Histogram Estimation under User-level Privacy with Heterogeneous Data

Yuhan Liu
Cornell University
yl2976@cornell.edu

Ananda Theertha Suresh
Google Research
theertha@google.com

Wennan Zhu
Google Research
wennanzhu@google.com

Peter Kairouz
Google Research
kairouz@google.com

Marco Gruteser
Google Research
gruteser@google.com

June 8, 2022

Abstract

We study the problem of histogram estimation under user-level differential privacy, where the goal is to preserve the privacy of all entries of any single user. While there is abundant literature on this classical problem under the item-level privacy setup where each user contributes only one data point, little has been known for the user-level counterpart. We consider the heterogeneous scenario where both the quantity and distribution of data can be different for each user. We propose an algorithm based on a clipping strategy that almost achieves a two-approximation with respect to the best clipping threshold in hindsight. This result holds without any distribution assumptions on the data. We also prove that the clipping bias can be significantly reduced when the counts are from non-i.i.d. Poisson distributions and show empirically that our debiasing method provides improvements even without such constraints. Experiments on both real and synthetic datasets verify our theoretical findings and demonstrate the effectiveness of our algorithms.

1 Introduction

Differential privacy (DP) [Dwork et al., 2006] provides a rigorous formulation of privacy and has been applied to many algorithmic and learning tasks that involves the access of private and sensitive information. Notable applications include private data release [Hardt et al., 2012], learning histograms [Dwork et al., 2006], statistical estimation [Diakonikolas et al., 2015; Kamath et al., 2019; Acharya et al., 2021; Kamath et al., 2020; Acharya et al., 2019a,b], and machine learning (Chaudhuri et al., 2011; Bassily et al., 2014; McMahan et al., 2018b; Dwork et al., 2014; Abadi et al., 2016).

In real applications, each user may contribute many data samples to a dataset. For example, one may have multiple health records in a hospital, or may type many words on their phone’s virtual keyboard. Naturally, users would hope that all of their private data is protected. To achieve this, private algorithms should guarantee user-level differential privacy.

However, most existing works assume that each user only contributes one data sample. Thus, an algorithm designed under this assumption can only be used to protect the privacy of each data sample but not the user. In other words, such algorithms achieve item-level privacy, but they cannot protect privacy at the user level and may not meet the increasing privacy concerns in most applications where users may contribute a lot of data. Therefore, there has been a growing interest in revisiting differential privacy in the user-level setting.

In this work, we study the problem of estimating histograms and discrete distributions under user-level privacy. Histogram estimation is a fundamental problem that arises in many real-world applications such
as demographic data and user preferences. For example, Chen et al. (2019) used histogram estimation to compute unigram language models via federated learning (McMahan et al., 2017; Kairouz et al., 2019). Beyond machine learning, federated analytics uses histogram estimation to support the Now Playing feature on Google’s Pixel phones, a tool that shows users what song is playing in the room around them (Ramage & Mazzocchi, 2020). There is abundant literature on histogram estimation in the context of item-level privacy, such as (Hay et al., 2010; Suresh, 2019; Xu et al., 2013). Little has been known, however, under user-level privacy.

Some other works (Durfee & Rogers, 2019; Silva Carvalho et al., 2020; Joseph et al., 2021) study the top-k selection problem under user-level differential privacy. However, these works differ from ours in several ways. First note that our goal is to estimate the entire histogram, whereas these works focus on estimating only the most frequent elements and do not focus on estimating the entire histogram. Further, they assume a bounded and known $\ell_0$ or $\ell_\infty$ norm on each user’s histogram. It is unclear how these algorithms can be applied when the bounded and known norm assumption is not satisfied.

User-level privacy is much more stringent than the item-level counterpart. Hence, many challenges arise even in the simple problem of estimating histograms. One challenge arises from the heterogeneity of histogram sizes. Since the number of samples each user contributes may potentially be unbounded, the sensitivity of the algorithm may be prohibitively large, which requires adding a large amount of noise. If we restrict user contribution to achieve low sensitivity, we may lose a large amount of useful data and suffer from bias. This dilemma is known as the bias-variance trade-off which has been discussed in Amin et al. (2019) in the context of empirical risk minimization.

In addition, users’ data may come from diverse distributions, especially in scenarios such as federated learning and analytics (Kairouz et al., 2019). Therefore, we cannot leverage techniques from statistical estimation problems which heavily rely on the i.i.d. assumption (Liu et al., 2020). Algorithms designed specifically for the i.i.d. case may perform poorly on datasets that do not meet the i.i.d. assumption. If each user has just one item, the non-i.i.d. problem can be circumvented by equivalently assuming that each user’s data comes from some common mixture distribution. With multiple samples per user, it can be difficult to find a simple reduction to the i.i.d. case. Motivated by the need for algorithms that work well without any data assumptions, we ask the following question:

*Can we design algorithms for histogram estimation with theoretical guarantees without making any assumptions on user data?*

Somewhat surprisingly, despite many recent works in the area, the above question has not been answered. We take a step towards answering the above question by designing algorithms that perform well in the heterogeneous setting where both the number of samples and data distribution can be unknown and different across users (hence both need to be viewed as private information). We first design an algorithm without any distribution assumptions on user’s data. We then investigate whether we can improve the performance of our algorithm by drawing conclusions from settings where assumptions on the user data distributions are made. Our contributions are as follows.

**Estimation with optimal clipping.** We design an algorithm that estimates the aggregate histogram using an adaptive clipping strategy. Our algorithm provides a 2-approximation of the best clipping threshold in hindsight, taking into account the private estimation of the chosen clipping threshold.

**Bias reduction.** For estimating counts, we design a debiasing algorithm that significantly reduces the clipping bias. Assuming the user counts come from independent Poisson distributions with different means, we provide a complete proof of the gap between the clipped-and-debiased and the clipped-only algorithms. Although designed with a distribution assumption, the debiasing method performs well on both real and synthetic datasets, even when the underlying data generating mechanism is not a Poisson distribution.

