When Random Sampling Preserves Privacy*

Kamalika Chaudhuri[1] and Nina Mishra[2]

[1] Computer Science Department, UC Berkeley, Berkeley, CA 94720
kamalika@cs.berkeley.edu
[2] Computer Science Department, University of Virginia, Charlottesville, VA 22904
nmishra@cs.virginia.edu

Abstract. Many organizations such as the U.S. Census publicly release samples of data that they collect about private citizens. These datasets are first anonymized using various techniques and then a small sample is released so as to enable "do-it-yourself" calculations. This paper investigates the privacy of the second step of this process: sampling. We observe that rare values – values that occur with low frequency in the table – can be problematic from a privacy perspective. To our knowledge, this is the first work that quantitatively examines the relationship between the number of rare values in a table and the privacy in a released random sample. If we require ϵ-privacy (where the larger ϵ is, the worse the privacy guarantee) with probability at least $1 - \delta$, we say that a value is rare if it occurs in at most $\tilde{O}(\frac{1}{\epsilon})$ rows of the table (ignoring log factors). If there are no rare values, then we establish a direct connection between sample size that is safe to release and privacy. Specifically, if we select each row of the table with probability at most ϵ then the sample is $O(\epsilon)$-private with high probability. In the case that there are t rare values, then the sample is $\tilde{O}(\epsilon\delta/t)$-private with probability at least $1 - \delta$.

1 Introduction

Private data is collected by numerous organizations for a wide variety of purposes including reporting, data mining, scientific discoveries, etc. In some circumstances, this data is in turn released in sanitized form for public consumption. The purpose of releasing such sanitized data is to enable others to discover large-scale statistical patterns, e.g., to learn averages, variances, clusters, decision trees, while hiding small-scale information, e.g., a particular individual's salary. The question we ask is: To what extent do these sanitized datasets preserve people's privacy?

While there are numerous examples of sanitization procedures, we investigate one commonly used technique: random sampling. Some organizations routinely collect data, anonymize it, and then release a sample so that others may use the data for data mining purposes. Since samples are known to preserve statistical characteristics of the data, samples can be a useful means for studying and understanding the underlying population.

* Research supported in part by NSF EIA-0137761.

C. Dwork (Ed.): CRYPTO 2006, LNCS 4117, pp. 198–213, 2006.

1.1 Motivating Examples

There are many examples of released samples of private data. We describe two, one from the U.S. Census and one from the Social Security Administration.

The U.S. Census Bureau releases a Public Use Microdata Sample (PUMS) [2]. This dataset contains private information in occupied housing units such as age, weight, income, and race. The Census gathers this data once every 10 years, anonymizes it and then releases a 1% or 5% sample. The purpose of releasing this microdata is to allow "do-it-yourself" calculations. Our work is motivated by such releases: Can we simply select each individual with probability 0.05 to be included in the sample? What role do rare values play in deciding what to release? What size sample is safe to release?

The Social Security Administration (SSA) also releases microdata, specifically Benefits and Earnings files [11]. Old-Age, Survivors, and Disability Insurance (OASDI) is a government-sponsored insurance program that individuals contribute to throughout their working careers. Benefits are paid to insured workers and family members when they retire or become disabled. This dataset contains annual earnings information for approximately 47 million individuals who receive OASDI benefits each month. Personal identifying information and distinguishing characteristics are removed or modified to prevent identification. Records are randomly permuted. The SSA then releases a 1% sample of this data.

Our work is a first attempt at formally understanding the privacy guarantees of just one step of the sanitization process: random sampling. In practice, these organizations employ multi-step anonymization processes prior to sampling that this paper does not analyze.

1.2 The Model

We consider the following simplified setting. The sanitizer starts with a table T consisting of k distinct private values. The k values can be anything, e.g., Boolean data over $\log k$ attributes, k real numbers, etc. The sanitizer then goes through each row of the table and includes it in the sample with probability p and does nothing with probability $1-p$. The sample is then randomly permuted and released. We then ask the question: for what p can we guarantee privacy? In order to understand this question, we next define privacy.

1.3 Privacy

The privacy definition that we use is motivated by [5]. The authors capture the interaction between a sanitizer and a hypothetical attacker via a transcript. In our case, the transcript is a random sample S of the data. Intuitively, for any pair of tables T and T' that differ in only one position, privacy is preserved if a hypothetical attacker upon seeing the transcript S is unable to distinguish between the case when the actual table is T or T'. In other words, an attacker knowing all but one person i's private information does not gain much information about i upon seeing the sample.

We consider two definitions of privacy, one where a hypothetical attacker tries to distinguish between two tables that differ in one row, and the other, where a hypothetical attacker tries to distinguish between two tables that differ in c rows. We say that a sanitization scheme is (c, ϵ, δ)-private if for every table T, with probability at least $1 - \delta$, the scheme produces a sample S such that for any set of c rows in the table, P, and for any two sets of c assignments V and V', $\frac{\Pr(S|T_{\{P \rightarrow V\}})}{\Pr(S|T_{\{P \rightarrow V'\}})} \leq 1 + \epsilon$. The privacy definition is discussed in Section 3.

1.4 Discussion

Our results do not apply in the case that the data is a collection of distinct points, say in R^d. The reason is that if every point is different from every other point in the table then a sample of size even one violates the privacy of that individual. As a simple example, suppose the table consists of five private values $\langle 1, 2, 3, 4, 5 \rangle$ and we release the sample point 2. Then the attacker can now easily tell that the data came from the actual table versus $\langle 1, 3, 3, 4, 5 \rangle$. This violates privacy since the attacker can now distinguish between two tables that differ in one row.

