Differentially Private Weighted Sampling

Edith Cohen
Google Research
Tel Aviv University

Ofir Geri
Stanford University

Tamas Sarlos
Google Research

Uri Stemmer
Ben-Gurion University
Google Research

Abstract

Common datasets have the form of *elements* with *keys* (e.g., transactions and products) and the goal is to perform analytics on the aggregated form of *key* and *frequency* pairs. A weighted sample of keys by (a function of) frequency is a highly versatile summary that provides a sparse set of representative keys and supports approximate evaluations of query statistics. We propose *private weighted sampling* (PWS): A method that sanitizes a weighted sample as to ensure element-level differential privacy, while retaining its utility to the maximum extent possible. PWS maximizes the reporting probabilities of keys and estimation quality of a broad family of statistics. PWS improves over the state of the art even for the well-studied special case of *private histograms*, when no sampling is performed. We empirically observe significant performance gains of 20%-300% increase in key reporting for common Zipfian frequency distributions and accurate estimation with $\times 2$-8 lower frequencies. PWS is applied as a post-processing of a non-private sample, without requiring the original data. Therefore, it can be a seamless addition to existing implementations, such as those optimizes for distributed or streamed data. We believe that due to practicality and performance, PWS may become a method of choice in applications where privacy is desired.

1 Introduction

Weighted sampling schemes are often used to obtain versatile summaries of large datasets. The sample constitutes a representation of the data and also facilitates efficient estimation of many statistics. Motivated by the increasing awareness and demand for data privacy, in this work we construct *privacy preserving* weighted sampling schemes. The privacy notion that we work with is that of *differential privacy* (Dwork et al., 2016), a strong privacy notion that is considered by many researchers to be a gold-standard for privacy preserving data analysis.

Before describing our new results, we define our setting more precisely. Consider an input dataset containing m elements, where each element contains a key x from some domain $\mathcal{X}$. For every key $x \in \mathcal{X}$ we write w_x to denote the multiplicity of x in the input dataset. (We also refer to w_x as the *frequency* of x in the data.) With this notation, it is convenient to represent the input dataset in its *aggregated* form $D = \{(x, w_x)\}$ containing pairs of a key and its frequency $w_x \geq 1$ in the data. Examples of such datasets are plentiful: Keys are search query strings and elements are search requests, keys are products and elements are transactions for the products, keys are locations and elements are visits by individuals, or keys are training examples and elements are activities that generate them. We aim here to protect the privacy of data elements. These example datasets tend to be very sparse, where the number of distinct keys in the data is much smaller than the size $|\mathcal{X}|$ of the domain. Yet, the number of distinct keys can be very large and samples serve as small summaries that can be efficiently stored, computed, and transmitted. We therefore aim for our private sample to retain this property and in particular only include keys that are in the dataset.

The (non-private) sampling schemes we consider are specified by a (non-decreasing) sequence $(q_i)_{i \geq 0}$ of probabilities $q_i \in [0, 1]$, where $q_0 := 0$. Such a sampling scheme takes an input dataset $D = \{(x, w_x)\}$ and returns a *sample* $S \subseteq D$, where each pair (x, w_x) is included in S independently, with probability q_{w_x}. Loosely speaking, given a (non-private) sampling scheme A, we aim in this paper to design a *privacy preserving* variant of A with the goal of preserving its "utility" to the extent possible under privacy constraints. We remark that an immediate consequence of the definition of differential privacy is that keys $x \in \mathcal{X}$ with very low frequencies cannot be included in the private sample S (except with very small probability). On the other hand, keys with high frequencies can be included

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

with probability (close to) 1. Private sampling schemes can therefore retain more utility when the dataset has many keys with higher frequencies or for tasks that are less sensitive to low frequency keys.

Informal Problem 1.1. *Given a (non-private) sampling scheme A, specified by a sampling function q, design a* private *sampling scheme that takes a dataset $D = \{(x, w_x)\}$ and outputs a "sanitized" sample $S^* = \{(x, w_x^*)\}$. Informally, the goals are:*

1. *Each pair $(x, w_x) \in D$ is sampled with probability "as close as possible" to the non-private sampling probability q_{w_x}.*
2. *The sanitized sample S^* provides utility that is "as close as possible" to that of a corresponding non-private sample S. In our constructions, the sanitized frequencies w_x^* would be random variables from which we can estimate ordinal and linear statistics with (functions of) the frequency w_x.*

Informal Problem 1.1 generalizes one of the most basic tasks in the literature of differential privacy – *privately computing histograms*. Informally, algorithms for private histograms take a dataset $D = \{(x, w_x)\}$ as input, and return, in a differentially private manner, a "sanitized" dataset $D^* = \{(x, w_x^*)\}$. It is often required that the output D^* is *sparse*, in the sense that if $w_x = 0$ then $w_x^* = 0$. Commonly, we seek to minimize the expected or maximum error of estimators applied to $\boldsymbol{w}^*$ of statistics on $\boldsymbol{w}$. One well-studied objective is to minimize $\max_{x \in X} \mid w_x - w_x^*|$. The work on private histograms dates all the way back to the paper that introduced differential privacy (Dwork et al., 2016), and it has received a lot of attention since then, e.g., (Korolova et al., 2009; Hardt and Talwar, 2010; Beimel et al., 2014, 2016; Bun and Steinke, 2016; Bun et al., 2019; Balcer and Vadhan, 2018; Bun et al., 2018). Observe that the private histogram problem is a special case of Informal Problem 1.1, where $q \equiv 1$.

At first glance, one might try to solve Informal Problem 1.1 by a reduction to the private histogram problem. Specifically, we consider the baseline where the data is first "sanitized" using an algorithm for private histograms, and then a (non-private) weighted sampling algorithm is applied to the sanitized data (treating the sanitized frequencies as actual frequencies). This framework, of first sanitizing the data and then sampling it was also considered in (Cormode et al., 2012). We show that this baseline is sub-optimal, and improve upon it in several axes.

1.1 Our Contributions

Our proposed framework, *Private Weighted Sampling* (PWS), takes as input a non-private weighted sample S that is produced by a (non-private) weighted sampling scheme. We apply a "sanitizer" to the sample S to obtain a respective privacy-preserving sample S^*. Our proposed solution has the following advantages.

Practicality. The private version is generated from the sample S as a post-processing step without the need to revisit the original dataset, which might be massive or unavailable. This means that we can augment existing implementations of non-private sampling schemes and retain their scalability and efficiency. This is particularly appealing for sampling schemes designed for massive distributed or streamed data that use small sketches and avoid a resource-heavy aggregation of the data (Gibbons and Matias, 1998; Estan and Varghese, 2002; Cohen et al., 2014; Andoni et al., 2011; Cohen et al., 2012; Cohen, 2018; Jayaram and Woodruff, 2018; Cohen and Geri, 2019; Cohen et al., 2020a). Our code is available at `github.com/google-research/google-research/tree/master/private_sampling`.

Benefits of end-to-end privacy analysis PWS achieves better utility compared to the baseline of first sanitizing the data and then sampling. In spirit, our gains follow from a well-known result in the literature of differential privacy stating that applying a differentially private algorithm on a random sample from the original data has the effect of boosting the privacy guarantees of the algorithm (Chaudhuri and Mishra, 2006; Kasiviswanathan et al., 2011; Bun et al., 2015). Our solution is derived from a precise *end-to-end* formulation of the privacy constraints that account for the benefits of the random sampling in our privacy analysis.