The paper is organized as follows. In Section 2, we discuss related work. In Section 3, we introduce the definition and problem formulation. In Section 4, we introduce our algorithms for the most general case with no distribution assumptions. In Section 5, we introduce our debiasing algorithm and its guarantees. In Section 6, we show the experiment results. In Section 7, we conclude with a discussion about how to extend our methods to the unknown-domain and federated settings.
2 Related work

Given its importance, user-level privacy has been studied by several works in the last decade. One of the primary motivations for user-level privacy is federated learning, where the goal is to learn a model at the server while keeping the raw data on edge devices such as cell phones (McMahan et al., 2017; Kairouz et al., 2019). Ensuring privacy at the user level is a crucial concern in federated learning. Even though users do not send their original data, various works (Phong et al., 2017; Wang et al., 2019) have shown it is still possible to reconstruct user’s data if additional privacy-preserving mechanisms are not used. Therefore, user-level privacy has been studied under various machine learning tasks in the federated learning setup (McMahan et al., 2018b,a; Augenstein et al., 2019). Indeed, understanding the fundamental privacy-utility trade-offs under user-level privacy is one of the main challenges in federated learning (Kairouz et al., 2019, Section 4.3.2).

Several works studied fundamental theoretical problems in user-level private learning. Ghazi et al. (2021) studied PAC learnability under user-level privacy. Levy et al. (2021) studied high-dimensional distribution estimation and optimization and designed efficient algorithms. Both works require i.i.d. data and assume a fixed number of samples across users. There are several recent works closely related to user-level private histogram estimation. Amin et al. (2019) studied the inherent bias and variance trade-off in bounding user contributions under user-level privacy for empirical risk minimization. Their analysis applies to estimating the total count of one symbol in the aggregate histogram. We extend their work to the setting of $d > 1$ symbols.

Liu et al. (2020) and Levy et al. (2021) studied a closely related problem of discrete distribution estimation and designed optimal algorithms in terms of user complexity (the minimum number of users required to learn an unknown distribution with given accuracy) up to logarithmic factors. However, their analysis assumes that all users’ data are drawn from nearly identical distributions. Furthermore, the algorithms algorithms in Liu et al. (2020) may be impractical due to time inefficiency and large constants in user complexity.

Esfandiari et al. (2021) studied robust high dimensional mean estimation under user-level privacy, assuming i.i.d. and fixed number of samples across users. While their algorithm is robust to at most 49% of the samples being arbitrarily adversarial, their result still requires that the remaining samples are independent and identically distributed. Cummings et al. (2021) studied mean estimation of Bernoulli random variables, which can be viewed as a special case of histogram estimation when the domain size is 2. They allow different distributions and number of samples for different users. However, the number of samples of each user is known and fixed in advance. Wilson et al. (2020) proposed differentially private SQL with bounded user contributions.

Several works (Durfee & Rogers, 2019; Silva Carvalho et al., 2020) studied top-$k$ selection or frequency estimation in a large alphabet. However, they require bounded $\ell_\infty$ or $\ell_0$ norm on each user’s histogram, and do not explore how that can be (optimally or near optimally) achieved when the requirement is not satisfied. Hence, their algorithms cannot be applied to the general case when the bound is not known beforehand.

3 Problem formulation

3.1 Definitions and notations

Differential privacy (DP) is studied in the central and local settings (Dwork et al., 2006; Kasiviswanathan et al., 2011; Duchi et al., 2013). In this paper, we study the problem under the lens of central differential privacy, where the goal is ensure the algorithm’s outcomes do not reveal too much information about any user’s data. We now define differential privacy, starting with the basic definition of neighboring datasets. Here, we assume the number of users is known and fixed and hence we use the replacement notion of neighboring datasets.

**Definition 3.1.** Let $D = \{X_1, \ldots, X_n\}$ represent a dataset of $n$ users. Each $X_i$ consists of $m_i$ samples $\{X_{i,j}\}_{j=1}^{m_i}$. Let $D' = \{X'_i\}_{i=1}^{n}$ be another dataset. We say $D$ and $D'$ are neighboring (or adjacent) datasets if
for some $i' \in [n],$

\[ X_i = X_i', \text{ for all } i \neq i'. \]

**Definition 3.2** (Differential privacy). A randomized mechanism $M$ with range $\mathcal{R}$ satisfies $(\varepsilon, \delta)$-differential privacy if for any two adjacent datasets $D, D'$ and for any subset of output $S \subseteq \mathcal{R}$, it holds that

\[ Pr[M(D) \in S] \leq e^\varepsilon Pr[M(D') \in S] + \delta. \]

If $\delta = 0$, then the privacy is also referred to as pure DP, and for simplicity we say that the algorithm satisfies $\varepsilon$-DP. If $\delta > 0$, we refer to it as approximate DP.

Let $\| \cdot \|_q$ denote the $\ell_q$ norm. For a vector $x \in \mathbb{R}^d$ and $C \in \mathbb{R}^+$, define the $\ell_q$ clipping function,

\[ \text{clip}_q(x, C) = C \cdot \frac{x}{\max(C, \|x\|_q)}. \]

### 3.2 Histogram estimation

We consider the following problem. There are $n$ users and user $i$ has a histogram $N_i = (N_{i,1}, \ldots, N_{i,d}) \in \mathbb{Z}_{\geq 0}^d$ over a discrete domain of size $d \in \mathbb{N}$. Without loss of generality, we can assume the domain to be $[d] := \{1, \ldots, d\}$. Let $m_i = \|N_i\|_1$ be the size of histogram $N_i$. The dataset $D$ is the collection of users’ histograms. The goal is to estimate some query subject to user-level differential privacy. Neighboring datasets differ by a single user (i.e., one histogram).

**Estimation with optimal clipping.** We first consider problem of estimating the population-level histogram, i.e., the sum of the histograms

\[ \bar{N}(D) = \sum_{i=1}^{n} N_i. \]

We make no assumptions about the distribution and size of each user’s histogram. Given an $(\varepsilon, \delta)$ differentially private algorithm whose output histogram is $\hat{N}$, we characterize its performance by the expected $\ell_1$ distance between the algorithm output and the true population-level histogram

\[ \|\bar{N} - \hat{N}\|_1 = \sum_{j=1}^{d} |\bar{N}_j - \hat{N}_j|. \]

**Bias reduction.** We prove that the bias from clipping can be significantly reduced when $N_i$’s satisfy some mild distribution assumptions and show that the debiasing method provides improvements even on real datasets where these assumptions may not necessarily hold. Consider the special case of $d = 1$, which we refer to as count estimation. Let $\mathcal{D}$ be a family of distributions over $\mathbb{Z}_{\geq 0}$. For each user $i$, $N_i$ is drawn independently from some distribution in $\mathcal{D}$ with mean $\lambda_i > 0$. $\lambda_i$’s can be arbitrary and do not need to be equal.