Problems arise even if a value is not unique, but occurs a few times in the table. We call such a value a rare value. Observing a rare value is problematic because a rare value can be assumed by only a small group of individuals, and then observing such a value can potentially increase the hypothetical attacker's confidence about the values assumed by this small select group of people. For example, consider a table in which two individuals can have a salary of one billion dollars, and an attacker knows the salary of one of them and not the other. If we release a sample in which a row with a salary of one billion dollars appears, then the confidence of the attacker about the second individual's salary increases. This is because such a sample is much more likely to have come from a table in which two people have a salary of a billion dollars than from a table in which one person's salary is a billion dollars.

Unique values are known to be problematic in the literature. Indeed, the phrase "population unique" is used to describe those individuals that are unlike anyone else in the population, e.g., 13-year-old college graduate. Population uniques are often first removed prior to data sanitization. To the best of our knowledge, we have not seen work that quantifiably links rare values to privacy. In this paper, we find such a link. If we desire ϵ privacy with probability at least $1 - \delta$, we define a *rare value* to be one that occurs less than $O(\frac{1}{\epsilon} \log(\frac{2k}{\delta}))$ times in the table. If i rows with a certain rare value v appear in the sample, it can lead to an $O(i\epsilon / \log(2k/\delta))$ breach of privacy.

One way to deal with rare values is to suppress such rows from the table. Indeed, in practice, organizations remove population uniques. We do not consider such sanitization algorithms because then the decisions made by the sanitizer cannot be mimicked or simulated by the attacker – and as a result, information may be leaked[1]. This may seem unintuitive at first – how can private information

[1] The notion of simulatability is already known to be important in cryptography [8] and also in privacy research [9,3].

that we do not even release breach privacy? An example illustrates the point. Suppose the sanitizer decides to suppress all values that occur < 100 times, and rows 1 to 100 of a table take the value 0, and no other rows take the value 0. Let $p > 1/100$. Now suppose that an attacker knows the value of the first 99 rows and is trying to decide what the value of the 100th row is. In this case, not seeing any row with value 0 in the sample violates the privacy of this 100th row.

Another reason why removing rare values is problematic is that just the size of the sample can leak information. In the case where the table contains n rows and 1/10th of the rows contain rare values, then the expected sample size is $9np/10$ instead of np. Alternatively, if the sanitization algorithm is to draw a sample and remove "sample uniques" (those individuals that are one-of-a-kind in a sample), then if every entry in the sample is unique, then nothing may be released. Thus just the size of the sample can leak information.

Because unique and rare values can lead to privacy breaches, we assume that k the number of distinct values is much smaller than n the number of individuals in the dataset. In practice, this is not true because each individual in the table has a uniquely identifying key. The assumption that $k << n$ implies that identifying information has been removed. This is admittedly a large assumption since it is an open question what information is "identifying" (see, for example, [12]). But we make this assumption so that we can focus our attention on understanding the privacy consequences of sampling.

One limitation of sampling is the probability that the sanitization algorithm fails to produce an output that preserves privacy is not negligble in n. Ideally, we would like to say that the sanitization algorithm fails with very low probability, e.g., $1/(2^n)$. With random sampling, we cannot guarantee a failure probability better than $1/n$. To see why, suppose the table has a unique value. If we sample each row of the table with probability $p = 1/n$, then the probability we pick this value is $1/n$. But once a unique value appears in the sample then we have completely breached this individual's privacy. So the probability we fail is $1/n$.

Even if there are no rare values at all, releasing a random sample of the data cannot preserve privacy with probability 1. As an example, consider a table that has $n/2$ rows with value 0 and $n/2$ rows with value 1, and a sample S from this table of size $n/2$ consisting of all 0s. The attacker knows the value of all rows in the table except for one row - that is, she is trying to decide whether the sample S came from a table with $n/2$ 0s and $n/2$ 1s or from a table with $n/2 + 1$ 0s and $n/2 - 1$ 1s. The latter event is about $n/2$ times more likely than the former. The attacker will therefore conclude that the value of the missing row is 0, and this will lead to a breach in privacy. It turns out that when we do sampling, we cannot avoid these situations entirely except to upper bound the relatively small probability that such unlikely samples occur. We also note that the probability of failure due to the occurrence of such unlikely samples is quite small compared to the probability of failure to preserve privacy because of the occurrence of one or more rows with rare values in the sample.

1.5 Contributions

Privacy. For the case where an attacker tries to distinguish between two tables that differ in one row, i.e., $(1, \epsilon, \delta)$-privacy, we define a rare value as one that occurs in at most $\frac{\log(\frac{2k}{\delta})}{\epsilon}$ rows of the table, where k is the number of distinct values in the table. We show that if there are no rare values, then a sampling frequency of at most ϵ preserves privacy. In the case where there are at most t rare values, we show that a sampling frequency of at most $\tilde{O}(\frac{\epsilon\delta}{t})$ preserves privacy. Observe that the higher the number of rare values, the lower the sampling frequency, as one would expect. We also demonstrate that the upper bound on the sampling frequency is tight up to log factors.

Furthermore, we consider the case where a hypothetical attacker already knows $n - c$ rows of the table and the goal is (c, ϵ, δ)-privacy. Now a rare value is one that occurs in at most $\frac{\log(\frac{2k}{\delta})}{\epsilon} + c$ rows of the table. We prove that when there are no rare values, a sampling frequency of at most ϵ still preserves $(c, O(c\epsilon), \delta)$-privacy. When there are at most t rare values, we show that a sampling frequency of $p < \tilde{O}(\frac{\epsilon\delta}{t})$ is $(c, O(c\epsilon), \delta)$-private.

The proof technique is similar in both cases. We partition the values in the table T into rare, infrequent and common depending on how often they occur. We then define a good sample to be one with no rare values, with infrequent values that do not occur very frequently, and common values that occur close to expectation. We prove that if we have a good sample, we have privacy. Then we prove that with high probability, random sampling produces a good sample.

Utility. For someone who is interested in discovering patterns in the released data, it is natural to ask whether sampling preserves patterns. Samples are in fact known to approximately preserve statistics about the actual table. But note again that, in practice, sampling is used in concert with other anonymization techniques. We are not making any claims about the utility of those anonymization procedures – we only discuss the utility of sampling.

