\phi(x) = \phi_i(x)$ for every $x \in \widehat{D}' \cup \widetilde{D}'$.

Let $q_1, \ldots, q_t$ be the fraction queries in $\mathcal{Q}$ that correspond to the predicates $\phi_1, \ldots, \phi_t$. Then, for any fraction query $q \in \mathcal{Q}$, there exists an $i \in \{1, \ldots, t\}$ such that $q(\widehat{D}') = q_i(\widehat{D}')$ and $q(\widetilde{D}') = q_i(\widetilde{D}')$.

Thus, B'' occurs if and only if $|q_i(\widehat{D}') - q_i(\widetilde{D}')| \geq \frac{\alpha}{2}$ for some $i \in \{1, \ldots, t\}$. Then, by the union bound, we have the following:

$$\begin{aligned}\Pr[B'' \mid \widehat{D}, \widetilde{D}] &\leq t \max_{1 \leq i \leq t} \Pr[|q_i(\widehat{D}') - q_i(\widetilde{D}')| \geq \frac{\alpha}{2} \mid \widehat{D}, \widetilde{D}] \\ &\leq \Pi_Q(2k) \max_{1 \leq i \leq t} \Pr[|q_i(\widehat{D}') - q_i(\widetilde{D}')| \geq \frac{\alpha}{2} \mid \widehat{D}, \widetilde{D}].\end{aligned}$$

Fix an $i \in \{1, \ldots, t\}$. For convenience, we order the elements in $\widehat{D}$, $\widetilde{D}$, $\widehat{D}'$, and $\widetilde{D}'$ as $\widehat{D}_1, \ldots, \widehat{D}_k$, $\widetilde{D}_1, \ldots, \widetilde{D}_k$, $\widehat{D}'_1, \ldots, \widehat{D}'_k$, and $\widetilde{D}'_1, \ldots, \widetilde{D}'_k$, respectively, so that for every $j \in \{1, \ldots, k\}$, $\widehat{D}_j$ is matched with $\widetilde{D}_j$, and $\widehat{D}'_j$ is matched with $\widetilde{D}'_j$ according to the pairing scheme described above. Now, observe that

$$\begin{aligned}\Pr[|q_i(\widehat{D}') - q_i(\widetilde{D}')| \geq \frac{\alpha}{2} \mid \widehat{D}, \widetilde{D}] &= \Pr[|\frac{1}{k}\sum_{j=1}^{k} \phi_i(\widehat{D}'_j) - \frac{1}{k}\sum_{j=1}^{k} \phi_i(\widetilde{D}'_j)| \geq \frac{\alpha}{2} \mid \widehat{D}, \widetilde{D}] \\ &= \Pr[|\frac{1}{k}\sum_{j=1}^{k} (\phi_i(\widehat{D}'_j) - \phi_i(\widetilde{D}'_j))| \geq \frac{\alpha}{2} \mid \widehat{D}, \widetilde{D}] \\ &= \Pr[|\frac{1}{k}\sum_{j=1}^{k} (|\phi_i(\widehat{D}_j) - \phi_i(\widetilde{D}_j)| \cdot z_j)| \geq \frac{\alpha}{2} \mid \widehat{D}, \widetilde{D}], \quad z_j \leftarrow \{-1, 1\} \\ &\leq 2e^{-\frac{k\alpha^2}{8}},\end{aligned}$$

where $z_j \leftarrow \{-1, 1\}$ means that z_j is sampled uniformly from $\{-1, 1\}$, and the last inequality follows from Hoeffding's inequality. Thus, $\Pr[B'' \mid \widehat{D}, \widetilde{D}] \leq 2\Pi_{\widehat{\mathcal{Q}}}(2k)e^{-\frac{k\alpha^2}{8}}$, so $\Pr[B''] \leq 2\Pi_{\widehat{\mathcal{Q}}}(2k)e^{-\frac{k\alpha^2}{8}}$. Therefore, $\Pr[B] \leq 2\Pr[B'] = 2\Pr[B''] \leq 4\Pi_{\widehat{\mathcal{Q}}}(2k)e^{-\frac{k\alpha^2}{8}}$, as required. □

The above lemma gives an upper bound on the probability that k random samples (chosen without replacement) does not simultaneously approximate a class $\mathcal{Q}$ of fraction queries well, and the upper bound involves $\Pi_{\widehat{\mathcal{Q}}}(2k)$. The following lemma gives an upper bound on $\Pi_{\widehat{\mathcal{Q}}}(2k)$ in terms of the VC dimension of $\widehat{\mathcal{Q}}$.

Lemma 22. *Let $\widehat{\mathcal{Q}}$ be any class of predicates with finite VC dimension $d \geq 1$. Then, for every integer $m \geq d$, we have $\Pi_{\widehat{\mathcal{Q}}}(m) \leq (\frac{em}{d})^d$.*

Proof. This lemma is a well-known result in learning theory and is a corollary of "Sauer's lemma", which states that for any class $\widehat{\mathcal{Q}}$ of predicates with finite VC dimension d, $\Pi_{\widehat{\mathcal{Q}}}(m) \leq \sum_{i=0}^{d} \binom{m}{i}$ for all nonnegative integers m (e.g., see [Sau72]). A proof of Sauer's lemma, as well as this lemma, can be found in [AB99]. □

We can now combine Lemmas 21 and 22 to get the following proposition.