In addition to the absolute error of counts $|\bar{N} - \hat{N}|$, we also want to characterize the accuracy for estimating the mean $\bar{\lambda} = \frac{1}{n} \sum_{i=1}^{n} \lambda_i$. Let $\hat{\lambda} = \bar{N}/n$ be an estimate of $\bar{\lambda}$. We are interested in the expected square error

\[ \mathbb{E}[(\bar{\lambda} - \hat{\lambda})^2], \]

where the expectation is over the randomness of the algorithm and the dataset.

In this work we set $\mathcal{D}$ to be the family of Poisson distributions since they arise in many applications. For example, they can be used to model the occurrences of a memoryless event in a fixed time window. Also, they are good approximations of the binomial distribution $\text{Bin}(m, p)$ when $mp$ is a constant (Le Cam [1960]), and can be very useful when estimating the count of one element in a histogram over a very large domain (e.g. the count of a particular word).
Algorithm 1: Clipping algorithm

Input: histograms \(N_1, \ldots, N_n\), clip threshold \(C\), privacy parameter \(\varepsilon, \delta\).

Clipping: for each user \(i\), do

\[ \tilde{N}_i^L = \text{clip}_1(N_i, C) \quad \text{or} \quad \tilde{N}_i^G = \text{clip}_2(N_i, C) \]

Add noise

\[ \hat{N}_L = \sum_i \tilde{N}_i + \text{Lap}(C/\varepsilon) \quad \delta = 0, \]

\[ \hat{N}_G = \sum_i \tilde{N}_i + N(0, 2C \log(1.32/\delta)/\varepsilon) \quad \delta > 0. \]

Return \(\hat{N}_L\) or \(\hat{N}_G\).

4 Estimation with optimal clipping

We first consider the problem of estimating the population-level histogram. A standard strategy is to clip each user contribution either in \(\ell_1\) or \(\ell_2\) norm and add the corresponding amount of noise. For completeness, the details are shown in Algorithm 1.

There is a bias-variance trade-off in choosing the clipping threshold \(C\). If \(C\) is small, then the noise magnitude (or variance) is small, but the clipped histogram would have large error (or bias). On the other hand, if \(C\) is large, the clipped histogram would be more accurate (less bias), but the added noise would be large (high variance).

The easiest choice of \(C\) is to choose a fixed value, or some fixed quantile of the \(\ell_1\) or \(\ell_2\) norms of all histograms (which can be estimated privately). However, Amin et al. (2019) showed that even in the 1-d case, such strategy cannot perform well in general. We provide an accurate characterization of the best threshold that balances the bias and variance. For the Laplace estimator, the proof is similar to that of Amin et al. (2019) and is omitted.

Lemma 4.1. Let \(L_L(C, D) = E[\|\hat{N}_L - \bar{N}(D)\|_1]\). Choosing \(C^*\) as the top \([d/\varepsilon]\) element in \(\{m_i\}_{i=1}^n\) yields 2-approximation

\[ L_L(C^*, D) \leq 2 \inf_{C \geq 0} L_L(C, D), \]

where the expectation is over the Laplace mechanism.

We now state the result for the Gaussian mechanism in Theorem 4.2. The complete proof is in Appendix A.1

Theorem 4.2. Let \(L_G(C, D) = E[\|\hat{N}_G - \bar{N}(D)\|_1]\). Let \(\varepsilon \leq 1\) and \(M = 2\sqrt{\frac{\log(1.32/\delta)}{\pi}} \frac{d}{\varepsilon}\). Choosing \(C^*\) such that

\[ C^* = \arg \min_{C \geq 0} \left\{ \sum_{i: \|N_i\|_2 > C} \frac{\|N_i\|_1}{\|N_i\|_2} \leq M \right\} \]

yields 2-approximation,

\[ L_G(C^*, D) \leq 2 \inf_{C \geq 0} L_G(C, D). \]

Next we discuss how to find \(C^*\) privately using an additional privacy budget of \((\varepsilon', \delta')\), and further provide complete guarantees in terms of excess error compared to \(2 \inf_{C \geq 0} L_G(C, D)\). We emphasize that it only
requires a very small extra privacy budget compared to the original \((\varepsilon, \delta)\) to achieve good performance, as we will later show in the experiments. We focus on \((\varepsilon, \delta)\)-DP because it is more practical, especially when \(d\) is large.

Note that computing the optimal \(C\) in a differentially private way is possible though difficult because the sensitivity of \(\sum_{i \in N_i} \|N_i\|_1 \|N_i\|_2\) can be very large for some datasets. However, observe that by Cauchy-Schwarz inequality,
\[
\|N_i\|_1 / \|N_i\|_2 \leq \sqrt{\|N_i\|_0}.
\]
Hence, if each user’s histogram has very few non-zero entries, then the sensitivity would be low. We observe this to be the case in practice. To illustrate, we plot the distinct number symbols each user has for the Sentiment140 (Go et al., 2009) and SNAP datasets in Figure 1. Observe that most users have fewer than 200 samples, even though \(d\) is large. Hence, we assume that each user’s histogram is at most \(s\) sparse. Under this assumption, the sensitivity is upper bounded by \(\sqrt{s}\).

We first note that \(C^*\) can be written as a minimizer to a convex function,
\[
G(C) = \sum_{i=1}^{n} f_i(C) + CM,
\]
where \(f_i(C) = \max\{1 - \frac{C \|N_i\|_1}{\|N_i\|_2}, 0\} \|N_i\|_1\). Hence we can use techniques from differentially private convex optimization algorithms. We consider two such algorithms and provide their corresponding guarantees.

**Choosing \(C^*\) with DP-SGD** We first consider the DP-SGD algorithm (Bassily et al., 2014, Algorithm 1) to estimate \(C^*\) by minimizing \(G(C)\). Using Bassily et al. (2014, Theorem 2.4), we have the following guarantee.

**Corollary 4.3.** Let \(C_m\) be an upper bound on \(C^*\) and let \(\hat{C}\) be the output of Bassily et al. (2014, Algorithm 1). Assume that \(n \geq d\). Then,
\[
\mathbb{E}[\mathcal{L}_G(\hat{C}, D)] - 2 \inf_{C \geq 0} \mathcal{L}_G(C, D) \leq O\left(\frac{C_m \left(\sqrt{s} + \frac{\sqrt{\log(1/\delta')}}{\varepsilon'}\right)}{\varepsilon'} \log^{3/2}(n/\varepsilon') \sqrt{\log(1/\delta')}\right).
\]

**Choosing \(C^*\) with output perturbation** We consider the second algorithm based on output perturbation (Chaudhuri et al., 2011), which ensures \((\varepsilon', 0)\)-DP and is good for small \(n\) and \(\varepsilon'\). Here, we solve a regularized convex optimization problem and perturb the output to provide differential privacy. The algorithm is outlined in **Algorithm 2**.