When we release a random sample of the data, we are essentially releasing an estimate of the *histogram* – the frequency of each value v. Random sampling can estimate the frequency of each value v with an additive noise of $\tilde{O}(\sqrt{\frac{t}{n\epsilon\delta}})$ when there are t rare values, and we want $(1, \epsilon, \delta)$-privacy.

Note that in contrast, [5] can release the histogram of a table by adding a tiny $O(\frac{2}{n\epsilon})$ additive noise to the frequency of each possible value, a significantly smaller additive noise. We also note that unlike sampling, the error in [5] is independent of the number of rare values. Also the more privacy that is required, the better [5] does compared to sampling.

If U is the total universe of values a row in the table can take, we note that [5] needs to release $|U|$ numbers in order to release a privacy-preserving histogram. We in contrast, need to release only k numbers. When the size of U is much larger than k, releasing a random sample is a more compact way of releasing the histogram.

2 Related Work

We partition related work according to what the sanitizer does with the private data. In the input perturbation family of methods, the private data is perturbed and published as a one-time operation. The perturbed dataset must withstand an unlimited number of queries. In the output perturbation family of methods, the sanitizer receives queries about the private dataset from an attacker. The sanitizer then outputs either the true answer, a perturbed answer, or refuses to answer altogether.

2.1 Input Perturbation

While we assume that the dataset is not a collection of distinct points in R^d, another approach is to redefine privacy with respect to this higher dimensional space. Such a compelling definition is given in [3] where a point is kept private if it "blends with the crowd". That paper offers simulatable methods for perturbing the input so that privacy is preserved. Several utility results are given including learning mixtures of Gaussians and k–Center clustering.

An alternate input perturbation technique was suggested in [7]. That paper describes a method for modifying private data (adding and deleting purchase behavior) so as to enable the discovery of frequent itemsets while maintaining privacy. The privacy guarantees given in that paper are quite strong. But the notion of utility is strongly tied to frequent itemsets.

An input perturbation technique based on pseudorandom sketches was given in [10]. The idea is that each individual takes their own data over d bits, represents it as a vector of length 2^d with a 1 in the single position corresponding to their private value and a 0 everywhere else. Each bit of this vector of length 2^d is then flipped with probability p. This perturbed vector is then replaced with a slightly biased coin that forms a seed s to a pseudorandom function. The authors show that privacy is preserved in a strong sense, i.e., for all individuals x_i and for all values v, v', $\Pr(s|x_i = v) \approx \Pr(s|x_i = v')$. Furthermore, various utility results are given including estimating the fraction of individuals that satisfy any conjunction of attributes, estimating the fraction of individuals that have private values $\leq x$, etc.

2.2 Output Perturbation

A different method for preserving privacy is output perturbation [4,6,1,5,9]. One specific output perturbation result that is very relevant to this paper is due to Dwork et al [5]. The authors introduce the notion of the *sensitivity* of a function which is how much the function f can change when one row of the data changes. Privacy is then shown to be preserved if the sanitizer answers each query with additive Laplacian noise that is proportional to the function's sensitivity. Specifically, the sanitizer returns the true answer plus Lap($\frac{sen(f)}{\epsilon}$) where Lap(λ) is the Laplace distribution with density proportional to $p(y) \propto e^{-\frac{|y|}{\lambda}}$. The more sensitive the query, the more noise is added. The sensitivity of a sequence of queries is the extent to which the sequence can change when one row of the table changes. For example, the sensitivity of a histogram is 2 since changing one row of the table at most removes a value from one bucket and adds it to another.

3 Preliminaries

We use the term *table* to mean the original unperturbed data and denote it by T. Each entry of the table is assumed to be a tuple of the form (i, j) where i is some unique identifier, e.g., SSN, name, and j is an integer that represents the individual's private data, e.g., if the data is in binary form, then one can view j as the integer representation of the binary data.

We assume that the table has n entries, where each entry can take an integer value. We assume that the total number of distinct values taken by the rows of the table is k.

We use the term *sample* or *sanitized table* to denote the result of the sanitization process that the attacker observes and we denote it by S. Note that S is a randomized object, whereas T is a deterministic input supplied to the sanitizer.

Given a table T, the goal of the sanitizer is to release a sample S of T where the sample does not give the attacker any additional information about any row of the table beyond what the attacker already knows from looking at the rest of the table.

3.1 Privacy Definition

Our definition of privacy is closely related to $(1, \epsilon)$-privacy proposed by [5] (where it was called ϵ-indistinguishability).

Definition 1. *A sanitization mechanism is* $(1, \epsilon)$-private *if for every pair of tables* T *and* T' *that differ in one row*

$$\frac{\Pr[S|T]}{\Pr[S|T']} \leq 1 + \epsilon$$

Here $\Pr[S|T]$ denotes the probability that the sanitization mechanism outputs S given as input the table T and it is taken over the random choices made by the sanitizer. This definition states that the posterior probability that the sample S came from table T is almost the same as the probability it came from table T'; therefore observing S does not enable the attacker to distinguish between these two tables reliably.

As mentioned in Section 1.4, we cannot ensure privacy with probability 1 as the table may have rare values or we may simply draw an unrepresentative sample. Consequently, we allow our sanitizer a δ probability of failure.

Definition 2. *A sanitization mechanism is* $(1, \epsilon, \delta)$-private *if, for every table* T*, with probability at least* $1 - \delta$*, the mechanism produces* S *such that for all values* v *and* v'*:*

$$\frac{\Pr[S|T_{\{i\rightarrow v\}}]}{\Pr[S|T_{\{i\rightarrow v'\}}]} \leq 1+\epsilon$$

where the probability is taken over the random choices made by the sanitizer.