Proposition 23. *Let $\mathcal{Q}$ be any class of fraction queries with finite VC dimension $d \geq 1$. Then, for every $n \geq 1$, $k \geq d/2$, and $\beta \in (0, 1]$, $\mathcal{Q}$ has (k, α, β)-sample complexity for databases of size n with the converter $f(q, x) = x$, where $\alpha = \frac{2\sqrt{2}}{\sqrt{k}}\sqrt{d\ln(\frac{2ek}{d}) + \ln(\frac{4}{\beta})}$.*

Also, for every $n \geq 1$ and $\alpha, \beta \in (0, 1]$, $\mathcal{Q}$ has (k, α, β)-sample complexity for databases of size n with converter $f(q, x) = x$, where k is any non-negative integer satisfying $k \geq \frac{16}{\alpha^2}(2d\ln(\frac{6}{\alpha}) + \ln(\frac{4}{\beta}))$.

Proof. Let $\widehat{\mathcal{Q}}$ be the class of predicates that corresponds to the class $\mathcal{Q}$ of fraction queries. Let $n \geq 1$, $k \geq d/2$, $\alpha > 0$, $\beta \in (0,1]$, and $D \in X^n$. By Lemma 21, we have $\Pr[|q(D) - q(\widehat{D})| \geq \alpha \text{ for some } q \in \mathcal{Q}] \leq 4\Pi_{\widehat{\mathcal{Q}}}(2k)e^{-\frac{k\alpha^2}{8}}$, where $\widehat{D}$ is a database formed by choosing k random samples without replacement from the database D. Thus, $\mathcal{Q}$ has $(k, \alpha, 4\Pi_{\widehat{\mathcal{Q}}}(2k)e^{-\frac{k\alpha^2}{8}})$-sample complexity for databases of size n with converter $f(q,x) = x$.

Rearranging the inequality $4\Pi_{\widehat{\mathcal{Q}}}(2k)e^{-\frac{k\alpha^2}{8}} \leq \beta$, we get $\alpha \geq \sqrt{\frac{8}{k}(\ln(\Pi_{\widehat{\mathcal{Q}}}(2k)) + \ln(\frac{4}{\beta}))}$. We have $\Pi_{\widehat{\mathcal{Q}}}(2k) \leq (\frac{2ek}{d})^d$ by Lemma 22, so $\alpha \geq \frac{2\sqrt{2}}{\sqrt{k}}\sqrt{d\ln(\frac{2ek}{d}) + \ln(\frac{4}{\beta})}$ implies that $4\Pi_{\widehat{\mathcal{Q}}}(2k)e^{-\frac{k\alpha^2}{8}} \leq \beta$. Thus, $\mathcal{Q}$ has (k, α', β)-sample complexity for databases of size n with converter $f(q,x) = x$, where $\alpha' = \frac{2\sqrt{2}}{\sqrt{k}}\sqrt{d\ln(\frac{2ek}{d}) + \ln(\frac{4}{\beta})}$, as required.

Now, let $\alpha \in (0,1]$ and k be any non-negative integer satisfying $k \geq \frac{16}{\alpha^2}(2d\ln(\frac{6}{\alpha}) + \ln(\frac{4}{\beta}))$. We note that $k \geq d/2$ still holds. Thus, from the argument above, to show that $\mathcal{Q}$ has (k, α, β)-sample complexity for databases of size n with converter $f(q,x) = x$, it suffices to show that $\alpha \geq \frac{2\sqrt{2}}{\sqrt{k}}\sqrt{d\ln(\frac{2ek}{d}) + \ln(\frac{4}{\beta})}$ holds. Rearranging $\alpha \geq \frac{2\sqrt{2}}{\sqrt{k}}\sqrt{d\ln(\frac{2ek}{d}) + \ln(\frac{4}{\beta})}$, we get $k \geq \frac{8}{\alpha^2}(d\ln(k) + d\ln(\frac{2e}{d}) + \ln(\frac{4}{\beta}))$.

We now use the inequality $\ln a \leq ab + \ln\frac{1}{b} - 1$, which holds for all $a, b > 0$; this can be easily shown by using the inequality $1 + x \leq e^x$ (which holds for all $x \in \mathbb{R}$), setting x to $ab - 1$, and rearranging the inequality. Applying the inequality $\ln a \leq ab + \ln\frac{1}{b} - 1$ with $a = k$ and $b = \frac{\alpha^2}{16d}$, we have $\ln k \leq \frac{\alpha^2}{16d}k + \ln(\frac{16d}{\alpha^2}) - 1 = \frac{\alpha^2}{16d}k + \ln(\frac{16d}{e\alpha^2})$.

Thus, it suffices to show that $k \geq \frac{8}{\alpha^2}(d\frac{\alpha^2}{16d}k + d\ln(\frac{16d}{e\alpha^2}) + d\ln(\frac{2e}{d}) + \ln(\frac{4}{\beta}))$. Rearranging this inequality, we get $k \geq \frac{16}{\alpha^2}(2d\ln(\frac{4\sqrt{2}}{\alpha}) + \ln(\frac{4}{\beta}))$, which holds by definition of k. □

4.2 Constructing Zero-Knowledge Private Mechanisms for a Class of Fraction Queries

Proposition 23 gives us a bound on the sample complexity of any class $\mathcal{Q}$ of fraction queries in terms of its VC dimension. Proposition 20 allows us to convert differentially private mechanism to zero-knowledge private mechanisms for any class of queries with low sample complexity. Thus, we now combine these two propositions with existing differentially private mechanisms for classes of fraction queries with low VC dimension.