Optimal reporting probabilities. PWS is optimal in that it maximizes the probability that each key x is included in the private sample. The private reporting probability of a key x depends on the privacy parameters, frequency, and sampling rate and is at most the non-private sampling probability q_{w_x}. The derivation is provided in Section 4.

Estimation of linear statistics. Linear statistics according to a function of frequency have the form: $\sum_x L(x)g(w_x)$, where $g(w_x) \geq 0$ is a non-decreasing function of frequency with $g(0) := 0$. The most common use case is when $L(x)$ is a predicate and $g(w) := w$ and the statistics is the sum of frequencies of keys that satisfy the selection L. Our PWS sanitizer in Section 5 maintains optimal reporting probabilities and provides private information on frequencies of keys. We show that generally differential privacy does not allow for unbiased estimators for statistics without significant increase in variance. We propose biased but nonnegative and low-variance estimators.

Estimation of ordinal statistics. Ordinal statistics, such as (approximate) quantiles and top-k sets, are derived from the order of keys that is induced by their frequencies. This order can be approximated by the order induced by PWS

sanitized frequencies. We show that PWS is optimal, over all DP sanitization schemes, for a broad class of ordinal statistics. In particular, PWS maximizes the probability that *any* pair is concordant and maximizes the expected Kendall-τ rank correlation between the order induced by sanitized and true frequencies.

Improvement over prior baselines. We show analytically and empirically in Section 8 that we obtain orders of magnitude increase in reporting probability in low-frequency regimes. For estimation tasks, both PWS and prior schemes have lower error for higher frequencies but PWS obtains higher accuracy for frequencies that are $\times$2-8 lower than prior schemes. This is particularly helpful for datasets/selections with many mid-low frequency keys.

Improvement for private histograms. As an important special case of our results, we improve upon the state-of-the-art constructions for private (sparse) histograms (Korolova et al., 2009; Bun et al., 2019). These existing constructions obtain privacy properties by adding Laplace or Gaussian noise to the frequencies of the keys whereas we directly formulate and solve elementary constraints. Let π^*_i denote the PWS reporting probability of a key with frequency i, when applied to the special case of private histograms. Let ϕ_i denote the reporting probability of the state-of-the-art solution for private (sparse) histograms of (Korolova et al., 2009; Bun et al., 2019). Clearly π^*_i is always at least ϕ_i. We show that in low-frequency regimes we have $\pi^*_i/\phi_i \approx 2i$. Similarly for estimation tasks, PWS provides more accurate estimates in these regimes. Qualitatively, PWS and private histograms have high reporting probabilities and low estimation error for high frequencies. But PWS significantly improves on low to medium frequencies, which is important for distributions with long tails. We empirically show gains of 20%-300% in overall key reporting for Zipf-distributed frequencies. As private histograms are one of the most important building blocks in the literature of differential privacy, we believe that our improvement is significant (both in theory and in practice).

Due to space limitations, we refer the reader to (Cohen et al., 2020b) for proofs and further details.

2 Related Work

The suboptimality of the Laplace mechanism for anonymization was noted by Ghosh et al. (2012). In our language, Ghosh et al. studied the non-sparse case, where all values, including 0 values, can be reported with added noise. They did not consider sampling, and studied pure differential privacy. Instead of Laplace noise, they propose the use of a symmetric Geometric distribution and establish it is optimal for certain estimation tasks. This can be viewed as a special case of what we do in that our schemes converge to that when there is no sampling, we use pure differential privacy, and when frequencies are large (so the effect of the sparse case constraint dissipates). Ghosh et al. establish the optimality of unbiased estimators for some frequency statistics when loss is symmetric. We show that bias is necessary in the sparse case and propose estimators that control the bias and variance.

Key reporting was formulated and studied as *differentially private set union* problem (Gopi et al., 2020). They studied it without sampling, in a more general user privacy setting, and proposed a truncated Laplace noise mechanism similar to (Korolova et al., 2009; Bun et al., 2019).

Recent independent work by Desfontaines et al. (2020) derived the optimal scheme for key reporting for sparse private histograms, a special case of our solution when there is no sampling.

3 Preliminaries

We consider data in the form of a set of elements $\mathcal{E}$, where each element $e \in \mathcal{E}$ has a key $e.\mathrm{key} \in \mathcal{X}$. The *frequency* of a key x, $w_x := |\{e \in \mathcal{E} \mid e.\mathrm{key} = x\}|$, is defined as the number of elements with $e.\mathrm{key} = x$. The *aggregated form* of the data, known in the DP literature as its *histogram*, is the set of key and frequency pairs $\{(x, w_x)\}$. We use the vector notation $\boldsymbol{w}$ for the aggregated form. We will use $m := |\mathcal{E}|$ for the number of elements and n for the number of distinct keys in the data.

3.1 Weighted Sampling

We consider a very general form of without-replacement sampling schemes. Each scheme is specified by non-decreasing probabilities $(q_i)_{i\geq 1}$. The probability that a key is sampled depends on its frequency – a key with frequency i is sampled independently with probability q_i. Our proposed methods apply with any non-decreasing (q_i).

Threshold sampling is a popular class of weighted sampling schemes. We review it for concreteness and motivation and use it in our empirical evaluation of PWS. A threshold sampling scheme (see Algorithm 1) is specified by $(\mathcal{D}, f, \tau)$, where $\mathcal{D}$ is a distribution, f is a function of frequency, and τ is a numeric threshold value that specifies the sampling rate. For each key we draw i.i.d. $u_x \sim \mathcal{D}$. The two common choices are $\mathcal{D} = \mathsf{Exp}[1]$ for a probability proportional to size without replacement (ppswor) sample (Rosén, 1972) and $\mathcal{D} = U[0, 1]$ for a Poisson Probability Proportional to Size (PPS) sample (Ohlsson, 1990, 1998; Duffield et al., 2007). A key x is included in the sample if $u_x \leq \tau f(w_x)$. The probability that a key with frequency i is sampled is

$$q_i := \Pr_{u\sim\mathcal{D}}[u < f(i)\tau] \ . \tag{1}$$

Threshold sampling is related to bottom-k (order) sampling (Rosén, 1997; Ohlsson, 1990; Duffield et al., 2007;

Cohen and Kaplan, 2007, 2008) but instead of specifying the sample size k we specify an inclusion threshold τ. Ppswor is equivalent to drawing keys sequentially with probability proportional to $f(w_x)$. The bottom-k version stops after k keys and the threshold version has a stopping rule that corresponds to the threshold. The bottom-k version of Poisson PPS sampling is known as sequential Poisson or Priority sampling.

Algorithm 1: Threshold Sampling

```
// Threshold Sampler:
Input: Dataset w of key frequency pairs (x, w_x); distribution D, function
       f, threshold τ
Output: Sample S of key-frequency pairs from w
begin
    S ← ∅
    foreach (x, w_x) ∈ w do
        Draw independent u_x ~ D
        if u_x < f(w_x)τ then
            S ← S ∪ {(x, w_x)}
    return S
```

Since PWS applies a sanitizer to a sample, it inherits the efficiency of the base sampling scheme. Threshold sampling (via the respective bottom-k schemes) can be implemented efficiently using small sketches (of size expected sample size) on aggregated data that can be distributed or streamed (Duffield et al., 2007; Rosén, 1997; Ohlsson, 1998; Cohen and Kaplan, 2007). On unaggregated datasets, it can be implemented using small sketches for some functions of frequency including the moments $f(w) = w^p$ for $p \in [0, 2]$ (Cohen et al., 2012; Cohen, 2018; Cohen and Geri, 2019; Cohen et al., 2020c).