This definition states that regardless of the table T, with high probability, the sample S produced does not significantly help the attacker distinguish between any two values v and v' for the ith individual in the table. While this quantifies over all possible values, it includes as a special case the ith individual's actual value and any other value.

Sometimes, there may be correlations between the values of a small number of rows in a database and these correlations may be known to the attacker. For example, the HIV status of a husband and wife are probably the same. This can be thought of more generally as follows. Suppose the table is partitioned into sets of rows $\{P_i\}$ such that if the attacker knows the value of one row in a partition P_j, she knows the value of every row in the partition. In such a situation, we might want to consider an attacker who knows the value of all rows in the table except for the rows in one partition, and examine what this attacker can learn by looking at the sanitized data. This motivates the notion of (c, ϵ)-privacy proposed by [5]. (c, ϵ)-privacy ensures that the probability that the sanitized data came from two tables T and T' that differ in at most c rows is almost the same.

Typically, more noise is needed to achieve (c, ϵ)-privacy than is needed to achieve $(1, \epsilon)$-privacy. This sounds counterintuitive at first; how could it be harder to guarantee privacy for an attacker who knows the value of only $n - c$ rows of a table than it is to guarantee privacy for an attacker who knows the value of $n - 1$ rows? This happens because we say that a violation of privacy occurs when there is a deviation from what the attacker already knows. To ensure that there is no deviation from the attacker's knowledge, we need to hide more from an attacker who knows less than from an adversary who knows more.

We can think of an analogous notion of (c, ϵ, δ)-privacy as well. The definition is identical to $(1, \epsilon, \delta)$-privacy except for any set of c rows in the table T, P_i, and for any pair of states V and V' the sample does not substantially help the attacker distinguish between $T_{\{P_i \rightarrow V\}}$ and $T_{\{P_i \rightarrow V'\}}$.

It is shown in [5] that a sanitization mechanism that is $(1, \frac{\epsilon}{c})$-private is also (c, ϵ)-private. This argument can be extended to show that a mechanism which is $(1, \frac{\epsilon}{c}, \frac{\delta}{c})$-private is also (c, ϵ, δ)-private.

3.2 Some Notation

We use the following notation for the rest of the paper. Let n be the total number of items in the table, and let $n_1, n_2, \ldots, n_k$ denote the number of items in the table with value $1, 2, \ldots, k$ respectively. Let s denote the size of the sample, and $s_1, s_2, \ldots s_k$ denote the number of items in the sample with value $1, 2, \ldots, k$.

Let $V = \{v_1, v_2, \ldots, v_c\}$ be a sequence of c values. We say that a sequence of c rows has *state* V if row i in the sequence has value v_i.

We use the notation $T_{\{i \to v\}}$ to denote a table T in which row i is set to have a value v, and $T \setminus \{i\}$ to denote the set of all rows in table T except row i. Similarly, for a set of rows P_i, we use the notation $T_{\{P_i \to V\}}$ to denote a table T in which the set of rows P_i have state V, and $T \setminus \{P_i\}$ to denote the set of all rows in table T except the rows in set P_i.

4 $(1, \epsilon, \delta)$-Privacy

In this section, we show what sampling probability is $(1, \epsilon, \delta)$-private. Given ϵ, δ, a table T and k, the number of distinct values in the table, we provide p, a sampling frequency that is $(1, \epsilon, \delta)$-private.

Our guarantees can be summarized as follows.

Theorem 3. *Given a table T, let $\alpha = \frac{\delta}{2}$, k be the number of distinct values in T and t be the total number of values in T that occur less than $\frac{2\log(\frac{k}{\alpha})}{\epsilon}$ times. Also let $\epsilon' = \max(2(p+\epsilon), 6p)$ and $p + \epsilon < \frac{1}{2}$.*

Then, a sample S of T drawn with frequency $p \leq \frac{\epsilon \log(\frac{1}{1-\alpha})}{4t\log(\frac{k}{\alpha})}$ is $(1, \epsilon', \delta)$-private when $t > 0$. When $t = 0$, a sample S of T drawn with frequency $p \leq \epsilon$ is $(1, \epsilon', \delta)$-private.

We need the assumption $p + \epsilon < \frac{1}{2}$ to make sure p is bounded away from 1 by a constant. (Any other constant than $\frac{1}{2}$ would do, but would change the constants in Theorem 3.) We want this condition because if p is too close to 1, all rows containing a certain value may appear in the sample, leading to a serious breach of privacy.

Note that for certain tables such as those consisting only of unique values, the upper bound on p according to our theorem is less than $1/n$. This should be interpreted as the fact that we cannot guarantee $(1, \epsilon, \delta)$-privacy for a sample even of size 1.

The theorem shows that for a given table T and a given failure probability δ, the lower the value of p, the better the privacy guarantee. Since $\log(\frac{1}{1-\alpha}) \approxeq \alpha$, p varies linearly as δ. This means that δ, the failure probability, has to be at least $\frac{1}{n}$ to ensure we draw a random sample even of size 1. This is expected, because in a table with a unique value v, the probability that any random sample selects this value is at least $\frac{1}{n}$. We see in Observation 6 that this dependence of p on δ is almost tight except for the factor of $\log(\frac{k}{\alpha})$.

Before we prove Theorem 3, we provide some intuition. For our proofs, we divide the set of values in the table T into three categories – rare values, infrequent values and common values.

Definition 4. *A value is said to be a* common value *if it occurs in at least $\frac{12\log(\frac{k}{\alpha})}{p}$ rows of the table, where p is the sampling frequency. A value v is called a* rare value *if it occurs in at most $\frac{2\log(\frac{k}{\alpha})}{\epsilon}$ rows of the table. A value that is neither rare nor common is called an* infrequent value.

A common value v has the property that the expected number of such values in the sample S is at least $\Omega(\log(\frac{k}{\alpha}))$, and therefore the number of such values in the sample is tightly concentrated around its mean.