**Algorithm 2: Clipping threshold estimation with output perturbation**

Input: histograms \(N_1, \ldots, N_n\), an approximation of \(C^*\) denoted by \(C_m\), sparsity parameter \(s\), privacy parameter \(\varepsilon'\).

Let \(\lambda = \frac{2\sqrt{2}}{C_m \sqrt{n \varepsilon'}}\) and \(\Delta = \frac{4\sqrt{2}}{\lambda n}\) Compute \(C'\), the minimizer of \(F(C) = \frac{1}{n} G(C) + \frac{1}{2} C^2\).

Return \(\hat{C} = C' + \text{Lap}(\Delta/\varepsilon')\).

With appropriate parameters, the combined algorithm almost achieves a 2-approximation with respect to the best clipping threshold.

**Corollary 4.4.** Algorithm 2 is \((\varepsilon', 0)\) differentially private. If \(C_m\) is an upper bound on \(C^*\), setting \(\lambda = \frac{2\sqrt{2}}{C_m \sqrt{n \varepsilon'}}\) yields an error
\[
\mathbb{E}[\mathcal{L}_G(\hat{C}, D)] \leq 2 \inf_{C \geq 0} \mathcal{L}_G(C, D) + 2\sqrt{2} C_m \sqrt{\frac{ns}{\varepsilon'}}.
\]
Comparing DP-SGD and Algorithm 4, we can see that DP-SGD has a better asymptotic dependence on $n$, and Algorithm 7 has a better dependence on $\epsilon'$. Furthermore, DP-SGD provides an approximate DP guarantee and Algorithm 9 gives a pure DP guarantee. Finally, the time complexity of DP-SGD is typically $O(n^2)$, however it has been improved recently to $O(n)$ with similar guarantees.

5 Bias reduction

In this section, we ask if the estimate obtained by Algorithm 4 can be improved by making assumptions on the histogram generating distribution. We prove that the bias from clipping can be significantly reduced when the user counts come from non-i.i.d. Poisson distributions and in Section 6, we show that the debiasing method provides improvements even when the counts do not come from Poisson distributions.

Our debiasing step is a post-processing procedure on the output of Algorithm 1 to reduce the clipping bias. Since this is a postprocessing step, it does not affect the differential privacy guarantees. In the rest of the section, we assume $d = 1$. Towards the end of the section, we discuss two ways of extending it to high dimensions and overview the pros and cons of both methods.

Algorithm 3: General debiasing algorithm for single item

Input: $N_1, \ldots, N_n$. 

$Y_i = f(N_i)$

$\hat{N} = g(\sum_{i=1}^N Y_i + \text{Lap}(\Delta f/\varepsilon))$

A general algorithm for estimating the count of a single item is given in Algorithm 3, which involves a preprocessing step on each user’s count denoted by $f$ and a postprocessing step on the differentially private output, denoted by $g$. If we choose $f(x) = \text{clip}(x, C) := \min\{x, C\}$ and $g(x) = x$, we recover the clipping algorithm.

We ask the following question, is it possible to find a post-processing function $g$ that gives better performance than the clipping algorithm even if the data generating distributions are distinct? It is not hard to see that if all users come from the same distribution i.e., $\lambda_i = \lambda$ for all $i$, then such an algorithm would help. Our main contribution is to show that one can design an $h$ such that even if the users’ data come from different distributions, it can be debiased and the performance can be improved.

We first note that in many datasets, counts of most symbols appear very few times. For example, in the Sentiment140 dataset, which contains counts for a total of roughly $6 \cdot 10^5$ words distributed across $6 \cdot 10^5$ users, the average counts of all words among the users are no more than two. Therefore we analyze the debiasing step when the $\lambda_i’s$ are small and prove the following result for our debiasing algorithm given in Algorithm 4.

Theorem 5.1. Suppose $N_i \sim \text{Poi}(\lambda_i)$. Let $\hat{\lambda} = \frac{1}{n} \sum\lambda_i$, $\Sigma = \frac{1}{n} \sum_{i=1}^n (\lambda_i - \hat{\lambda})^2$ and $\hat{\lambda} = \min\{1, \max\{0, N/n\}\}$, where $\hat{N}$ is the output of Algorithm 4. If $\lambda \leq 1$, then

$$\mathbb{E}[(\hat{\lambda} - \lambda)^2] \leq \gamma_C^2 \left( \frac{C^2}{n^2 \varepsilon^2} + \frac{\hat{\lambda}}{n} + \min\left\{ 1, \frac{1}{8\pi(C - 1)} \right\} \Sigma^2 \right).$$ (1)

where $\gamma_C = \text{Pr}[\text{Poi}(1) < C]^{-1} \leq \max\left\{ e, \frac{1}{1 - e^{-1/(C - 1)^2}} \right\}$. If we further assume that $\lambda_i \leq 1$, then

$$\mathbb{E}[(\bar{\lambda} - \lambda)^2] \leq \gamma_C^2 \left( \frac{C^2}{n^2 \varepsilon^2} + \frac{\bar{\lambda}}{n} + \frac{1}{4((C - 1))^2} \cdot \Sigma^2 \right).$$ (2)

The error consists of three terms. The first term is the error due to added noise, which is proportional to the clipping threshold $C$. The second term is essentially the variance of the random variable $\frac{1}{n} \sum\lambda_i$, which is an inherent error due to the randomness in the counts $N_i’s$. The third term is a bias term which depends on the closeness of user distributions, characterized by $\Sigma$, the variance of $\lambda_i's$. If the users’ distributions are similar, then we can expect the estimation error to be small. Note that the bias term has a $1/C$ rate.
The detailed proof is provided in Appendix A.3. We provide a bound with a better dependence on $C$ in Appendix A.4.

Next we analyze the the optimal choice of threshold $C$ after the debiasing step. Observe that to minimize the error upper bound in (1), roughly we want to choose

$$C \propto (n \Sigma)^{2/3} + 1.$$  

Therefore, when user’s distribution are similar, we can use a smaller clipping threshold. This implies that as long as

$$\Sigma = O(n^{-1/4}),$$

we can find a $C$ that ensures a squared error of $O(1/n)$, which matches the error for the i.i.d. case.

From (2), when $\lambda_i \leq 1$ for any user $i$, we can choose

$$C \propto 1 + \log(1 + n \Sigma).$$

This choice of $C$ always guarantees $O(1/n)$ error since when $\lambda_i \leq 1$ for all $i$, $\Sigma \leq 1$.