Our methods apply with a fixed threshold τ. But the treatment extends to when the threshold is privately determined from the data. If we have a private approximation of the total count $\|f(\boldsymbol{w})\|_1 := \sum_x f(w_x)$ we can set $\tau \approx k/\|f(\boldsymbol{w})\|_1$. This provides (from the non-private sample that corresponds to the threshold) estimates with additive error $\|f(\boldsymbol{w})\|_1/\sqrt{k}$ for statistics with function of frequency $g = f$ and when L is a predicate.

3.2 Differential Privacy

The privacy requirement we consider is *element-level* differential privacy. Two datasets with aggregated forms $\boldsymbol{w}$ and $\boldsymbol{w}'$ are neighbors if $\|\boldsymbol{w} - \boldsymbol{w}'\|_1 = 1$, that is, the frequencies of all keys but one are the same and the difference is at most 1 for that one key. The privacy requirements are specified using two parameters $\varepsilon, \delta \geq 0$.

Definition 3.1 (Dwork et al., 2016)**.** *A mechanism M is (ε, δ)-differentially private if for any two neighboring inputs $\boldsymbol{w}$, $\boldsymbol{w}'$ and set of potential outputs T,*

$$\Pr[M(\boldsymbol{w}) \in T] \leq e^{\varepsilon} \Pr[M(\boldsymbol{w}') \in T] + \delta \ . \qquad (2)$$

Algorithm 2: Private Weighted Samples

```
// Sanitized Keys:
Input: (ε, δ), weighted sample S, taken with non-decreasing probabilities
       (q_i)_{i≥1}
Output: Private sample of keys S*
Compute (p_i)_{i≥1}     // Reporting probabilities per freq.
begin // Sanitize using scheme
    S* ← ∅
    foreach (x, w_x) ∈ S do
        With probability p_{w_x}, S* ← S* ∪ {x}
    return S*
// Sanitized keys and frequencies:
Input: (ε, δ), weighted sample S, taken with non-decreasing probabilities
       (q_i)_{i≥1}
Output: Sanitized sample S*
Compute probability vectors (p_{i•})_{i≥1}          // Reported values
begin // Sanitize using scheme
    S* ← ∅
    foreach (x, w_x) ∈ S do
        Draw j ~ p_{w_x •}          // By probability vector
        if j > 0 then
            S* ← S* ∪ {(x, j)}
    return S*
// Estimator:
Input: Sanitized sample S* = {(x, j_x)}, {π_{i,j}} (where π_{ij} := p_{ij} q_i ,
       functions g(i), L(x)
Output: Estimate of the linear statistics Σ_x L(x)g(x)
begin
    Compute (a_j)_{j≥1} using {π_{ij}} and g(i)          // Per-key
      estimates for g()
    return Σ_{(x,j_x)∈S*} L(x) a_{j_x}
```

3.3 Private Weighted Samples

Given a (non-private) weighted sample S of the data in the form of key and frequency pairs and (a representation) of the sampling probabilities $(q_i)_{i\geq 1}$ that guided the sampling, our goal is to release as much of S as we can without violating element-level differential privacy.

We consider two utility objectives. The basic objective, *sanitized keys*, is to maximize the reporting probabilities of keys in S. The private sample in this case is simply a subset of the keys in S. The refined objective is to facilitate estimates of linear frequency and order statistics. The private sample includes sanitized keys from S together with information on their frequencies. The formats of the sanitizers and estimators are provided as Algorithm 2.

4 Sanitized Keys

Algorithm 3: Compute π_i for Sanitizing Keys

```
Input: (ε, δ), non-decreasing sampling probabilities (q_i)_{i≥1},
       Max_Frequency
π_0 ← 0
foreach i = 1, ..., Max_Frequency do
    π_i ← min{q_i, e^ε π_{i-1} + δ, 1 + e^{-ε}(π_{i-1} + δ - 1)}
return (π_i)_{i=1}^{Max_Frequency}
```

A sanitizer C uses a representation of the non-decreasing $(q_i)_{i\geq 1}$ and computes respective probabilities $(p_i)_{i\geq 1}$. A non-private sample S can then be sanitized by considering

each pair $(x, w_x) \in S$ and reporting the key x independently with probability p_{w_x}.

We find it convenient to express constraints on $(p_i)_{i\geq 1}$ in terms of the *end-to-end* reporting probability of a key x with frequency i (probability that x is sampled and then reported):

$$\pi_i := p_i q_i = \Pr[x \in C(A(\boldsymbol{w}))] \ .$$

Keys of frequency 0 are not sampled or reported and we have $q_0 = 0$ and $\pi_0 := 0$. The objective of maximizing p_i corresponds to maximizing π_i. We establish the following:

Lemma 4.1. *Consider weighted sampling scheme A where keys are sampled independently according to a non-decreasing $(q_i)_{i\geq 1}$ and a key sanitizer C (Algorithm 2) is applied to the sample. Then the probabilities $p_i \leftarrow \pi_i/q_i$, where π_i are the iterates computed in Algorithm 3, are each at the maximum under the DP constraints for $C(A())$. Moreover, $(\pi_i)_{i\geq 1}$ is non-decreasing.*

4.1 Structure and Properties of $(\pi_i)_{i\geq 1}$

The solution as computed in Algorithm 3 applies with any non-decreasing q_i. We explore properties of the solution that allow us to compute and store it more efficiently and understand the reporting loss (reduction in reporting probabilities) that is due to the privacy requirement.

We provide closed-form expressions of the solution π^*_i that corresponds to $q_i = 1$ for all i (aka the private histogram problem). We will use the following definition of $L(\varepsilon, \delta)$. To simplify the presentation, we assume that ε and δ are such that L is an integer (this assumption can be removed).

$$L(\varepsilon, \delta) := \frac{1}{\varepsilon} \ln\left(\frac{e^\varepsilon - 1 + 2\delta}{\delta(e^\varepsilon + 1)}\right) \approx \frac{1}{\varepsilon}\ln\left(\frac{\varepsilon}{2\delta}\right) \quad (3)$$

Lemma 4.2. *When $q_i = 1$ for all i, the sequence $(\pi_i)_{i\geq 1}$ computed by Algorithm 3 has the form:*

$$\pi^*_i = \begin{cases} \delta\frac{e^{\varepsilon i}-1}{e^\varepsilon - 1} & i \leq L+1 \\ 1 - \delta\frac{e^{\varepsilon(2L+2-i)}-1}{e^\varepsilon - 1} & L+1 \leq i \leq 2L+1 \\ 1 & i \geq 2L+2 \end{cases} \quad (4)$$

For the general case where the q_i's can be smaller than 1, we bound the number of frequency values for which $\pi_i < q_i$. On these frequencies, the private reporting probability is strictly lower than that of the original non-private sample, and hence there is reporting loss due to privacy.

Lemma 4.3. *There are at most $2L(\varepsilon, \delta) + 1$ values i such that $\pi_i < q_i$, where L is as defined in* (3).

We now consider the structure of the solution for threshold sampling. The solution has a particularly simple form that can be efficiently computed and represented.