If a value v is not a common value, we can only show using Chernoff Bounds that the number of occurrences of v in T is away from its expected value by at most $O(\log(\frac{k}{\alpha}))$. If about $\log(\frac{k}{\alpha})$ rows with a rare value v occur in the released sample, the posterior probability $\Pr[S|T_{\{i \to v\}}]$ can increase by more than a $(1+\epsilon)$ fraction. To deal with this, we hide all such rows. This is achieved by making p less than the inverse of the total number of such rare values.

A value that is neither common nor rare is called an *infrequent value.* Such a value may appear in a sample S, but the number of such values cannot be guaranteed to be tightly concentrated around its expectation. However, releasing about $O(\log(\frac{k}{\alpha}))$ rows with such a value does not lead to an ϵ breach in privacy.

As we showed earlier, releasing any sample drawn from a table does not ensure $(1, \epsilon, \delta)$-privacy. We show that privacy is preserved when we draw a sample with certain properties, and such a sample occurs with high probability. A sample possessing these properties is called a *good sample.*

Definition 5. *A* good sample *is one that has the following properties: (1) A rare value v does not occur. (2) An infrequent value v occurs in at most $n_v p + 2\log(\frac{k}{\alpha})$ rows. (3) A common value v occurs in at most $n_v p + \sqrt{3 n_v p \log(\frac{k}{\alpha})}$ rows.*

In Lemma 1, we show that releasing a good sample preserves privacy. In Lemma 2, we show that a good sample occurs with high probability. Combining Lemmas 1 and 2, we get a proof of Theorem 3.

Lemma 1. *Let S be a good sample drawn from table T. Then for any row i and any pair of values v and v',*

$$\frac{\Pr[S|T_{\{i \to v\}}]}{\Pr[S|T_{\{i \to v'\}}]} \leq 1 + \epsilon'$$

where $\epsilon' = \max(2(p+\epsilon), 6p)$ for $p + \epsilon < \frac{1}{2}$.

Proof. For any value u, let n_u^1 be the number of rows in table $T \setminus \{i\}$ with value u, and let s_u be as usual the number of rows with value u in the sample S. Then, $T_{\{i \to v\}}$ has $n_v^1 + 1$ rows with value v and $n_{v'}^1$ rows with value v'.

We now show that since S is a good sample, $s_v < n_v^1 + 1$. If row i of T takes any other value than v, this holds trivially; otherwise, we claim that at most n_v^1 rows of T that take value v appear in S. If v is a rare value, there are no rows with value v in a good sample. If v is an infrequent value, the maximum number of rows with value v in the sample is at most $(n_v^1+1)p+2\log(\frac{k}{\alpha})$ which is at most $n_v^1(p+\epsilon)+p < n_v^1$ for $p+\epsilon < 1/2$. If v is a common value, the maximum number of rows with value v in the good sample S is at most $(n_v^1+1)p+\sqrt{3(n_v^1+1)p\log(\frac{k}{\alpha})}$, which is at most $(n_v^1+1)p\left(1+\sqrt{\frac{3\log(\frac{k}{\alpha})}{(n_v^1+1)p}}\right) \leq \frac{3}{2}(n_v^1+1)p < n_v^1+1$ when $p < \frac{1}{2}$.

Therefore,

$$\frac{\Pr[S|T_{\{i\to v\}}]}{\Pr[S|T_{\{i\to v'\}}]} = \frac{\binom{n_v^1+1}{s_v}\binom{n_{v'}^1}{s_{v'}}}{\binom{n_v^1}{s_v}\binom{n_{v'}^1+1}{s_{v'}}} = \frac{1-\frac{s_{v'}}{n_{v'}^1+1}}{1-\frac{s_v}{n_v^1+1}}$$

For a rare value v, $s_v = 0$. Therefore,

$$\frac{1-\frac{s_{v'}}{n_{v'}^1+1}}{1-\frac{s_v}{n_v^1+1}} = 1 - \frac{s_{v'}}{n_{v'}^1+1} \leq 1$$

For an infrequent value v, since there are at most n_v^1+1 rows with value v in the table T, $s_v \leq (n_v^1+1)p + 2\log(\frac{k}{\alpha})$ and $n_v^1 + 1 \geq \frac{2\log(\frac{k}{\alpha})}{\epsilon}$. This implies that, $\frac{s_v}{n_v^1+1} \leq p+\epsilon$ and assuming $p+\epsilon < 1/2$,

$$\frac{1-\frac{s_{v'}}{n_{v'}^1+1}}{1-\frac{s_v}{n_v^1+1}} \leq 1 + \frac{p+\epsilon}{1-(p+\epsilon)} \leq 1 + 2(p+\epsilon)$$

For a common value v, as there are either n_v^1 or n_v^1+1 rows with value v in the table, s_v is at most $(n_v^1+1)p + \sqrt{3(n_v^1+1)p\log(\frac{k}{\alpha})}$, and $(n_v^1+1)p \geq 12\log(\frac{k}{\alpha})$. Therefore $\frac{s_v}{n_v^1+1} \leq \frac{3}{2}p$ which implies that

$$\frac{1-\frac{s_{v'}}{n_{v'}^1+1}}{1-\frac{s_v}{n_v^1+1}} \leq 1 + \frac{\frac{3}{2}p}{1-\frac{3}{2}p} \leq 1 + 6p$$

for $p < 1/2$. □

We now state a condition on p that ensures S is a good sample with high probability.

Lemma 2. *If the sampling frequency* $p < \frac{\epsilon\log\left(\frac{1}{1-\alpha}\right)}{4t\log(\frac{k}{\alpha})}$, *the probability that a good sample is drawn is at least* $(1-\alpha)^2$.

Proof. We observe that given a fixed table T,the number of rows in S with value u is independent of the number of rows in S with any other value u'.