In practice, we can also privately choose $C$ as the top $\lceil 1/\varepsilon \rceil$ count as suggested by Lemma 4.1.

So far we have characterized the effect of debiasing under heterogeneous data. We next show that under mild assumptions debiasing helps even if data is non i.i.d.. The formal result is stated in Theorem 5.2. The proof is in Appendix A.5.

**Theorem 5.2.** Let $h = \frac{1}{n} \sum_{i=1}^{n} h(\lambda_i)$. Let $\hat{\lambda}_L = \hat{N}_L/n$ be the average count obtained by Algorithm 4 with Laplace noise. Assume that $\hat{h} \geq h_{\text{min}} := h(\lambda) - \frac{\lambda - h(\lambda)}{\gamma C}$. Write $\hat{h} = h_{\text{min}} + \alpha (\hat{\lambda} - h(\lambda))$ where $\alpha \in (0, 1]$. Then the gap between Algorithm 4 and Algorithm 1 is

$$\mathbb{E}[(\bar{\lambda} - \hat{\lambda}_L)^2] - \mathbb{E}[(\bar{\lambda} - \hat{\lambda})^2] \geq \frac{(2\gamma C - (\gamma C + 1)\alpha)}{\gamma C - 1} (\bar{\lambda} - h(\lambda))^2 - O_C \left( \frac{1}{n} \right).$$  

(3)

This implies that for any fixed $C$, under the assumptions stated in the theorem, with $n$ sufficiently large, there is always a constant gap between the two algorithms and debiasing helps even if the data is not i.i.d.. This result justifies the choice of $C$ as the optimal quantile suggested by Lemma 4.1.

We argue that the assumption of $\hat{h} \geq h_{\text{min}}$ is not too restrictive. It essentially requires that either $\lambda_i$’s are sufficiently similar, or $C$ is sufficiently larger than $\bar{\lambda}$. Indeed, if $\lambda_i = \bar{\lambda}$ for all user $i$, then $h(\bar{\lambda}) = \hat{h}$; if $C$ is sufficiently large, then $\hat{h}$ is almost linear near $\bar{\lambda}$ and hence $\hat{h}$ is close to $h(\bar{\lambda})$.

As a specific example, set $C = 2, \bar{\lambda} = 1$. If all $\lambda_i \in [0, 2]$, due to concavity of $h$, we have $\hat{h} \geq h(2)/2 \geq 0.729$. With some arithmetic, $h_{\text{min}} \leq 0.61$, and the first term in (3) is at least 0.0217. Note that this is the difference between squared errors. The gap between absolute errors could well be of order 0.1, which is significant considering that $\bar{\lambda} = 1$. This example shows that Algorithm 4 can achieve significant improvement even when the variance of $\lambda_i$’s is constant.

**Extension to $d > 1$.** We now discuss two possible extensions to the general $d$.

1. A natural extension to the entire histogram is to apply Algorithm 4 to each symbol in the histogram separately. To ensure $(\varepsilon, \delta)$ differential privacy, we assign each symbol a privacy budget of $O(\varepsilon/\sqrt{d \log(1/\delta)})$ by strong composition [Kairouz et al. 2017]. The main disadvantage is that when $d$ is large, clipping each coordinate separately may perform poorly compared to clipping the $\ell_1$ or $\ell_2$ norm of the entire histogram.

2. We can generalize Algorithm 4 to $d > 1$ by replacing 1-d clipping with the high dimensional clipping functions as defined in Algorithm 1. Then choose a suitable function $g$ that essentially inverts the expectation of the clipped vector $Y_i$. However finding such inverse may be difficult in high dimensions as it likely involves non-convex optimization.

6 Experiments

We run experiments with two datasets: Sentiment140 [Go et al. 2009], a twitter dataset that contains user tweets, and SNAP [Cho et al. 2011], a social network dataset that contains the location information of
check-ins by users. For Sentiment140, we parse each user’s tweets to words, and treat each word as an element. For SNAP, each element is a location, and each user has check-ins to multiple locations in the dataset.

In all experiments, the privacy budget for estimating $C$ is $\varepsilon = 0.1, \delta = 1/2n$, and the budget for Algorithm 1 is $\varepsilon = 1, \delta = 1/2n$. For DP-SGD with sparsity assumptions, we set $s = 0.1d$ and clip each $\|N_i\|_1/\|N_i\|_2$ to $\sqrt{s}$ when estimating $C^*$. This introduces bias when the assumption is not satisfied for some users. However if the percentage of such users is small, this effect can be negligible.

### 6.1 Estimation with optimal clipping

To conduct experiments on real life datasets, we calculate the top $d$ elements in the datasets, and only run experiments on these elements. We measure the relative loss of $\hat{N}$, defined as:

$$\sum_{j=1}^{d} \frac{|\bar{N}_j - \hat{N}_j|}{\|\bar{N}\|_1}.$$  

We evaluate different algorithms for estimating the clipping threshold $C$ for the Gaussian mechanism given in Algorithm 1. We compare the performance of the following methods: (a) $C^*$: The non-private clipping threshold given in Theorem 4.2. (b) DP-M-quantile: inspired by the 2-approximation quantile in [Amin et al., 2019], we set $C$ to be the $M^{th}$ largest value of $m_i$, where $M$ is given in Theorem 4.2. We estimate it by gradient descent with differential privacy, e.g. Andrew et al. (2021, Section 2). (c) $C_{DPSGD}$: estimation of $C^*$ with DPSGD algorithm (Corollary 4.3). and (d) $C_{output}$: estimation of $C^*$ with output pertubation (Algorithm 2).

In Figure 2, we show the comparison of these threshold estimation algorithms with different choices of $d$ in [100, 5000]. The results with both datasets are similar, but SNAP has much higher errors, possibly because of the location information in SNAP is more non-i.i.d compared to the words in Sentiment140. Setting $C$ to $DP-M$-quantile inspired by the 2-approximation quantile in [Amin et al., 2019], without any theoretical support, and the errors are relatively high. For Algorithm 2, we run experiments with $C_m = DP-M$-quantile, and a fixed $C_m = 150$ (see the Supplementary materials). Of all the algorithms, $C_{DPSGD}$ has similar performance to the true $C^*$ without differential privacy.