Lemma 4.4. *When the sampling probabilities $(q_i)_{i\geq 1}$ are those of threshold ppswor sampling with $f(i) = i$ then the solution has the form $\pi_i = \pi^*_i$ for $i < \ell$ and $\pi_i = q_i$ for $i \geq \ell$, where $\ell = \min\{i : \pi^*_i > q_i\}$ is the lowest position with $\pi^*_i > q_i$ and π^*_i is as defined in* (4).

5 Sanitized Keys and Frequencies

Algorithm 4: Compute $(\pi_{i,j})$ for Sanitized Frequencies

Input: $((\varepsilon, \delta)$, non-decreasing $(q_i)_{i\geq 1})$, Max_Frequency
Output: $(\pi_{i,j})$ for $0 \leq j \leq i \leq$ Max_Frequency
$\pi_{0,0} \leftarrow 1, \pi_0 = 0$
foreach $i = 1, \ldots,$ *Max_Frequency* **do** // Iterate over rows
  $\pi_i \leftarrow \min\{q_i, e^\varepsilon \pi_{i-1} + \delta, 1 + e^{-\varepsilon}(\pi_{i-1} + \delta - 1)\}$
  // End-to-end probability to output a key with frequency i
  $\pi_{i,0} \leftarrow 1 - \pi_i$
  foreach $j = 1, \ldots, i-1$ **do** // Set lower bound values; use $[a]_+ := \max\{a, 0\}$
    $\pi_{i,j} \leftarrow e^{-\varepsilon}\left(\sum_{h=1}^{j} \pi_{i-1,h} - \delta\right) - \sum_{h=1}^{j-1} \pi_{i,h}$
    $+ \left[e^{-\varepsilon}\pi_{i-1,0} - \pi_{i,0}\right]_+$
    $\pi_{i,j} \leftarrow [\pi_{i,j}]_+$
  $R \leftarrow \pi_i - \sum_{h=1}^{i-1} \pi_{i,h}$ // Remaining probability to assign
  foreach $j = i, \ldots, 1$ **do** // Set final values for $\pi_{i,j}$
    if $R = 0$ **then Break**
    $U \leftarrow e^\varepsilon \sum_{h=j}^{i-1} \pi_{i-1,h} + \delta - \sum_{h=j+1}^{i} \pi_{i,h}$ // Max value allowed for $\pi_{i,j}$
    if $U - \pi_{i,j} \leq R$ **then**
      $R \leftarrow R - (U - \pi_{i,j})$
      $\pi_{i,j} \leftarrow U$
    else
      $\pi_{i,j} \leftarrow \pi_{i,j} + R$
      $R \leftarrow 0$
return $(\pi_{i,j})$

A frequency sanitizer C returns keys x together with sanitized information on their frequency. We use $p_{i,j}$ for the probability that C reports $j \in [t]$ for a sampled key that has frequency i, with $p_{i,0}$ being the probability that the sampled key is not reported. We have that $\sum_{j=1}^{t} p_{i,j}$ is the total probability that a sampled key with frequency i is reported by C. We use

$$\pi_{i,j} \leftarrow q_i p_{i,j}$$

for the end-to-end probability that a key with frequency i is sampled and reported in the private sample with sanitized value j. For notation convenience, we use $\pi_{i,0} := 1 - \sum_{j=1}^{t} \pi_{i,t}$ for the probability that a key is not reported, making $\pi_{i,\bullet}$ probability vectors. The reader can interpret the returned value j as a token from an ordered domain. The estimators we propose depend only on the order of tokens and not their values and hence are invariant to a mapping of the domain that preserves the order.

We express constraints on $(\pi_{i,j})_{i\geq 0, j\geq 0}$. For a solution to be *realizable*, we must have end-to-end reporting probabil-

ities that do not exceed the sampling probabilities:

$$\forall i,\ \sum_{j=1}^{t} \pi_{i,j} \leq q_i \ . \tag{5}$$

The DP constraints are provided in the sequel. Note that we must have $\sum_{j=1}^{t} \pi_{i,j} \leq \pi_i$, where $(\pi_i)_{i\geq 1}$ is the solution for sanitized keys (Algorithm 3), this because the sanitized frequencies DP constraints are a superset of the sanitized keys constraints – we obtain the latter in the former by considering outputs that group together all outputs with a key x with all possible values of $j > 0$. For optimality, we seek solutions that (informally) *maximally separate* the distributions of different frequencies (minimize the overlap), over all possible DP frequency reporting schemes. We will see that maximum separation can (i) always be achieved by a discrete distribution (when the maximum frequency is bounded) and (ii) can be simultaneously achieved between any pair of frequencies. In particular, we maintain optimal reporting, that is, $\sum_{j=1}^{t} \pi_{i,j} = \pi_i$ and $\pi_{i,0} = \overline{\pi}_i$. The solutions we express are such that for $i_1 > i_2$, $\pi_{i_1,\bullet}$ (first-order) stochastically dominates $\pi_{i_2,\bullet}$: That is, for any h, the probability of a token $j \geq h$ is non-decreasing with frequency.

We present two algorithms that express $(\pi_{i,j})$. Algorithm 4 provides a simplified construction, where t is equal to the maximum frequency and we always report $j \leq i$ for a key with frequency i. The sanitizer satisfies realizability and DP and has optimal key reporting but attains maximum separation only under some restrictions. The values $\pi_{i,j}$ are specified in order of increasing i, where the row $\pi_{i,\bullet}$ is set so that the probability mass of π_i is pushed to the extent possible to higher j values.

Algorithm 5 specifies PDFs (f_i) for a frequency sanitizer. The PDFs have a discrete point mass at 0 (that corresponds to the probability of not reporting) and are piecewise constant elsewhere. The scheme is a refinement of the scheme of Algorithm 4 and, as we shall see, for any (q_i) and (ε, δ), it maximally separates sanitized values for different frequencies. The construction introduces at most $3m$ distinct breakpoints for frequencies up to m and can be discretized to have an equivalent $(\pi_{i,j})$ form with $j \in [3m]$.

Theorem 5.1. *The sanitizer with $(\pi_{i,j})$ expressed in Algorithm 5 satisfies:*

1. *$\forall i,\ \sum_{j=1}^{i} \pi_{i,j} = \pi_i$, and in particular, (5) holds and the sanitizer is realizable.*
2. *(ε, δ)-DP*
3. *Maximum separation: For each i, there is an index c_i so that subject to the above and to row $\pi_{i-1,\bullet}$, for all $j' \leq c_i$, the sum $\sum_{j=1}^{j'} \pi_{ij}$ is at a minimum and for all $j' \geq c_i$, the sum $\sum_{j=j'}^{i} \pi_{ij}$ is at a maximum.*

The $(\pi_{i,j})$ expressed by Algorithm 4 satisfy maximum separation (Property 3) under the particular restrictions on the reported values (that only i different outputs are possible for frequencies up to i).

The $(\pi_{i,j})$ expressed by Algorithm 5 and then discretized satisfy maximum separation (Property 3) unconditionally, over all DP frequency sanitization schemes.

For the special case where $q_i = 1$ for all i, Algorithm 4 provides maximum separation. We provide a closed-form expression for the solution $\pi^*_{i,j}$.