The following proposition is obtained by using the differentially private mechanism in [BLR08].

Proposition 24. *Let $\mathcal{Q}$ be any class of fraction queries with finite VC dimension $d \geq 1$, and suppose the data universe X is finite. Let $n \geq 1$ and $\epsilon, \alpha, \beta \in (0,1]$. Then, for every integer $k = k(n)$ satisfying $k \geq O(\frac{(\log|X|)d\log(1/\alpha)+\log(1/\beta)}{\alpha^3\epsilon})$, there exists a mechanism San that is (α, β)-useful with respect to $\mathcal{Q}$ for databases of size n, and is $\frac{4k}{n}\epsilon$-zero-knowledge private with respect to $RS(k(\cdot))$.*

Proof. By Proposition 23, $\mathcal{Q}$ has $(k', \frac{\alpha}{2}, \frac{\beta}{2})$-sample complexity for databases of size n with the converter $f(q,x) = x$, where k' is any non-negative integer satisfying $k' \geq \frac{64}{\alpha^2}(2d\ln(\frac{12}{\alpha}) + \ln(\frac{8}{\beta}))$. We note that $|f(q,x) - f(q,y)| \leq 1 \cdot |x - y|$ for every $x, y \in \mathbb{R}, q \in \mathcal{Q}$. Let San_{DP} be the ϵ-differentially private mechanism in [BLR08] that is $(\frac{\alpha}{2}, \frac{\beta}{2})$-useful with respect to $\mathcal{Q}$ for databases of size $k'' \geq O(\frac{(\log|X|)d\log(1/\alpha)}{\alpha^3\epsilon} + \frac{\log(1/\beta)}{\alpha\epsilon})$. (This result in [BLR08] actually assumes that $X = \{0,1\}^{d'}$ for some d', but as mentioned in the paper, the result can be easily extended to any finite set X.)

Let $k \geq \max\{\frac{64}{\alpha^2}(2d\ln(\frac{12}{\alpha}) + \ln(\frac{8}{\beta})), O(\frac{(\log|X|)d\log(1/\alpha)}{\alpha^3\epsilon} + \frac{\log(1/\beta)}{\alpha\epsilon})\} = O(\frac{(\log|X|)d\log(1/\alpha)+\log(1/\beta)}{\alpha^3\epsilon})$. Then, by Proposition 20, we can use San_{DP} to construct a mechanism San_{ZK} that is $(\frac{4k}{n}\epsilon, \frac{4k}{n}\delta)$-

zero-knowledge private with respect to $RS(k(\cdot))$ and is (α, β)-useful with respect to $\mathcal{Q}$ for databases of size n. □

We now use the differentially private mechanism in [DRV10] to obtain the following proposition.

Proposition 25. *Let $\mathcal{Q}$ be any finite class of fraction queries with finite VC dimension $d \geq 1$, and suppose the data universe X is finite. Let $n \geq 1$, $\epsilon \in (0, 1]$, and $\kappa \geq 1$. Then, for every integer $k = k(n)$ satisfying $k \geq \frac{d}{2}$, there exists a mechanism San that is $(\tilde{O}(\frac{\sqrt{d+\kappa}}{\sqrt{k}} + \frac{\sqrt{\log|X|}(\log|\mathcal{Q}|)\kappa^{3/2}}{\epsilon\sqrt{k}}), e^{-\kappa})$-useful with respect to $\mathcal{Q}$ for databases of size n, and is $(\frac{4k}{n}\epsilon, \frac{4k}{n}e^{-\kappa})$-zero-knowledge private with respect to $RS(k(\cdot))$.*

Proof. Let $k \geq \frac{d}{2}$. Then, by Proposition 23, $\mathcal{Q}$ has $(k, \alpha, \frac{1}{2}e^{-\kappa})$-sample complexity for databases of size n with the converter $f(q, x) = x$, where $\alpha = \frac{2\sqrt{2}}{\sqrt{k}}\sqrt{d\ln(\frac{2ek}{d}) + \ln(8e^{\kappa})} = \tilde{O}(\frac{\sqrt{d+\kappa}}{\sqrt{k}})$. We note that $|f(q, x) - f(q, y)| \leq 1 \cdot |x - y|$ for every $x, y \in \mathbb{R}, q \in \mathcal{Q}$. Let San_{DP} be the $(\epsilon, e^{-\kappa})$-differentially private mechanism in [DRV10] that is $(\tilde{O}(\frac{\sqrt{\log|X|}(\log|\mathcal{Q}|)\kappa^{3/2}}{\epsilon\sqrt{k}}), \frac{1}{2}e^{-\kappa})$-useful with respect to $\mathcal{Q}$ for databases of size k.