### 6.2 Bias reduction

In this section, we run experiments for Algorithm 3 with both synthetic datasets and words in Sentiment140. For the synthetic part, we generate $n = 10^6$ users with $\lambda_1, \ldots, \lambda_n$ from a Dirichlet distribution with parameter $\alpha$. Larger $\alpha$ means that the $\lambda_i$s are closer. In this experiment, we set $C$ to the privately estimated top $1/\varepsilon$ count among users, as discussed in [Amin et al., 2019]. Figure 3 shows that the debiased output of

---

1The choice $s = 0.1d$ is arbitrary (i.e. not a function of the underlying datasets) and has not been tuned.
Figure 3: Total counts estimation for single item on synthetic Poisson datasets. Larger $\alpha$ means that user data distributions are more similar.

| word | clipping avg loss±std | debiasing avg loss±std |
|------|----------------------|-----------------------|
| the  | 0.0289±2.40⋅10^{-5}  | 0.0257±2.48⋅10^{-5}   |
| today| 0.0381±4.34⋅10^{-5}  | 0.0155±3.34⋅10^{-5}   |
| you  | 0.0745±2.83⋅10^{-5}  | 0.0671±7.26⋅10^{-5}   |

Algorithm 4 greatly reduces the error compared to the original output of Algorithm 1, especially for large $\alpha$ (meaning the dataset is more i.i.d. like).

We also run experiments on three population words in Sentiment140: “the”, “today” and “you”. Table 1 shows that Algorithm 4 performs better than Algorithm 1, but the gain is not as much as synthetic datasets that are close to i.i.d. distributions.

7 Conclusion and discussion

We study the problem of estimating a population-level histogram under user-level privacy with heterogeneous data. We extend previous works by relaxing i.i.d. assumptions on the user data distribution and allowing for a different number of samples per user. We propose algorithms based on adaptive clipping without any distribution assumptions. We further show that the bias of clipping can be reduced under the special case of count estimation. Future directions include: (a) extending the bias reduction algorithm to high dimensional histograms and proving theoretical lower bounds, (b) designing similar clipping and debiasing strategies for histogram estimation over unknown domains, and (c) applying these methods to the federated setting. With respect to (c), we note that Algorithm 2 can be federated by having users only send the $\ell_1$ and $\ell_2$ norms of their local histograms to the server across rounds. This allows the server to run the output perturbation algorithm and (privately) learn the near-optimal clipping threshold. Combining this approach with secure aggregation (Bonawitz et al., 2017) and a distributed DP (Kairouz et al., 2021; Agarwal et al., 2021) guarantee is left for future work.
References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 308–318, 2016.

Acharya, J., Canonne, C. L., and Tyagi, H. Inference under information constraints: Lower bounds from chi-square contraction. Proceedings of Machine Learning Research vol, 99:1–15, 2019a.

Acharya, J., Sun, Z., and Zhang, H. Hadamard response: Estimating distributions privately, efficiently, and with little communication. In The 22nd International Conference on Artificial Intelligence and Statistics, pp. 1120–1129, 2019b.

Acharya, J., Sun, Z., and Zhang, H. Differentially private assouad, fano, and le cam. In Algorithmic Learning Theory, pp. 48–78. PMLR, 2021.

Agarwal, N., Kairouz, P., and Liu, Z. The skellam mechanism for differentially private federated learning. Advances in Neural Information Processing Systems, 34, 2021.

Amin, K., Kulesza, A., Munoz, A., and Vassilvtiskii, S. Bounding user contributions: A bias-variance trade-off in differential privacy. In International Conference on Machine Learning, pp. 263–271. PMLR, 2019.

Andrew, G., Thakkar, O., McMahan, H. B., and Ramaswamy, S. Differentially private learning with adaptive clipping. In Advances in Neural Information Processing Systems, 2021. URL https://openreview.net/forum?id=RUQ1zwZRS.

Augenstein, S., McMahan, H. B., Ramage, D., Ramaswamy, S., Kairouz, P., Chen, M., Mathews, R., and Arcas, B. A. Generative models for effective ml on private, decentralized datasets. In International Conference on Learning Representations, 2019.

Bassily, R., Smith, A., and Thakurta, A. Private empirical risk minimization: Efficient algorithms and tight error bounds. In 2014 IEEE 55th Annual Symposium on Foundations of Computer Science, pp. 464–473. IEEE, 2014.

Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., Ramage, D., Segal, A., and Seth, K. Practical secure aggregation for privacy-preserving machine learning. In proceedings of the 2017 ACM Conference on Computer and Communications Security, pp. 1175–1191, 2017.

Chaudhuri, K., Monteleoni, C., and Sarwate, A. D. Differentially private empirical risk minimization. Journal of Machine Learning Research, 12(Mar):1069–1109, 2011.

Chen, M., Suresh, A. T., Mathews, R., Wong, A., Allauzen, C., Beaufays, F., and Riley, M. Federated learning of n-gram language models. arXiv preprint arXiv:1910.03432, 2019.

Cho, E., Myers, S. A., and Leskovec, J. Friendship and mobility: user movement in location-based social networks. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 1082–1090, 2011.

Cummings, R., Feldman, V., McMillan, A., and Talwar, K. Mean estimation with user-level privacy under data heterogeneity. In NeurIPS 2021 Workshop Privacy in Machine Learning, 2021.

Diakonikolas, I., Hardt, M., and Schmidt, L. Differentially private learning of structured discrete distributions. In Advances in Neural Information Processing Systems 28, NIPS ’15, pp. 2566–2574. Curran Associates, Inc., 2015.

Duchi, J. C., Jordan, M. I., and Wainwright, M. J. Local privacy and statistical minimax rates. In 2013 IEEE 54th Annual Symposium on Foundations of Computer Science, pp. 429–438. IEEE, 2013.
Durfee, D. and Rogers, R. M. Practical differentially private top-k selection with pay-what-you-get composition. In Wallach, H., Larochelle, H., Beygelzimer, A., d’Alché-Buc, F., Fox, E., and Garnett, R. (eds.), Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/b139e104214a08ae3f2ebcce149cddf6e-Paper.pdf

Dwork, C., McSherry, F., Nissim, K., and Smith, A. Calibrating noise to sensitivity in private data analysis. In Theory of cryptography conference, pp. 265–284. Springer, 2006.

Dwork, C., Talwar, K., Thakurta, A., and Zhang, L. Analyze Gauss: Optimal bounds for privacy-preserving principal component analysis. In Proceedings of the 46th Annual ACM Symposium on the Theory of Computing, STOC ’14, pp. 11–20, New York, NY, USA, 2014. ACM.

Esfandiari, H., Mirrokni, V., and Narayanan, S. Tight and robust private mean estimation with few users. arXiv preprint arXiv:2110.11876, 2021.