Lemma 5.2. *Let the DP parameters (ε, δ) be such that $L(\varepsilon, \delta)$ as in (3) is integral. Let (π^*_{ij}) be the solution computed in Algorithm 4 for $q_i = 1$ for all i. Then the matrix with entries π^*_{ij} for $i, j \geq 1$ has a lower triangular form, with the non-zero entries as follows:*

For $j \in \{\max\{1, i-2L\}, \ldots, i\}$, $\pi^*_{ij} = \pi^*_{i-j+1} - \pi^*_{i-j}$.

Equivalently, $\pi^*_{i,j} = \begin{cases} \delta e^{(i-j)\varepsilon} & \text{if } 0 \leq i-j \leq L \\ \delta e^{(2L-(i-j))\varepsilon} & \text{if } L+1 \leq i-j \leq 2L \end{cases}$.

Algorithm 5: Compute (f_i) for Sanitizing Frequencies

Input: (ε, δ), non-decreasing sampling probabilities $(q_i)_{i\geq 1}$, Max_Frequency

Output: $(f_i)_{i=0}^{\text{Max\_Frequency}}$, where $f_i : [0, i]$ // PDF of sanitized frequency for frequency i: discrete mass at $f_i(0)$ (probability of not reporting) and density on $(0, i]$

$f_0(0) \leftarrow 1$; $\pi_i \leftarrow 0$ // Keys with frequency 0 are never reported

for $i \leftarrow 1$ **to** *Max_Frequency* **do** // Specify $f_i : [0, i]$

  $\pi_i \leftarrow \min\{q_i, e^{\varepsilon}\pi_{i-1} + \delta, 1 + e^{-\varepsilon}(\pi_{i-1} + \delta - 1)\}$;

  $f_i(0) \leftarrow 1 - \pi_i$ // Reporting probability for i

  $f_i(i-1, i] \leftarrow \min\{\pi_i, \delta\}$

  // Represent a function $f_L : (0, i-1]$ that "lower bounds" f_i

  if $\max\{0, e^{-\varepsilon} f_{i-1}(0) - f_i(0)\} + \int_{0+}^{i-1} f_{i-1}(x)dx \leq \delta$ **then**

    $f_L(0, i-1] \leftarrow 0$

  else

    $b_i \leftarrow z$ that solves $\max\{0, e^{-\varepsilon} f_{i-1}(0) - f_i(0)\} + \int_{0+}^{z} f_{i-1}(x)dx = \delta$

    // Well defined, as from DP, we always have $\max\{0, e^{-\varepsilon} f_{i-1}(0) - f_i(0)\} \leq \delta$

    $f_L(0, b_i] \leftarrow 0$

    for $x \in (b_i, i-1]$ **do** $f_L(x) \leftarrow e^{-\varepsilon} f_{i-1}(x)$

  // Point where $f_i(x) - f_{i-1}(x)$ switches sign

  $c_i \leftarrow z$ that solves $\int_0^z f_L(x)dx + e^{\varepsilon} \int_z^{i-1} f_{i-1}(x)dx = \pi_i - \min\{\pi_i, \delta\}$

  // Any solution $z \in (0, i-1]$

  for $x \in (c_i, i-1]$ **do** $f_i(x) = e^{\varepsilon} f_{i-1}(x)$

  for $x \in (0, c_i]$ **do** $f_i(x) \leftarrow f_L(x)$

return $(f_i)_{i=0}^{\text{Max\_Frequency}}$

6 Estimation of Ordinal Statistics

The sanitized frequencies can be used for estimation of statistics specified with respect to the actual frequencies. In this section we consider *ordinal* statistics, that only depend on the order of frequencies but not on their nominal values. Ordinal statistics include (approximate) top-k

set, quantiles, rank of a key, set of keys with a higher (or lower) rank than a specified key, and more. We approximate ordinal statistics from the ordering of keys that is induced by sanitized frequencies. The quality of estimated ordinal statistics is determined by the match between the order induced by exact frequencies and the order induced by sanitized frequencies. We say that the two orders are *concordant* on a subset of keys $\{x_i\}$, when they match on that subset. Since the output of our sanitizer is stochastic, we consider the probability of a subset being concordant. When sanitized values are discrete and two keys have the same sanitized value, we use probability of 0.5 that two keys are concordant.

We define a measure of separation between distributions f_{i_1} and f_{i_2} at a certain quantile value α and show that the (f_i) constructed by Algorithm 5 maximize it pointwise for any i_1, i_2, α. This measure generalizes and follows from Property 3 stated in Theorem 5.1. As a corollary we show:

Corollary 6.1. *The sanitizing scheme specified by the (f_i) computed by Algorithm 5 maximizes the following: The probability that a subset of keys is concordant, the probability that a key is correctly ordered with respect to all other keys, and the expected Kendall-τ rank correlation.*

Note that we get optimality in a strong sense – there is no Pareto front where concordant probability on some pairs of frequencies needs to be reduced in order to get a higher value for other pairs.

7 Estimation of Linear Frequency Statistics

The objective is to estimate statistics of the form

$$s := \sum_x L(x)g(w_x) \ . \tag{6}$$

We briefly review estimators for the non-private setting where the sample consists of pairs (x, w_x) of keys and their frequency. We use the per-key inverse-probability estimators (Horvitz and Thompson, 1952) (also known as importance sampling). The estimate $\widehat{g(w_x)}$ of $g(w_x)$ is 0 if key x is not included in the sample and otherwise the estimate is $a_{w_x} := \frac{g(w_x)}{q_{w_x}}$. These estimates are nonnegative, a desired property for nonnegative values, and are also unbiased when $q_{w_x} > 0$. The estimate for the query statistics (6) is

$$\hat{s} := \sum_{(x,w_x)} L(x)\widehat{g(w_x)} = \sum_{(x,w_x)\in S} L(x)a_{w_x} \ . \tag{7}$$

7.1 Estimation with Sanitized Samples

We now consider estimation from sanitized samples S^*. We specify our estimators $(a_j)_{j\geq 1}$ in terms of the reported sanitized frequencies j. The estimate is 0 for keys that are not reported and are a_j when reported with value j. The estimate of the statistics is

$$\widehat{s} := \sum_{(x,j)\in S^*} L(x)a_j \ . \tag{8}$$

As for choosing $(a_j)_{j\geq 1}$, a first attempt is the unique unbiased estimator: The unbiasedness constraints $\forall i, \ \sum_{j=1}^{i} \pi_{ij}a_j = g(i)$ form a triangular system with a unique solution $(a_j)_{j\geq 1}$. However, $(a_j)_{j\geq 1}$ may include negative values and estimates have high variance. We argue that bias is unavoidable with privacy: First, the inclusion probability of keys with frequency $w_x = 1$ can not exceed δ. Therefore, the variance contribution of the key to any unbiased estimate is at least $1/\delta$. Typically, δ is chosen so that $1/\delta \gg nk$, where n is the support size and k the sample size, so this error can not be mitigated. Second, we show in the full version that even for the special case of $q = 1$, *any* unbiased estimator applied to the output of *any* sanitized keys and frequencies scheme with optimal reporting probabilities must assume negative values. That is, DP schemes do not admit unbiased nonnegative estimators without compromising reporting probabilities. We therefore seek estimators that are biased but balance bias and variance and are nonnegative. In our evaluation we use the following Maximum Likelihood estimator (MLE):

$$a_j \leftarrow \frac{g(i)}{\pi_i}, \text{ where } i = \arg\max_h \pi_{hj} \ . \tag{9}$$

This estimate is "right" for the frequency i for which the probability of reporting j is maximized.