Then, by Proposition 20, we can use San_{DP} to construct a mechanism San_{ZK} that is $(\frac{4k}{n}\epsilon, \frac{4k}{n}\delta)$-zero-knowledge private with respect to $RS(k(\cdot))$ and is $(\tilde{O}(\frac{\sqrt{d+\kappa}}{\sqrt{k}} + \frac{\sqrt{\log|X|}(\log|\mathcal{Q}|)\kappa^{3/2}}{\epsilon\sqrt{k}}), e^{-\kappa})$-useful with respect to $\mathcal{Q}$ for databases of size n. □

In general, Propositions 23 and 20 can be used to convert useful differentially private mechanisms to useful zero-knowledge private mechanisms for classes of fraction queries with low VC dimension.

Recall that if $\mathcal{Q}$ is a finite class of fraction queries, then the VC dimension of $\mathcal{Q}$ is at most $\log_2|\mathcal{Q}|$. Thus, we can replace the VC dimension d in Proposition 25 by $\log|\mathcal{Q}|$ (we can also do this in Proposition 24 if we assume $\mathcal{Q}$ is finite). However, it is possible that the VC dimension of a class $\mathcal{Q}$ of fraction queries is substantially smaller than $\log_2|\mathcal{Q}|$ (e.g., see [AB99]), especially if $\mathcal{Q}$ is infinite.

5 Zero-Knowledge Private Release of Graph Properties

In this section, we first generalize statistical (row) databases to graphs with personal data so that we can model a social network and privately release information that is dependent on the graph structure. We then discuss how to model privacy in a social network, and we construct a sample of zero-knowledge private mechanisms that release certain information about the graph structure of a social network.

We represent a social network using a graph whose vertices correspond to people (or other social entities) and whose edges correspond to social links between them, and a vertex can have certain personal data associated with it. There are various types of information about a social network one may want to release, such as information about the people's data, information about the structure of the social network, and/or information that is dependent on both. In general, we want to ensure privacy of each person's personal data as well as the person's links to other people (i.e., the list of people the person is linked to via edges).

To formally model privacy in social networks, let $\mathcal{G}_n$ be a class of graphs on n vertices where each vertex includes personal data. (When we refer to a graph $G \in \mathcal{G}_n$, the graph always includes the personal data of each vertex.) The graph structure is represented by an adjacency matrix, and each vertex's personal data is represented by an element in X. For the privacy of individuals, we use our zero-knowledge privacy definition with some minor modifications:

- ϵ-zero-knowledge privacy is defined as before except we change "database $D \in X^n$" to "graph $D \in \mathcal{G}_n$", and we define $(D_{-i}, \perp)$ to be the graph D except the personal data of vertex i is replaced by $\perp$ and all the edges incident to vertex i are removed (by setting the corresponding entries in the adjacency matrix to 0); thus $(D_{-i}, \perp)$ is essentially D with person i's personal data and links removed.

We now consider functions $g : \mathcal{G}_n \to \mathbb{R}^m$, and we redefine the L_1-sensitivity of g to be $\Delta(g) = \max\{||g(D') - g(D'')||_1 : D', D'' \in \mathcal{G}_n \text{ s.t. } (D'_{-i}, \perp) = (D''_{-i}, \perp) \text{ for some } i \in [n]\}$. We also redefine $RS(k(\cdot))$ so that the algorithms in $RS(k(\cdot))$ are given a graph $D \in \mathcal{G}_n$ and are allowed to choose $k(n)$ random vertices without replacement and read their personal data; however, the algorithms are not allowed to read the structure of the graph, i.e., the adjacency matrix. It is easy to verify that all our previous results still hold when we consider functions $g : \mathcal{G}_n \to \mathbb{R}^m$ on graphs and use the new definition of $\Delta(g)$ and $RS(k(\cdot))$.

Since a social network has more structure than a statistical database containing a list of values, we now consider more general models of aggregate information that allow us to release more information about social networks:

- $RSE(k(\cdot), s) = k(\cdot)$ random samples with exploration of s vertices: the class of algorithms T such that on input a graph $D \in \mathcal{G}_n$, T chooses $k(n)$ random vertices uniformly with or without replacement (or a combination of both). For each sampled vertex v, T is allowed to explore the graph locally at v until s vertices (including the sampled vertex) have been visited. The data of any visited vertex can be read. (RSE stands for "random samples with exploration".)
- $RSN(k(\cdot), d) = k(\cdot)$ random samples with neighborhood of radius d: same as $RSE(k(\cdot), s)$ except that while exploring locally, instead of exploring until s vertices have been visited, T is allowed to explore up to a distance of d from the sampled vertex. (RSN stands for "random samples with neighborhood".)

Note that these models of aggregate information include $RS(k(\cdot))$ as a special case. We can also consider variants of these models where instead of allowing the data of any visited vertex to be read, only the data of the $k(n)$ randomly chosen vertices can be read. (The data of the "explored" vertices cannot be read.)

Remark. In the above models, vertices (people) in the graph with high degree may be visited with higher probability than those with low degree. Thus, the privacy of these people may be less protected. However, this is often the case in social networks, where people with very many friends will naturally have less privacy than those with few friends.