For a common value v, the probability that S has more than $n_v p + \sqrt{3 n_v p \log(\frac{k}{\alpha})}$ rows with value v is at most

$$e^{-3\log(\frac{k}{\alpha})/3} = \frac{\alpha}{k}$$

using Chernoff Bounds.

For an infrequent value v, the probability that S has more than $n_v p + 2\log(\frac{k}{\alpha})$ rows with value v is at most

$$e^{-2\log(\frac{k}{\alpha})/2} = \frac{\alpha}{k}$$

Since there are k values altogether, the total probability that the sample has the requisite number of common and infrequent values is at least $1 - k \cdot \frac{\alpha}{k} = 1 - \alpha$.

The probability that a rare value v does not occur in S is $(1-p)^{n_v}$. If there are at most t rare values, then the probability that none of these values occur in S is at least $(1-p)^{2t\log(\frac{k}{\alpha})/\epsilon} \geq e^{-4pt\log(\frac{k}{\alpha})/\epsilon}$. For $p < \epsilon \frac{\log(\frac{1}{1-\alpha})}{4t\log(\frac{k}{\alpha})}$, this probability is at least $1 - \alpha$.

The total probability of seeing a good sample is therefore at least $(1-\alpha)^2$. □

Finally, we present an example to show that the upper bound on p is tight up to a $\log(\frac{k}{\alpha})$ factor.

Observation 6. *There exists a table T for which a sampling frequency $p < \frac{\epsilon\log(\frac{1}{1-\alpha})}{t}$ violates $(1, \frac{\epsilon}{2}, \alpha)$-privacy. Here $t+1$ is the number of values with frequency at most $\frac{1}{\epsilon}$.*

We illustrate this through an example. Consider a table T with $t+2$ distinct values; values $1, 2, \ldots, t$ each occur in $\frac{1}{\epsilon}$ rows, value $t+1$ occurs in all the remaining rows except for one row, and value $t+2$ occurs in row i.

Consider a sample S drawn from this table with the property that $s_u > 0$ for some $u \in [1, \ldots, t]$. Consider an attacker who knows the value of all rows of the table except for row i and is trying to find out what the value of row i is. If $s_{t+2} > 0$, $\Pr[S|T_{\{i \to u\}}] = 0$ for any u other than value $t+2$, and $\frac{\Pr[S|T_{\{i\to t+2\}}]}{\Pr[S|T_{\{i\to u\}}]}$ is unbounded.

Otherwise, $s_{t+2} = 0$. Let u be a value in $[1, \ldots, t]$ for which $s_u > 0$. If n_v is the number of rows with value v in the table $T \setminus \{i\}$,

$$\frac{\Pr[S|T_{\{i\to t+2\}}]}{\Pr[S|T_{\{i\to u\}}]} = \frac{\binom{n_u}{s_u}\binom{n_{t+2}+1}{s_{t+2}}}{\binom{n_u+1}{s_u}\binom{n_{t+2}}{s_{t+2}}} = 1 - \frac{s_u}{n_u+1}$$

Since $s_u \geq 1$ and $n_u = \frac{1}{\epsilon}$, this quantity is at most $1 - \frac{\epsilon}{2}$ for any $\epsilon < 1$. In other words, if we see a sample S with the property mentioned above, $(1, \epsilon/2)$-privacy is violated.

Now the probability of choosing a sample S with this property is $(1 - (1 - p)^{\frac{t}{\epsilon}}) \geq 1 - e^{-\frac{pt}{\epsilon}}$. If $p > \frac{\epsilon\log(\frac{1}{1-\alpha})}{t}$, this probability is more than α. This means that we need a sampling frequency $p < \frac{\epsilon\log(\frac{1}{1-\alpha})}{t}$ to ensure $(1, \frac{\epsilon}{2}, \alpha)$-privacy.

5 (c, ϵ, δ)-Privacy

The techniques of [5] show that a mechanism which is $(1, \frac{\epsilon}{c}, \frac{\delta}{c})$-private is also (c, ϵ, δ)-private. The proof looks at a sequence of intermediate tables, each of which differs from the previous one by one row, and shows that $(1, \frac{\epsilon}{c}, \frac{\delta}{c})$-privacy for each of these tables implies (c, ϵ, δ)-privacy for the original table. It is not apparent that the proof applies to us : we do not guarantee $(1, \epsilon, \delta)$-privacy for all tables for a uniform value of p, so a sampling frequency that is $(1, \epsilon, \delta)$-private for the starting table may not maintain the same ϵ, δ guarantees for an intermediate one. In this section, we show an upper bound on the sampling frequency p so that (c, ϵ, δ)-privacy is ensured. Our guarantees are better than the guarantees in [5] in terms of δ and slightly worse in terms of ϵ.

As in the previous section, given ϵ, δ, a table T and k, the total number of distinct values in the table and c, we provide a sampling frequency p that is (c, ϵ, δ)-private. Our main guarantees can be summarized as follows.

Theorem 7. *Given a table T, let $\alpha = \frac{1}{2}\delta$, k be the number of distinct values in T and t be the total number of values in T that occur less than $\frac{2\log(\frac{k}{\alpha})}{\epsilon} + c$ times. Also let $\epsilon' = \max\left(6c\left(1 + \frac{\epsilon c}{2\log(\frac{k}{\alpha})}\right)p, 2c\left(1 + \frac{\epsilon c}{2\log(\frac{k}{\alpha})}\right)(p+\epsilon)\right)$ and let $(1 + \frac{\epsilon c}{2\log(\frac{k}{\alpha})})(p+\epsilon) < \frac{1}{2}$. Then a sample S of T drawn with frequency $p < \frac{\epsilon\log\left(\frac{1}{1-\alpha}\right)}{4t\log(\frac{k}{\alpha})(1+\frac{\epsilon c}{2\log(\frac{k}{\alpha})})}$ is (c, ϵ', δ)-private for $t > 0$. When $t = 0$, a sample S of T drawn with frequency $p \leq \epsilon$ is (c, ϵ', δ)-private.*

We need the assumption $(1 + \frac{\epsilon c}{2\log(\frac{k}{\alpha})})(p+\epsilon) < \frac{1}{2}$ to make sure p is bounded away from 1 by a constant and also ϵ is small compared to c. (Any other constant than $\frac{1}{2}$ would do, but would change the constants in Theorem 7.) If p is too close to 1 or if ϵc is too big, all rows containing a rare value may appear in the sample, leading to a serious breach of privacy.