We express the expected value, bias, Mean Squared Error (MSE), and variance of the per-key estimate for a key with frequency i: $\mathsf{E}_i := \sum_{j=1}^{i} \pi_{i,j}a_j$, $\mathsf{Bias}_i := \mathsf{E}_i - g(i)$, $\mathsf{MSE}_i := \overline{\pi}_i g(i)^2 + \sum_{j=1}^{i} \pi_{i,j}(a_j - g(i))^2$, $\mathsf{Var}_i := \mathsf{MSE}_i - \mathsf{Bias}_i^2$. For the sum estimate (8) we get: $\mathsf{Bias}[\widehat{s}] = \sum_x L(x)\mathsf{Bias}_{w_x}$, $\mathsf{Var}[\widehat{s}] = \sum_x L(x)^2\mathsf{Var}_{w_x}$, $\mathsf{MSE}[\widehat{s}] = \mathsf{Var}[\widehat{s}] + \mathsf{Bias}[\widehat{s}]^2$, and $\mathsf{NRMSE}[\widehat{s}] = \frac{\sqrt{\mathsf{MSE}[\widehat{s}]}}{s}$. Note that the variance component of the normalized squared error $\mathsf{MSE}[\widehat{s}]/s^2$ decreases linearly with support size whereas the bias component may not. We therefore consider both the variance and bias of the per-key estimators and qualitatively seek low bias and "bounded" variance. We measure quality of statistics estimators using the Normalized Root Mean Squared Error (NRMSE).

8 Performance Analysis

We study the performance of PWS on the key reporting and estimation objectives and compare with a baseline method that provides the same privacy guarantees. We use precise expressions (not simulations) to compute probabilities, bias, variance, and MSE of the different methods.

8.1 Private Histograms Baseline

We review the *Stability-based Histograms* (SbH) method of (Bun et al., 2019; Korolova et al., 2009; Vadhan, 2017), which we use as a baseline. SbH, provided as Algorithm 6, is designed for the special case when $q_i = 1$ for all frequencies. The input S is the full data of pairs of keys and positive frequencies (x, w_x). The private output S^* is a subset of the keys with positive sanitized frequencies (x, w^*_x).

Algorithm 6: Stability-based Histograms (SbH)

Input: (ε, δ) , $S = \{(x, w_x)\}$ where $w_x > 0$
Output: Key value pairs S'
$S^* \leftarrow \emptyset$ `// Initialize private histogram`
$T \leftarrow (1/\varepsilon)\ln(1/\delta) + 1$ `// Threshold`
foreach $(x, w_x) \in S$ **do**
  $w^*_x \leftarrow w_x + \mathsf{Lap}[\frac{1}{\varepsilon}]$ `// Add Laplace random variable`
  if $w^*_x \geq T$ **then**
    $S^* \leftarrow S^* \cup (x, w^*_x)$

The SbH method is considered the state of the art for sparse histograms (only keys with $w_x > 0$ can be reported). The method returns non-negative $w^*_x > 0$ sanitized frequencies. For the case of no sampling, we compare PWS (with $q \equiv 1$) with SbH. We use the SbH sanitized frequencies directly for estimation. For sampling, our baseline is *sampled-SbH*: The data is first sanitized using SbH and then sampled, using a weighted sampling algorithm with q, while treating the sanitized frequencies as actual frequencies. For estimation, we apply the estimator (7) (which in this context is biased).

Reporting Probabilities: No Sampling We start with the case of no sampling and the objective of maximizing the number of privately reported keys. We compare the PWS (optimal) probabilities π^* (4) to the baseline SbH reporting probabilities ϕ. Figure 1 shows reporting probability per frequency for selected DP parameters. We can see that with both private methods the reporting probability reaches 1 for high frequencies but PWS (Opt) reaches the maximum earlier and is significantly higher than ϕ along the way. Analytically from the expressions we can see that for $i \leq L(\varepsilon, \delta)$, $\pi^*_i/\phi_i \in [2, 2/\varepsilon]$ and for lower i we have $\pi^*_i/\phi_i \approx 2i$. We can also see that π^*_{2L+1} reaches 1 at $2L \approx \frac{2}{\varepsilon}\ln(\varepsilon/\delta)$ whereas $\phi_i > 1 - \delta$ for $i \approx \frac{2}{\varepsilon}\ln(1/\delta)$. The ratio between the frequency values when maximum reporting is reached is $\approx \ln(1/\delta)/\ln(\varepsilon/\delta)$. Figure 2 shows the expected numbers of reported keys with PWS (Opt) and SbH for frequency distributions that are $\mathsf{Zipf}[\alpha]$ with $\alpha = 1, 2$ as we sweep the privacy parameter δ. Overall we see that PWS gains 20%-300% in the number of keys reported over baseline. Note that as expected, the optimal PWS reports *all* keys when $\delta = 1$ (i.e., no privacy guarantees) but SbH incurs reporting loss. In the full version, we show similar results on two real-world datasets.

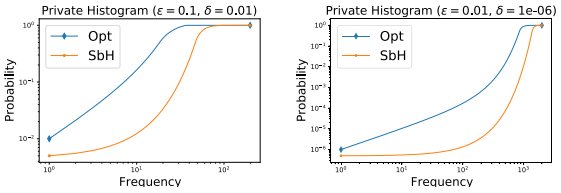

Figure 1: Key reporting probability for frequency. No sampling ($q = 1$) with PWS (Opt) and SbH for $(\varepsilon, \delta) = (0.1, 0.01), (0.01, 10^{-6})$

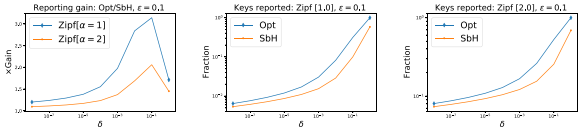

Figure 2: Expected fraction of keys that are privately reported with PWS (Opt) and SbH for $\mathsf{Zipf}[\alpha]$ frequency distributions. For $\alpha = 1, 2$, $\varepsilon = 0.1$ and sweeping δ between 1 and 10^{-8}. Left: The respective ratio of PWS to SbH.

Reporting Probabilities with Sampling Figure 3 shows reporting probabilities with PWS (optimal reporting probabilities), sampled-SbH, and non-private sampling, for representative sampling rates and privacy parameters. As expected, for sufficiently large frequencies both private methods have reporting probabilities that match the sampling probabilities q of the non-private scheme. But PWS reaches q at a lower frequency than sampled-SbH and has significantly higher reporting probabilities for lower frequencies. Figure 4 shows the fraction of keys reported for Zipf distributions as we sweep the sampling rate (threshold τ). PWS reports more keys than sampled-SbH and the gain persists also with low sampling rates. We can see that with PWS, thanks to end-to-end privacy analysis, the reporting loss due to sampling mitigates the reporting loss needed for privacy – reporting approaches that of the non-private sampling when the sampling rate τ approaches δ. Sampled-SbH, on the other hand, incurs reporting loss due to privacy on top of the reporting loss due to sampling.

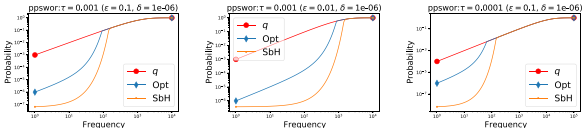

Figure 3: Reporting probability as a function of frequency. For ppswor sampling with threshold τ (q), PWS private samples (Opt), and sampled-SbH private samples.