We now show how to combine Proposition 12 (the connection between sample complexity and zero-knowledge privacy) with recent sublinear time algorithms to privately release information about the graph structure of a social network. For simplicity, we assume that the degree of every vertex is bounded by some constant $d_{\max}$ (which is often the case in a social network anyway).[3]

Let $\mathcal{G}_n$ be the set of all graphs on n vertices where every vertex has degree at most $d_{\max}$. We assume that $d_{\max}$ is publicly known. Let $M = \frac{d_{\max} n}{2}$ be an upper bound on the number of edges of a graph in $\mathcal{G}_n$. For any graph $G \in \mathcal{G}$, the (relative) distance from G to the some property Π, denoted $dist(G, \Pi)$, is the least number of edges that need to be modified (added/removed) in G in order to make it satisfy property Π, divided by M.

Theorem 26. *Let Conn, Eul, and CycF be the property of being connected, Eulerian*[4]*, and cycle-*

[3] Weaker results can still be established without this assumption.

[4] A graph G is Eulerian if there exists a path in G that traverses every edge of G exactly once.

free, respectively. Let $\bar{d}(G)$ denote the average degree of a vertex in G. Let $\epsilon, \delta > 0$, and let $K \in \mathbb{Z}^+$. Then, for the class of graphs $\mathcal{G}_n$, we have the following results:

1. *The mechanism $San(G) = dist(G, Conn) + Lap(\frac{2/n+\delta}{\epsilon})$ is $\epsilon + e^{-(K-\epsilon/\delta)}$-zero-knowledge private with respect to $RSE(k(\cdot), s)$, where $k(n) = O(\frac{K}{(\delta d_{\max})^2})$ and $s = O(\frac{1}{\delta d_{\max}})$.*

2. *The mechanism $San(G) = dist(G, Eul) + Lap(\frac{4/n+\delta}{\epsilon})$ is $\epsilon + e^{-(K-\epsilon/\delta)}$-zero-knowledge private with respect to $RSE(k(\cdot), s)$, where $k(n) = O(\frac{K}{(\delta d_{\max})^2})$ and $s = O(\frac{1}{\delta d_{\max}})$.*

3. *The mechanism $San(G) = dist(G, CycF) + Lap(\frac{2/n+\delta}{\epsilon})$ is $\epsilon + e^{-(K-\epsilon/\delta)}$-zero-knowledge private with respect to $RSE(k(\cdot), s)$, where $k(n) = O(\frac{K}{\delta^2})$ and $s = O(\frac{1}{\delta d_{\max}})$.*

4. *The mechanism $San(G) = \bar{d}(G) + Lap(\frac{2d_{\max}/n+\delta L}{\epsilon})$ is $\epsilon + e^{-(K-\epsilon/\delta)}$-zero-knowledge private with respect to $RSN(k(\cdot), 2)$, where $k(n) = O(K\sqrt{n}\log^2 n \cdot \frac{1}{\delta^{9/2}}\log(\frac{1}{\delta}))$. Here, we further assume that $\delta \in (0, \frac{1}{2})$ and every graph in $\mathcal{G}$ has no isolated vertices and the average degree of a vertex is bounded by L.*

The results of the above theorem are obtained by combining Proposition 12 (the connection between sample complexity and zero-knowledge privacy) with sublinear time algorithms from [MR09] (for results 1, 2, and 3) and [GR08] (for result 4). Intuitively, the sublinear algorithms give bounds on the sample complexity of the functions ($dist(G, Conn)$, etc.) with respect to $RSE(k(\cdot), s)$ or $RSN(k(\cdot), d)$.

Proof.

Distance approximation to connectivity: Let $San(G) = dist(G, Conn) + Lap(\lambda)$, where $\lambda = \frac{2/n+\delta}{\epsilon}$. In [MR09], Marko and Ron have given an algorithm that approximates the distance to connectivity to within an additive error δ with probability at least $\frac{2}{3}$. The algorithm does this by randomly choosing $O(\frac{1}{(\delta d_{\max})^2})$ vertices, and for each chosen vertex, exploring the graph locally from the vertex until at most $O(\frac{1}{\delta d_{\max}})$ vertices have been reached. Here is the algorithm from [MR09] (modified slightly to fit this context):

1. Uniformly and independently sample $t = \frac{32}{(\delta d_{\max})^2}$ vertices from G. Let S be the multiset of the sampled vertices.

2. For every $v \in S$, perform a BFS starting from v until $\frac{4}{\delta d_{\max}}$ vertices have been reached or v's connected component has been found. Let $\widehat{n}_v$ be the number of vertices in v's connected component in case it was found. Otherwise $\widehat{n}_v = \infty$.

3. Let $\widehat{C} = \frac{n}{t}\sum_{v \in S}(\frac{1}{\widehat{n}_v})$ and output $\frac{1}{M}(\widehat{C} - 1)$.

By running the above algorithm $O(K)$ times and outputting the median value, we can increase the success probability to $1 - e^{-K}$. Thus, $dist(G, Conn)$ has $(\delta, 1 - e^{-K})$ sample complexity with respect to $RSE(k(\cdot), s)$, where $k(n) = O(\frac{K}{(\delta d_{\max})^2})$ and $s = O(\frac{1}{\delta d_{\max}})$. By Proposition 12, San is $\ln(e^{\frac{2/n+\delta}{\lambda}} + e^{\frac{1}{\lambda}-K})$-zero-knowledge private with respect to $RSE(k(\cdot), s)$.