Comparing these guarantees with those in Section 4, we observe that for a table T, a sampling frequency p that is $(1, \frac{\epsilon}{c(1+\frac{\epsilon c}{2\log(\frac{k}{\alpha})})}, \delta)$-private is also (c, ϵ, δ)-private. We therefore do better than the kind of bound given in [5] in terms of δ and a little worse in terms of ϵ.

Consider all sets of $n - c$ rows in table T and let n_v^c be the minimum number of rows with value v in any such set. In this section, we call a value v a *rare value* if $n_v^c < \frac{2\log(\frac{k}{\alpha})}{\epsilon}$. A *common value* v has $n_v^c > \frac{12\log(\frac{k}{\alpha})}{p}$, where p is the sampling frequency. A value that is neither rare nor common is called an *infrequent value.*

Just as in the previous section, we show that privacy is preserved when we draw a sample with certain properties. We call a sample possessing these properties a *good sample*, with the same definition as in Section 4.

In Lemma 3, we show that releasing a good sample preserves privacy. Because the definition of rare and infrequent values have changed, the fact that a good sample occurs with high probability does not automatically follow from Lemma 2. Instead in Lemma 4, we show that a good sample occurs with high probability. Combining Lemmas 3 and 4, we get a proof of Theorem 7.

Lemma 3. *Let S be a good sample drawn from table T. Then for any set of c rows P_i and any pair of states V and V',*

$$\frac{\Pr[S|T_{\{P_i \to V\}}]}{\Pr[S|T_{\{P_i \to V'\}}]} \leq 1 + \epsilon'$$

where $\epsilon' = \max\left(6c\left(1+\frac{\epsilon c}{2\log(\frac{k}{\alpha})}\right)p, 2c\left(1+\frac{\epsilon c}{2\log(\frac{k}{\alpha})}\right)(p+\epsilon)\right)$, *assuming that* $(1+\frac{\epsilon c}{2\log(\frac{k}{\alpha})})(p+\epsilon) < \frac{1}{2}$.

Proof. Without loss of generality, we assume that the state V has c rows with value v and no rows with value V' and the state v' has no rows with value v and c rows with value v'. The proofs go through when this assumption does not hold, and so we simplify the notation accordingly.

For any value u, let n_u^c denote the number of rows in $T \backslash \{P_i\}$ with value u, and let s_u be the number of rows with value u in the sample S. Note that $T_{\{P_i \to V\}}$ has $n_v^c + c$ rows with value v and $n_{v'}^c$ rows with value v', whereas $T_{\{P_i \to V'\}}$ has $n_{v'}^c + c$ rows with value v' and n_v^c rows with value v.

We now show that since S is a good sample, $s_v < n_v^c + 1$. If the set of rows P_i in T includes no row with value v, this holds trivially; otherwise, we claim that at most n_v^c rows of T that take value v appear in S. Note that T can have at most $n_v^c + c$ rows with value v.

If v is a rare value, there are no rows with value v in a good sample. If v is an infrequent value, the maximum number of rows with value v in the sample is at most $(n_v^c + c)p + 2\log(\frac{k}{\alpha})$ which is at most $n_v^c(1+\frac{c}{n_v^c})(p+\epsilon) < n_v^c(1+\frac{c\epsilon}{2\log(\frac{k}{\alpha})})(p+\epsilon) < n_v^c$ for $(1+\frac{c\epsilon}{2\log(\frac{k}{\alpha})})(p+\epsilon) < \frac{1}{2}$. If v is a common value, the maximum number of rows with value v in the good sample S is at most $(n_v^c + c)p + \sqrt{3(n_v^c + c)p\log(\frac{k}{\alpha})}$, which is at most $(n_v^c + c)p\left(1+\sqrt{\frac{3\log(\frac{k}{\alpha})}{(n_v^c+c)p}}\right) \leq \frac{3}{2}n_v^c(1+\frac{c\epsilon}{2\log(\frac{k}{\alpha})})p < n_v^c + 1$ when $(1+\frac{c\epsilon}{2\log(\frac{k}{\alpha})})p < \frac{1}{2}$.

Therefore,

$$\frac{\Pr[S|T_{\{P_i \to V\}}]}{\Pr[S|T_{\{P_i \to V'\}}]} = \frac{\binom{n_v^c+c}{s_v}\binom{n_{v'}^c}{s_{v'}}}{\binom{n_v^c}{s_v}\binom{n_{v'}^c+c}{s_{v'}}} \leq \frac{1}{(1-\frac{s_v}{n_v^c+1})(1-\frac{s_v}{n_v^c+2})\dots(1-\frac{s_v}{n_v^c+c})}$$

For a rare value v, $s_v = 0$. Therefore,

$$\frac{1}{(1-\frac{s_v}{n_v^c+1})(1-\frac{s_v}{n_v^c+2})\dots(1-\frac{s_v}{n_v^c+c})} \leq 1$$

Note that for any value v, $n_v^c \leq n_v \leq n_v^c + c$. For an infrequent value v, $s_v \leq (n_v^c + c)p + 2\log(\frac{k}{\alpha})$ and $n_v \geq n_v^c \geq \frac{2\log(\frac{k}{\alpha})}{\epsilon}$. This implies that $\frac{s_v}{n_v^c+1} \leq (1+\frac{c}{n_v^c})(p+\epsilon) \leq (1+\frac{c\epsilon}{2\log(\frac{k}{\alpha})})(p+\epsilon)$. Assuming $(1+\frac{c\epsilon}{2\log(\frac{k}{\alpha})})(p+\epsilon) < 1/2$,

$$\frac{1}{(1-\frac{s_v}{n_v^c+1})(1-\frac{s_v}{n_v^c+2})\dots(1-\frac{s_v}{n_v^c+c})} \leq 1 + 2c\left(1+\frac{c\epsilon}{2\log(\frac{k}{\alpha})}\right)(p+\epsilon)$$

Because v is not a rare value, this quantity is at most $1+2c(1+\frac{\epsilon c}{2\log(\frac{k}{\alpha})})(p+\epsilon)$.