Ghazi, B., Kumar, R., and Manurangsi, P. User-level differentially private learning via correlated sampling. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), Advances in Neural Information Processing Systems, 2021. URL https://openreview.net/forum?id=PqiCvohYSAx

Go, A., Bhayani, R., and Huang, L. Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, 1(12), 2009.

Hardt, M., Ligett, K., and McSherry, F. A simple and algorithm for differentially private data release. In Proceedings of the 25th International Conference on Neural Information Processing Systems-Volume 2, pp. 2339–2347, 2012.

Hay, M., Rastogi, V., Miklau, G., and Suciu, D. Boosting the accuracy of differentially private histograms through consistency. Proceedings of the VLDB Endowment, 3(1-2):1021–1032, 2010.

Joseph, M., Gillenwater, J., Ribero, M., et al. A joint exponential mechanism for differentially private top-k set. In NeurIPS 2021 Workshop Privacy in Machine Learning, 2021.

Kairouz, P., Oh, S., and Viswanath, P. The composition theorem for differential privacy. IEEE Transactions on Information Theory, 63(6):4037–4049, 2017.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al. Advances and open problems in federated learning. arXiv preprint arXiv:1912.04477, 2019.

Kairouz, P., Liu, Z., and Steinke, T. The distributed discrete gaussian mechanism for federated learning with secure aggregation. In International Conference on Machine Learning, pp. 5201–5212. PMLR, 2021.

Kamath, G., Li, J., Singhal, V., and Ullman, J. Privately learning high-dimensional distributions. In Proceedings of the 32nd Annual Conference on Learning Theory, 2019.

Kamath, G., Singhal, V., and Ullman, J. Private mean estimation of heavy-tailed distributions. In Conference on Learning Theory, pp. 2204–2235. PMLR, 2020.

Kasiviswanathan, S. P., Lee, H. K., Nissim, K., Raskhodnikova, S., and Smith, A. What can we learn privately? SIAM Journal on Computing, 40(3):793–826, 2011.

Le Cam, L. An approximation theorem for the poisson binomial distribution. Pacific Journal of Mathematics, 10(4):1181–1197, 1960.

Levy, D., Sun, Z., Amin, K., Kale, S., Kulesza, A., Mohri, M., and Suresh, A. T. Learning with user-level privacy. In Advances in Neural Information Processing Systems, 2021.
Liu, Y., Suresh, A. T., Yu, F. X. X., Kumar, S., and Riley, M. Learning discrete distributions: user vs item-level privacy. In *Advances in Neural Information Processing Systems*, volume 33, pp. 20965–20976, 2020.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pp. 1273–1282. PMLR, 2017.

McMahan, B., Andrew, G., Mironov, I., Papernot, N., Kairouz, P., Chien, S., and Úlfar Erlingsson. A general approach to adding differential privacy to iterative training procedures. 2018a. URL https://arxiv.org/pdf/1812.06210.pdf. Workshop on Privacy Preserving Machine Learning (NeurIPS 2018).

McMahan, H. B., Ramage, D., Talwar, K., and Zhang, L. Learning differentially private recurrent language models. In *International Conference on Learning Representations*, 2018b.

Phong, L. T., Aono, Y., Hayashi, T., Wang, L., and Moriai, S. Privacy-preserving deep learning: Revisited and enhanced. In *International Conference on Applications and Techniques in Information Security*, pp. 100–110. Springer, 2017.

Ramage, D. and Mazzocchi, S. Federated analytics: Collaborative data science without data collection. https://ai.googleblog.com/2020/05/federated-analytics-collaborative-data.html, 2020.

Silva Carvalho, R., Wang, K., Gondara, L., and Miao, C. Differentially private top-k selection via stability on unknown domain. In Peters, J. and Sontag, D. (eds.), *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 124 of *Proceedings of Machine Learning Research*, pp. 1109–1118. PMLR, 03–06 Aug 2020. URL https://proceedings.mlr.press/v124/silva-carvalho20a.html

Suresh, A. T. Differentially private anonymized histograms. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Wang, Z., Song, M., Zhang, Z., Song, Y., Wang, Q., and Qi, H. Beyond inferring class representatives: User-level privacy leakage from federated learning. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pp. 2512–2520. IEEE, 2019.

Wilson, R. J., Zhang, C. Y., Lam, W., Desfontaines, D., Simmons-Marengo, D., and Gipson, B. Differentially private sql with bounded user contribution. *Proceedings on Privacy Enhancing Technologies*, 2:230–250, 2020.

Xu, J., Zhang, Z., Xiao, X., Yang, Y., Yu, G., and Winslett, M. Differentially private histogram publication. *The VLDB Journal*, 22(6):797–822, 2013.
A Detailed proofs of lemmas and theorems

A.1 Proof of Theorem 4.2

Proof. Recall that $M = 2\sqrt{\frac{\log(1.32/d)}{\varepsilon}}$. We can upper bound the error as follows.

$$
\mathbb{E}[||\hat{N} - \bar{N}||_1] = \mathbb{E}
\left[
\left\|
\sum_{i=1}^{n} \text{clip}_2(N_i, C) + \mathcal{N}(0, I\sigma^2) - \sum_{i=1}^{n} N_i
\right\|_1
\right]
\leq \mathbb{E}
\left[
\left\|
\sum_{i} \text{clip}_2(N_i, C)
\right\|_1 + \mathbb{E}[||\mathcal{N}(0, I\sigma^2)||_1]
\right]
= \sum_{i: ||N_i||_2 > C} \left(1 - \frac{C}{||N_i||_2}\right) ||N_i||_1 + C \cdot M
= \sum_{i=1}^{n} \max \left\{1 - \frac{C}{||N_i||_2}, 0\right\} ||N_i||_1 + C \cdot M = G(C). \tag{4}
$$

Equation (4) is convex with respect to $C$. To optimize the upper bound on the error, we will take the sub-derivative with respect to $C$ and set it to zero. This gives us the following equation

$$
\sum_{i: ||N_i||_2 > C} \frac{||N_i||_1}{||N_i||_2} = M. \tag{5}
$$

Roughly we want to choose $C$ that satisfies the above equality. The precise value of $C^*$ is

$$
C^* = \arg \min_{C \geq 0} \left\{ \sum_{i: ||N_i||_2 > C} \frac{||N_i||_1}{||N_i||_2} \leq M \right\}
$$