Now, observe that $\ln(e^{\frac{2/n+\delta}{\lambda}} + e^{\frac{1}{\lambda}-K}) \leq \ln(e^{\epsilon} + e^{\epsilon/\delta - K}) \leq \epsilon + e^{\epsilon/\delta - K} = \epsilon + e^{-(K-\epsilon/\delta)}$. Thus, we have the following result:

- The mechanism $San(G) = dist(G, Conn) + Lap(\frac{2/n+\delta}{\epsilon})$ is $\epsilon + e^{-(K-\epsilon/\delta)}$-zero-knowledge private with respect to $RSE(k(\cdot), s)$, where $k(n) = O(\frac{K}{(\delta d_{\max})^2})$ and $s = O(\frac{1}{\delta d_{\max}})$.

Distance approximation to being Eulerian: Let $San(G) = dist(G, Eul) + Lap(\lambda)$, where $\lambda = \frac{4/n+\delta}{\epsilon}$. In [MR09], Marko and Ron have given an algorithm that approximates the distance to being Eulerian to within an additive error δ with probability at least $\frac{2}{3}$. The algorithm does this by randomly choosing $O(\frac{1}{(\delta d_{\max})^2})$ vertices, and for each chosen vertex, exploring the graph locally from the vertex until at most $O(\frac{1}{\delta d_{\max}})$ vertices have been reached.

By a similar analysis as in the "distance approximation to connectivity" example, we get the following result:

- The mechanism $San(G) = dist(G, Eul) + Lap(\frac{4/n+\delta}{\epsilon})$ is $\epsilon + e^{-(K-\epsilon/\delta)}$-zero-knowledge private with respect to $RSE(k(\cdot), s)$, where $k(n) = O(\frac{K}{(\delta d_{\max})^2})$ and $s = O(\frac{1}{\delta d_{\max}})$.

Distance approximation to cycle freeness: Let $San(G) = dist(G, CycF) + Lap(\lambda)$, where $\lambda = \frac{2/n+\delta}{\epsilon}$. In [MR09], Marko and Ron have given an algorithm that approximates the distance to being cycle-free to within an additive error δ with probability at least $\frac{2}{3}$. The algorithm does this by randomly choosing $O(\frac{1}{\delta^2})$ vertices, and for each chosen vertex, exploring the graph locally from the vertex until at most $O(\frac{1}{\delta d_{\max}})$ vertices have been reached.

By a similar analysis as in the "distance approximation to connectivity" example, we get the following result:

- The mechanism $San(G) = dist(G, CycF) + Lap(\frac{2/n+\delta}{\epsilon})$ is $\epsilon + e^{-(K-\epsilon/\delta)}$-zero-knowledge private with respect to $RSE(k(\cdot), s)$, where $k(n) = O(\frac{K}{\delta^2})$ and $s = O(\frac{1}{\delta d_{\max}})$.

Approximating the average degree of a graph: Let $San(G) = \bar{d}(G) + Lap(\lambda)$, where $\lambda = \frac{2d_{\max}/n+\delta L}{\epsilon}$. In [GR08], Goldreich and Ron have shown that $\bar{d}(G)$ can be approximated by an algorithm (which needs the extra assumptions stated in the above theorem) to within a multiplicative error of $(1+\delta)$ with probability at least $\frac{2}{3}$, by randomly choosing $O(\sqrt{n}\log^2 n \cdot \frac{1}{\delta^{9/2}}\log(\frac{1}{\delta}))$ vertices, and for each chosen vertex, exploring the graph locally from the vertex up to a distance of 2. By running the approximation algorithm $O(K)$ times and outputting the median value, we can increase the success probability to $1 - 2^{-K}$.

Such an algorithm is a $(\delta L, 1 - 2^{-K})$-sampler for $\bar{d}(G)$ with respect to $RSN(k(\cdot), 2)$, where $k(n) = O(K\sqrt{n}\log^2 n \cdot \frac{1}{\delta^{9/2}}\log(\frac{1}{\delta}))$. By Proposition 12, San is $\ln(e^{\frac{2d_{\max}/n+\delta L}{\lambda}} + e^{\frac{L}{\lambda}-K})$-zero-knowledge private with respect to $RSN(k(\cdot), 2)$.

Now, observe that $\ln(e^{\frac{2d_{\max}/n+\delta L}{\lambda}} + e^{\frac{L}{\lambda}-K}) \leq \ln(e^{\epsilon} + e^{\epsilon/\delta-K}) \leq \epsilon + e^{-(K-\epsilon/\delta)}$. Thus, we have the following result:

- The mechanism $San(G) = \bar{d}(G) + Lap(\frac{2d_{\max}/n+\delta L}{\epsilon})$ is $\epsilon + e^{-(K-\epsilon/\delta)}$-zero-knowledge private with respect to $RSN(k(\cdot), 2)$, where $k(n) = O(K\sqrt{n}\log^2 n \cdot \frac{1}{\delta^{9/2}}\log(\frac{1}{\delta}))$.

□

There are already many (non-private) sublinear time algorithms for computing information about graphs whose accuracy is proved formally (e.g., see [GR08, CRT05, MR09, GR97, KKR04, GR98, PR02]) or demonstrated empirically (e.g, see [LF06, KFC+05]). We leave for future work to investigate whether these (or other) sublinear algorithms can be used to get zero-knowledge private mechanisms.