Estimation of Linear Statistics We evaluate estimation quality for linear statistics (6) when $g(w) = w$ and $L(x)$ is a selection predicate. The statistics is simply the sum of frequencies of selected keys. We compare performance

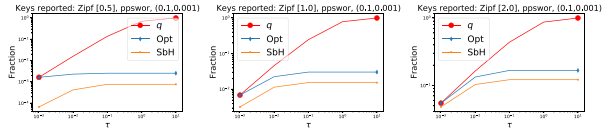

Figure 4: Fraction of total keys reported with threshold-ppswor, sampled-SbH, and PWS (Opt), as we sweep the sampling rate τ. For $\mathsf{Zipf}[\alpha]$, $\varepsilon = 0.1$ and $\delta = 0.001$ the gains in reporting of PWS over Sampled-SbH are at least 230% ($\alpha = 0.5$), 97% ($\alpha = 1$) and 37% ($\alpha = 2$).

of PWS with the MLE estimator (9), the baseline sampled-SbH, and for reference, the estimator of the respective non-private sample (7). Figure 5 (top) shows normalized bias Bias_i/i as a function of the frequency i for the two private methods (the non-private estimator is unbiased and not shown). With both methods, the bias decreases with frequency and diminishes for $i \gg 2\varepsilon^{-1}\ln(1/\delta)$. PWS has lower bias at lower frequencies than SbH, allowing for more accurate estimates on a broader range. We can see that with PWS, the bias decreases when the sampling rate (τ) decreases and diminishes when τ approaches δ. This is a benefit of the end-to-end privacy analysis. The bias of the baseline method does not change with sampling rate.

Figure 6 shows the normalized variance Var_i/i^2 per frequency i for representative parameter settings. The private methods PWS and sampled-SbH maintain low variance across frequencies: The value is fractional with no sampling and is of the order of that of the non-private unbiased estimator with sampling. In particular this means that the bias is a good proxy for performance and that the improvement in bias of PWS with respect to baseline does not come with a hidden cost in variance. For high frequencies (not shown), keys with all methods are included with probability (close to) 1. The non-private method that reports exact frequencies have 0 variance whereas the private methods maintain a low variance, but the normalized variance diminishes for all methods.

For statistics estimation, the per-key performance suggest that when the selection has many high frequency keys, the private methods perform well and are similar to non-private sampling. When the selection is dominated by very low frequencies, the private methods perform poorly and well below the respective non-private sample. But for low to medium frequencies, PWS can provide drastic improvements over SbH and the gain increases with lower sampling rates. Figure 5 (bottom) shows the NRMSE as a function of sampling rate for estimating the sum of frequencies on a selection of 2×10^5 keys with frequencies uniformly drawn between 1 and 200. We can see that the error of non-private sampling and of sampled-SbH decreases with higher sampling rate. Note the perhaps counter-intuitive phenomenon that PWS (MLE) hits its sweet spot midway: This is due to a balance of the two components of the error, the variance which increases and the bias that decreases when the sampling rate decreases. Also note that PWS significantly improves over SbH also with no sampling ($\tau = 1$).

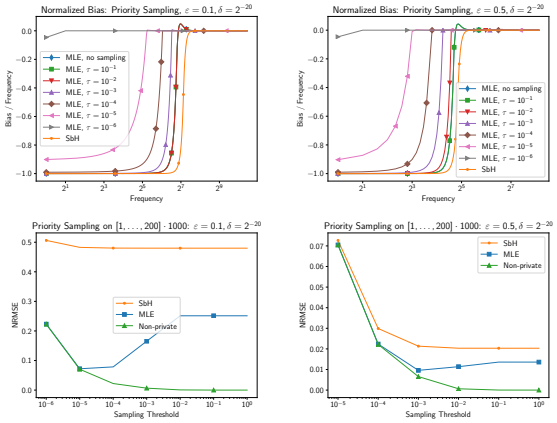

Figure 5: Top: Normalized bias for PWS (MLE) and sampled-SbH as a function of frequency, for different sampling rates. The bias of the sampled-SbH estimates (shown once) does not change with sampling rate. Bottom: NRMSE as a function of sampling rate for a selection of 2×10^5 keys with frequencies drawn uniformly $[1, 200]$.

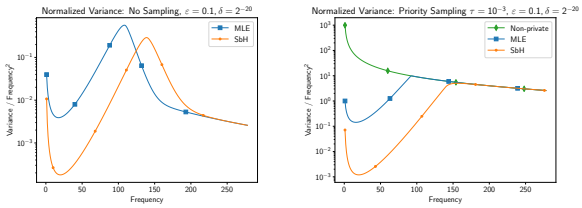

Figure 6: Normalized variance Var_i/i^2 for PWS (MLE) and sampled-SbH as a function of the frequency i.

Conclusion

We presented Private Weighted Sampling (PWS), a method to post-process a weighted sample and produce a version that is differentially private. Our private samples maximize the number of reported keys subject to the privacy constraints and support estimation of linear and ordinal statistics. We demonstrate significant improvement over prior methods for both reporting and estimation tasks, even for the well studied special case of private histograms (when there is no sampling).

An appealing direction for future work is to explore the use of PWS to design *composable* private sketches, e.g., in the context of coordinated samples. Threshold and bottom-k samples of different datasets are *coordinated* when using consistent $\{u_x\}$. Coordinated samples generalize MinHash sketches and support estimation of similarity measures and statistics over multiple datasets (Brewer et al., 1972; Saavedra, 1995; Cohen, 1997; Broder, 2000; Cohen, 2014a,b).

Acknowledgments

Part of this work was done while Ofir Geri was an intern at Google Research. This work was partially supported by Moses Charikar's Google Faculty Research Award.

References

A. Andoni, R. Krauthgamer, and K. Onak. Streaming algorithms via precision sampling. In *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011*, 2011. URL https://doi.org/10.1109/FOCS.2011.82.

V. Balcer and S. P. Vadhan. Differential privacy on finite computers. In A. R. Karlin, editor, *9th Innovations in Theoretical Computer Science Conference, ITCS 2018, January 11-14, 2018, Cambridge, MA, USA*, volume 94 of *LIPIcs*, pages 43:1–43:21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. doi: 10.4230/LIPIcs.ITCS.2018.43. URL https://doi.org/10.4230/LIPIcs.ITCS.2018.43.

A. Beimel, H. Brenner, S. P. Kasiviswanathan, and K. Nissim. Bounds on the sample complexity for private learning and private data release. *Mach. Learn.*, 94(3):401–437, 2014. doi: 10.1007/s10994-013-5404-1. URL https://doi.org/10.1007/s10994-013-5404-1.

A. Beimel, K. Nissim, and U. Stemmer. Private learning and sanitization: Pure vs. approximate differential privacy. *Theory Comput.*, 12(1):1–61, 2016. doi: 10.4086/toc.2016.v012a001. URL https://doi.org/10.4086/toc.2016.v012a001.

K. R. W. Brewer, L. J. Early, and S. F. Joyce. Selecting several samples from a single population. *Australian Journal of Statistics*, 14(3):231–239, 1972.

A. Z. Broder. Identifying and filtering near-duplicate documents. In *Proc.of the 11th Annual Symposium on Combinatorial Pattern Matching*, volume 1848 of *LNCS*, pages 1–10. Springer, 2000.