For a common value v, s_v is at most $(n_v^c + c)p + \sqrt{3(n_v^c + c)p\log(\frac{k}{\alpha})}$, and $n_v^c p \geq 12\log(\frac{k}{\alpha})$. Therefore $\frac{s_v}{n_v^c+1} \leq \frac{3}{2}(1+\frac{c}{n_v^c})p \leq \frac{3}{2}(1+\frac{c\epsilon}{2\log(\frac{k}{\alpha})})p$.

This implies that

$$\frac{1}{(1-\frac{s_v}{n_v^c+1})(1-\frac{s_v}{n_v^c+2})\dots(1-\frac{s_v}{n_v^c+c})} \leq 1+6\left(1+\frac{\epsilon c}{2\log(\frac{k}{\alpha})}\right)pc$$

for $p(1+\frac{c\epsilon}{2\log(\frac{k}{\alpha})}) < 1/2$. □

Lemma 4. *If the sampling frequency* $p < \frac{\epsilon\log(\frac{1}{1-\alpha})}{4t\log(\frac{k}{\alpha})(1+\frac{\epsilon c}{2\log(\frac{k}{\alpha})})}$, *the probability that a good sample is drawn is at least* $(1-\alpha)^2$.

Proof. Following exactly the same argument as in Lemma 2, the probability that the number of common and infrequent values lie within the requisite bounds is at least $1-\alpha$.

The probability that a rare value v does not occur in S is $(1-p)^{n_v}$. Now there are at most $c+\frac{2\log(\frac{k}{\alpha})}{\epsilon}$ rows with a rare value v. If there are at most t rare values, then the probability that none of these values occur in S is at least $(1-p)^{2t\log(\frac{k}{\alpha})/\epsilon+tc} \geq e^{-4pt(2\log(\frac{k}{\alpha})/\epsilon+c)}$. For $p < \epsilon\frac{\log(\frac{1}{1-\alpha})}{4t\log(\frac{k}{\alpha})(1+\frac{\epsilon c}{2\log(\frac{k}{\alpha})})}$, this probability is at least $1-\alpha$.

The total probability of seeing a good sample is therefore at least $(1-\alpha)^2$. □

6 Future Work

There are many avenues for future work. Our work assumes that the random sample is published in unperturbed form. But it is quite possible that one can draw larger random samples if noise is added to the sample. Such a technique may be useful when k is large. One would have to also understand what impact such noise would have on utility.

Another direction for future research is the study of data streams where individual data points arrive in sequential, not necessarily random order and the question is how to maintain a random sample of the stream without breaching privacy. In this context, the attacker may know who the next person is in the stream, but not know their private values. Existing techniques for maintaining random samples over a stream such as reservoir sampling [13] violate privacy since the sample only changes to include a new person's value when that person arrives.

Finally, in practice, prior to sampling, organizations typically employ other anonymization procedures including, for example, *top-coding*, where individuals with values above a certain percentage of the distribution are placed into a single category, *geographic population thresholds*, where individuals that live in a geographic unit below a specified population level are not disclosed, *random rounding*, wherein numbers that are not multiples of say 10 are randomly rounded to one of the two nearest multiples. Analysis of the privacy/utility of these multi-step anonymization procedures that precede sampling would be an interesting direction for future work.

Acknowledgements

We are very grateful to Kobbi Nissim for extensive discussions. We thank Cynthia Dwork for her insights and Amit Agarwal for early discussion. We thank Shankar Bhamidi for useful suggestions. Finally, we thank the anonymous reviewers for their many thoughtful suggestions.

References

1. A. Blum, C. Dwork, F. McSherry, and K. Nissim. Practical privacy: the sulq framework. In *PODS*, pages 128–138, 2005.
2. U.S. Census Bureau. Public use microdata sample (pums). In *http://www.census.gov/Press-Release/www/2003/PUMS.html*, 2003.
3. S. Chawla, C. Dwork, F. McSherry, A. Smith, and H. Wee. Toward privacy in public databases. In *Theory of Cryptography Conference*, pages 363–385, 2005.
4. I. Dinur and K. Nissim. Revealing information while preserving privacy. In *PODS*, pages 202–210, 2003.
5. C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference*, pages 265–284, 2006.
6. C. Dwork and K. Nissim. Privacy-preserving datamining on vertically partitioned databases. In *CRYPTO*, pages 528–544, 2004.
7. A. Evfimievski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *PODS*, pages 211–222, 2003.
8. O. Goldreich. *Foundations of Cryptography, Volumes I and II*. Cambridge University Press, 2004.
9. K. Kenthapadi, N. Mishra, and K. Nissim. Simulatable auditing. In *PODS*, pages 118–127, 2005.
10. N. Mishra and M. Sandler. Privacy via pseudorandom sketches. In *PODS*, 2006.
11. Social Security Administration: Office of Policy Data. Benefits and earnings public-use file. In *http://www.ssa.gov/policy/docs/microdata/earn/index.html*, 2004.
12. L. Sweeney. Guaranteeing anonymity when sharing medical data, the datafly system. In *Proceedings AMIA Annual Fall Symposium*, 1997.
13. J. Vitter. Random sampling with a reservoir. *ACM Transactions on Mathematical Software*, 11(1):37–57, March 1985.