6 Acknowledgements

We thank Cynthia Dwork, Ilya Mironov, and Omer Reingold for helpful discussions, and we also thank the anonymous reviewers of the Eighth Theory of Cryptography Conference (TCC 2011) for their helpful comments.

This material is based upon work supported by the National Science Foundation under Grants 0627680 and 1012593, by the New York State Foundation for Science, Technology, and Innovation under Agreement C050061, and by the iAd Project funded by the Research Council of Norway. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the sponsors.

References

[AB99] Martin Anthony and Peter L. Bartlett, *Neural network learning: Theoretical foundations*, Cambridge University Press, 1999.

[BDK07] Lars Backstrom, Cynthia Dwork, and Jon Kleinberg, *Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography*, WWW '07: Proc. of the 16th international conference on World Wide Web, 2007, pp. 181–190.

[BLR08] Avrim Blum, Katrina Ligett, and Aaron Roth, *A learning theory approach to non-interactive database privacy*, STOC '08: Proc. of the 40th annual ACM symposium on Theory of computing, 2008, pp. 609–618.

[CKLM09] Bee-Chung Chen, Daniel Kifer, Kristen LeFevre, and Ashwin Machanavajjhala, *Privacy-preserving data publishing*, Foundations and Trends in Databases **2** (2009), no. 1-2, 1–167.

[CRT05] Bernard Chazelle, Ronitt Rubinfeld, and Luca Trevisan, *Approximating the minimum spanning tree weight in sublinear time*, SIAM J. Comput. **34** (2005), no. 6, 1370–1379.

[Dal77] Tor Dalenius, *Towards a methodology for statistical disclosure control*, Statistik Tidskrift **15** (1977), 429–444.

[DKM+06] Cynthia Dwork, Krishnaram Kenthapadi, Frank Mcsherry, Ilya Mironov, and Moni Naor, *Our data, ourselves: Privacy via distributed noise generation*, In EUROCRYPT, 2006, pp. 486–503.

[DMNS06] Cynthia Dwork, Frank Mcsherry, Kobbi Nissim, and Adam Smith, *Calibrating noise to sensitivity in private data analysis*, Proc. of the 3rd Theory of Cryptography Conference, 2006, pp. 265–284.

[DN08] Cynthia Dwork and Moni Naor, *On the difficulties of disclosure prevention in statistical databases or the case for differential privacy*, 2008.

[DRV10] Cynthia Dwork, Guy Rothblum, and Salil Vadhan, *Boosting and differential privacy*, Proc. of the 51st Annual IEEE Symposium on Foundations of Computer Science, 2010.

[Dwo06] Cynthia Dwork, *Differential privacy*, ICALP, 2006, pp. 1–12.

[Dwo09] C. Dwork, *The differential privacy frontier*, Proc. of the 6th Theory of Cryptography Conference (TCC), 2009.

[FWCY10] Benjamin C. M. Fung, Ke Wang, Rui Chen, and Philip S. Yu, *Privacy-preserving data publishing: A survey of recent developments*, ACM Comput. Surv. **42** (2010), no. 4, 1–53.

[GLP11] Johannes Gehrke, Edward Lui, and Rafael Pass, *Towards privacy for social networks: a zero-knowledge based definition of privacy*, Proceedings of the 8th conference on Theory of cryptography, TCC'11, 2011, pp. 432–449.

[GR97] Oded Goldreich and Dana Ron, *Property testing in bounded degree graphs*, Proc. of the 29th annual ACM symposium on Theory of computing, 1997, pp. 406–415.

[GR98] Oded Goldreich and Dana Ron, *A sublinear bipartiteness tester for bounded degree graphs*, Proc. of the 30th annual ACM Symposium on Theory of Computing, 1998, pp. 289–298.

[GR08] Oded Goldreich and Dana Ron, *Approximating average parameters of graphs*, Random Struct. Algorithms **32** (2008), no. 4, 473–493.

[HMJ+08] Michael Hay, Gerome Miklau, David Jensen, Don Towsley, and Philipp Weis, *Resisting structural re-identification in anonymized social networks*, Proc. VLDB Endow. **1** (2008), 102–114.

[Hoe63] Wassily Hoeffding, *Probability inequalities for sums of bounded random variables*, Journal of the American Statistical Association **58** (1963), no. 301, 13–30.

[JM09] Carter Jernigan and Behram Mistree, *Gaydar*, `http://www.telegraph.co.uk/technology/facebook/6213590/Gay-men-can-be-identified-by-their-Facebook-friends.html`, 2009.

[KFC+05] V. Krishnamurthy, M. Faloutsos, M. Chrobak, L. Lao, J-H Cui, and A. G. Percus, *Reducing large internet topologies for faster simulations*, IFIP NETWORKING, 2005.

[Kif09] Daniel Kifer, *Attacks on privacy and definetti's theorem*, SIGMOD Conference, 2009, pp. 127–138.

[KKR04] Tali Kaufman, Michael Krivelevich, and Dana Ron, *Tight bounds for testing bipartiteness in general graphs*, SIAM J. Comput. **33** (2004), no. 6, 1441–1483.

[LF06] Jure Leskovec and Christos Faloutsos, *Sampling from large graphs*, KDD '06: Proc. of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, 2006, pp. 631–636.