M. Bun and T. Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In M. Hirt and A. D. Smith, editors, *Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part I*, volume 9985 of *Lecture Notes in Computer Science*, pages 635–658, 2016. doi: 10.1007/978-3-662-53641-4_24. URL https://doi.org/10.1007/978-3-662-53641-4_24.

M. Bun, K. Nissim, U. Stemmer, and S. P. Vadhan. Differentially private release and learning of threshold functions. In V. Guruswami, editor, *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 634–649. IEEE Computer Society, 2015. doi: 10.1109/FOCS.2015.45. URL https://doi.org/10.1109/FOCS.2015.45.

M. Bun, C. Dwork, G. N. Rothblum, and T. Steinke. Composable and versatile privacy via truncated CDP. In I. Diakonikolas, D. Kempe, and M. Henzinger, editors, *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 74–86. ACM, 2018. doi: 10.1145/3188745.3188946. URL https://doi.org/10.1145/3188745.3188946.

M. Bun, K. Nissim, and U. Stemmer. Simultaneous private learning of multiple concepts. *J. Mach. Learn. Res.*, 20:94:1–94:34, 2019. URL http://jmlr.org/papers/v20/18-549.html.

K. Chaudhuri and N. Mishra. When random sampling preserves privacy. In *Proceedings of the 26th Annual International Conference on Advances in Cryptology*, CRYPTO'06, page 198–213, Berlin, Heidelberg, 2006. Springer-Verlag. ISBN 3540374329. doi: 10.1007/11818175_12. URL https://doi.org/10.1007/11818175_12.

E. Cohen. Size-estimation framework with applications to transitive closure and reachability. *J. Comput. System Sci.*, 55:441–453, 1997.

E. Cohen. Distance queries from sampled data: Accurate and efficient. In *KDD*. ACM, 2014a. full version: http://arxiv.org/abs/1203.4903.

E. Cohen. Estimation for monotone sampling: Competitiveness and customization. In *PODC*. ACM, 2014b. URL http://arxiv.org/abs/1212.0243. full version http://arxiv.org/abs/1212.0243.

E. Cohen. Stream sampling framework and application for frequency cap statistics. *ACM Trans. Algorithms*, 14(4):52:1–52:40, 2018. URL https://doi.org/10.1145/3234338. preliminary version published in KDD 2015. arXiv:http://arxiv.org/abs/1502.05955.

E. Cohen and O. Geri. Sampling sketches for concave sublinear functions of frequencies. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

E. Cohen and H. Kaplan. Summarizing data using bottom-k sketches. In *ACM PODC*, 2007.

E. Cohen and H. Kaplan. Tighter estimation using bottom-k sketches. In *Proceedings of the 34th VLDB Conference*, 2008. URL http://arxiv.org/abs/0802.3448.

E. Cohen, G. Cormode, and N. Duffield. Don't let the negatives bring you down: Sampling from streams of signed updates. In *Proc. ACM SIGMETRICS/Performance*, 2012.

E. Cohen, N. Duffield, H. Kaplan, C. Lund, and M. Thorup. Algorithms and estimators for accurate summarization of unaggregated data streams. *J. Comput. System Sci.*, 80, 2014.

E. Cohen, O. Geri, and R. Pagh. Composable sketches for functions of frequencies: Beyond the worst case. In *ICML*, 2020a. URL https://arxiv.org/abs/2004.04772.

E. Cohen, O. Geri, T. Sarlos, and U. Stemmer. Differentially private weighted sampling, 2020b.

E. Cohen, R. Pagh, and D. P. Woodruff. Wor and p's: Sketches for ℓ_p-sampling without replacement. In *NeurIPS*, 2020c. URL https://arxiv.org/abs/2007.06744.

G. Cormode, C. Procopiuc, D. Srivastava, and T. T. L. Tran. Differentially private summaries for sparse data. In *Proceedings of the 15th International Conference on Database Theory*, ICDT '12, page 299–311, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450307918. doi: 10.1145/2274576.2274608. URL https://doi.org/10.1145/2274576.2274608.

D. Desfontaines, J. Voss, B. Gipson, and C. Mandayam. Differentially private partition selection, 2020. URL https://arxiv.org/abs/2006.03684.

N. Duffield, M. Thorup, and C. Lund. Priority sampling for estimating arbitrary subset sums. *J. Assoc. Comput. Mach.*, 54(6), 2007.

C. Dwork, F. McSherry, K. Nissim, and A. D. Smith. Calibrating noise to sensitivity in private data analysis. *J. Priv. Confidentiality*, 7(3):17–51, 2016. doi: 10.29012/jpc.v7i3.405. URL https://doi.org/10.29012/jpc.v7i3.405.

C. Estan and G. Varghese. New directions in traffic measurement and accounting. In *SIGCOMM*. ACM, 2002.

A. Ghosh, T. Roughgarden, and M. Sundararajan. Universally utility-maximizing privacy mechanisms. *SIAM J. Comput.*, 41(6):1673–1693, 2012. URL https://doi.org/10.1137/09076828X.

P. Gibbons and Y. Matias. New sampling-based summary statistics for improving approximate query answers. In *SIGMOD*. ACM, 1998.

S. Gopi, P. Gulhane, J. Kulkarni, J. H. Shen, M. Shokouhi, and S. Yekhanin. Differentially private set union. In *ICML*, 2020. URL https://arxiv.org/abs/2002.09745.

M. Hardt and K. Talwar. On the geometry of differential privacy. In L. J. Schulman, editor, *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, 5-8 June 2010*, pages 705–714. ACM, 2010. doi: 10.1145/1806689.1806786. URL https://doi.org/10.1145/1806689.1806786.

D. G. Horvitz and D. J. Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47(260):663–685, 1952.

R. Jayaram and D. P. Woodruff. Perfect lp sampling in a data stream. In *FOCS*, 2018. URL https://doi.org/10.1109/FOCS.2018.00058.

S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. D. Smith. What can we learn privately? *SIAM J. Comput.*, 40(3):793–826, 2011. doi: 10.1137/090756090. URL https://doi.org/10.1137/090756090.

A. Korolova, K. Kenthapadi, N. Mishra, and A. Ntoulas. Releasing search queries and clicks privately. In J. Quemada, G. León, Y. S. Maarek, and W. Nejdl, editors, *Proceedings of the 18th International Conference on World Wide Web, WWW 2009, Madrid, Spain, April 20-24, 2009*, pages 171–180. ACM, 2009. doi: 10.1145/1526709.1526733. URL https://doi.org/10.1145/1526709.1526733.

E. Ohlsson. Sequential poisson sampling from a business register and its application to the swedish consumer price index. Technical Report 6, Statistics Sweden, 1990.

E. Ohlsson. Sequential poisson sampling. *J. Official Statistics*, 14(2):149–162, 1998.

B. Rosén. Asymptotic theory for successive sampling with varying probabilities without replacement, I. *The Annals of Mathematical Statistics*, 43(2):373–397, 1972. URL http://www.jstor.org/stable/2239977.

B. Rosén. Asymptotic theory for order sampling. *J. Statistical Planning and Inference*, 62(2):135–158, 1997.

P. J. Saavedra. Fixed sample size pps approximations with a permanent random number. In *Proc. of the Section on Survey Research Methods*, pages 697–700, Alexandria, VA, 1995. American Statistical Association.

S. Vadhan. *The Complexity of Differential Privacy*. 04 2017. ISBN 978-3-319-57047-1. doi: 10.1007/978-3-319-57048-8_7.