[MR09] Sharon Marko and Dana Ron, *Approximating the distance to properties in bounded-degree and general sparse graphs*, ACM Trans. Algorithms **5** (2009), no. 2, 1–28.

[New03] M. E. J. Newman, *Ego-centered networks and the ripple effect*, Social Networks **25** (2003), no. 1, 83 – 95.

[PR02] Michal Parnas and Dana Ron, *Testing the diameter of graphs*, Random Struct. Algorithms **20** (2002), no. 2, 165–183.

[Sau72] N. Sauer, *On the density of families of sets*, Journal of Combinatorial Theory, Series A **13** (1972), no. 1, 145 – 147.

Appendix A

Example (A more detailed explanation and analysis of the Democrats vs. Republicans example)**.** Consider a a social network of n people that are grouped into cliques of size c. (For simplicity, assume that c divides n.) In each clique, either most people are Democrats, or most people are Republicans. To model this situation, we first let $\alpha \in [0, 0.2]$. For each clique, we choose a number p in $[0, \alpha] \cup [1 - \alpha, 1]$ randomly and uniformly, and we decide that each person in the clique is a Democrat with probability p, or a Republican with probability $1 - p$. This gives us a probability distribution over databases, each with a binary attribute $X = \{0, 1\}$ and n rows, where each row states the political preference of a single person; a value of 1 represents Democrat, while a value of 0 represents Republican.

Now, let $g : X^n \to \mathbb{R}^{n/c}$ be the function that computes the proportion of Democrats in each clique. Let *San* be the mechanism that, on input a database $D \in X^n$, first computes $g(D)$ and then adds $Lap(\frac{1}{c\epsilon})$ noise to each component of $g(D)$. *San* then releases this vector of noisy proportions. The L_1-sensitivity (see [DMNS06]) $\Delta(g)$ of the function g being computed is $1/c$, since if a single person changes his or her political preference, the value of g changes only by $1/c$ in one of the components (cliques). Recall from [DMNS06] that a mechanism that computes a function $h(D)$ and then adds $Lap(\frac{\Delta(h)}{\epsilon})$ noise to each component of $h(D)$ is ϵ-differentially private. Thus, *San* is ϵ-differentially private, so for small ϵ, one may think that it is safe to release such information without violating the privacy of any particular person. That is, the released data should not allow us to guess correctly with probability significantly greater than $1/2$ whether a particular person is a Democrat or a Republican. However, this is not the case.

To see this, suppose we know which clique some person i is in. We look at the data released by *San* to obtain the noisy proportion $\hat{p}$ for the clique person i is in. If $\hat{p} \geq 0.5$, we guess that person i's clique mostly consists of Democrats, so we guess that person i is a Democrat; otherwise, we guess that person i's clique mostly consists of Republicans, so we guess that person i is a Republican. Since *San* adds $Lap(\frac{1}{c\epsilon})$ noise to the true proportion p of person i's clique, we have $\Pr[\hat{p} - p \geq \frac{1}{2} - \alpha] = \Pr[p - \hat{p} \geq \frac{1}{2} - \alpha] = F(-(\frac{1}{2} - \alpha)) = \frac{1}{2}e^{-(\frac{1}{2}-\alpha)c\epsilon}$, where $F(x) = \frac{1}{2}e^{xc\epsilon}$ is the cumulative distribution function of the Laplace distribution $Lap(\frac{1}{c\epsilon})$ for $x < 0$.

We note that if $p \in [0, \alpha]$, then $\hat{p} - p < \frac{1}{2} - \alpha$ implies that our guess for person i's clique is correct, so this occurs with probability at least $1 - \frac{1}{2}e^{-(\frac{1}{2}-\alpha)c\epsilon}$. Similarly, if $p \in [1 - \alpha, 1]$, then $p - \hat{p} < \frac{1}{2} - \alpha$ implies that our guess for person i's clique is correct, so this occurs with probability at least $1 - \frac{1}{2}e^{-(\frac{1}{2}-\alpha)c\epsilon}$. In both cases, our guess for person i's clique is correct with probability at least $1 - \frac{1}{2}e^{-(\frac{1}{2}-\alpha)c\epsilon}$. Therefore, our guess for person i herself is correct with probability at least $(1 - \frac{1}{2}e^{-(\frac{1}{2}-\alpha)c\epsilon})(1 - \alpha)$.

With $\epsilon = 0.1$, $\alpha = 0.2$, and $c = 200$, our guess for person i is correct with probability at least $(1 - \frac{1}{2}e^{-(\frac{1}{2}-\alpha)c\epsilon})(1 - \alpha) \approx 0.799$. This is significantly higher than $0.5 \cdot e^{\epsilon} = 0.5 \cdot e^{0.1} \approx 0.553$, which one might think is supposed to be an upper bound on the probability that our guess is correct, since *San* satisfies ϵ-differential privacy with $\epsilon = 0.1$ (see the appendix in [DMNS06]; the 0.5 comes from guessing randomly).

With $\epsilon = 0.01$, $\alpha = 0.2$, and $c = 200$, our guess for person i is correct with probability at least $(1 - \frac{1}{2}e^{-(\frac{1}{2}-\alpha)c\epsilon})(1 - \alpha) \approx 0.580$. This is still a lot higher than $0.5 \cdot e^{\epsilon} = 0.5 \cdot e^{0.01} \approx 0.505$.