$C^*$ minimizes the right hand side of (4), and it also makes the expected $\ell_1$ loss at most twice the loss of the optimal loss with this algorithm. Formally, suppose $Q$ is the $\ell_2$-norm that minimizes $\mathbb{E}[||\bar{N} - \hat{N}||_1]$. Let $Z = [Z_1, \ldots, Z_d] \sim \mathcal{N}(0, I\sigma^2) \text{ and } \text{clip}_2(N_i, Q)_j \text{ be the } j \text{ coordinate of } \text{clip}_2(N_i, Q), \text{ then:}$

$$
\mathbb{E}
\left[
\left\|
\sum_{i} \text{clip}_2(N_i, Q) + Z - \sum_{i} N_i
\right\|_1
\right]
= \sum_{j=1}^{d} \mathbb{E} [||\text{clip}_2(N_i, Q)_j - N_{i,j} + Z_j||]
= \sum_{j=1}^{d} \mathbb{E} [||\sum_{i} \text{clip}_2(N_i, Q)_j + Z_j - \sum_{i} N_{i,j}|| | Z_j < 0] \cdot \mathbb{P}(Z_j < 0)
+ \sum_{j=1}^{d} \mathbb{E} [||\sum_{i} \text{clip}_2(N_i, Q)_j + Z_j - \sum_{i} N_{i,j}|| | Z_j \geq 0] \cdot \mathbb{P}(Z_j \geq 0)
\geq \sum_{j=1}^{d} \mathbb{E} [||\sum_{i} \text{clip}_2(N_i, Q)_j + Z_j - \sum_{i} N_{i,j}|| | Z_j < 0] \cdot \mathbb{P}(Z_j < 0)
= \frac{1}{2} \sum_{j=1}^{d} \mathbb{E} [||\sum_{i} \text{clip}_2(N_i, Q)_j + Z_j - \sum_{i} N_{i,j}|| | Z_j < 0]
= \frac{1}{2} \left( Q \cdot M + \sum_{i: ||N_i||_2 > Q} \left(1 - \frac{Q}{||N_i||_2}\right) ||N_i||_1 \right)
$$

14
\[ \geq \frac{1}{2} \left( C^* \cdot M + \sum_{i : \|N_i\|_2 > C^*} \left( 1 - \frac{C^*}{\|N_i\|_2} \right) \|N_i\|_1 \right) = \frac{1}{2} G(C^*). \]

This shows that \( C^* \) yields a 2-approximation.

\[\]

**A.2 Proof of Corollary 4.4**

To estimate \( C^* \) privately, one can use the output perturbation algorithm. For ease of analysis we consider the regularized problem. More precisely, let

\[ f_i(C) = \max \{1 - C/\|N_i\|_2, 0\}\|N_i\|_1. \]

Note that \( f_i \) is \( L \)-Lipschitz where \( L = \sqrt{d} \). The goal is to minimize the following function

\[ F_1(C) = \frac{1}{n} \sum_{i=1}^{n} f_i(C) + \frac{CM}{n} + \frac{\lambda}{2} C^2. \]  

(6)

Let \( C^*_1 = \arg \min_{C \geq 0} F_1(C) \). We first compute the sensitivity of \( C^*_1 \) as a function of the dataset.

We first compute the sensitivity of \( C' \). Consider a pair of neighboring datasets \( D \) and \( D' \) which only differ by the \( n \)th user.

**Lemma A.1.** Let \( N'_n \) be a histogram and \( f'_n(C) \) defined similarly as \( f_n(C) \) with \( N_n \) replaced by \( N'_n \). Let \( F_1(C) = \frac{1}{n} \sum_{i=1}^{n} f_i(C) + \frac{CM}{n} + \frac{\lambda}{2} C^2 \) and \( F_2(C) = \frac{1}{n} \sum_{i=1}^{n-1} f_i(C) + \frac{1}{n} f'_n(C) + \frac{CM}{n} + \frac{\lambda}{2} C^2 \). Let \( C^*_1 = \arg \min_{C \geq 0} F_2(C) \) and \( C^*_2 = \arg \min_{C \geq 0} F_2(C) \). Then,

\[ |C^*_1 - C^*_2| \leq \Delta \coloneqq \frac{4 \sqrt{s}}{\lambda n}, \]

**Proof.** Observe that \( f_i(C) \) is \( \sqrt{s} \) Lipschitz. Let \( L = \sqrt{s} \).

\[
\begin{align*}
    n(F_1(C^*_2) - F_1(C^*_1)) & = \sum_{i=1}^{n} f_i(C^*_2) - \sum_{i=1}^{n} f_i(C^*_1) + M(C^*_2 - C^*_1) + \frac{n \lambda}{2} ((C^*_2)^2 - (C^*_1)^2) \\
    & = \sum_{i=1}^{n-1} f_i(C^*_2) - \sum_{i=1}^{n-1} f_i(C^*_1) + M(C^*_2 - C^*_1) + \frac{n \lambda}{2} ((C^*_2)^2 - (C^*_1)^2) + f_n(C^*_2) - f_n(C^*_1) \\
    & = n(F_2(C^*_2) - F_2(C^*_1)) + f_n(C^*_2) - f_n(C^*_1) - (f'_n(C^*_2) - f'_n(C^*_1)) \\
    & \leq |f_n(C^*_2) - f_n(C^*_1)| + |f'_n(C^*_2) - f'_n(C^*_1)| \\
    & \leq 2L|C^*_2 - C^*_1| 
\end{align*}
\]

Since \( F_1 \) is \( \lambda \)-strongly convex, we have

\[ F_1(C^*_2) - F_1(C^*_1) \geq \frac{\lambda}{2} |C^*_2 - C^*_1|^2 \]

Combining the two parts,

\[ |C^*_2 - C^*_1| \leq \frac{4L}{\lambda n}. \]

Now we can characterize performance of the combined algorithm which uses the output of Algorithm 2 \( \hat{C} \), as the clipping threshold in Algorithm 1.
Lemma A.2. Let $C_m$ be an upper bound on $C^*$. Then
\[
\mathbb{E}[\mathcal{L}_G(\hat{C}, D)] - 2 \inf_{C > 0} \mathcal{L}_G(C, D) \leq \frac{n\lambda C_m^2}{2} + \frac{4s}{\lambda^{\varepsilon'}}.
\]

Proof. Recall the definition of $C^*_1$ from Lemma A.1. First we write the expression,
\[
\mathbb{E}[\mathcal{L}_G(\hat{C}, D)] - 2 \inf_{C > 0} \mathcal{L}_G(C, D) \leq \mathbb{E}[\mathcal{M}(\hat{C})] - 2 \mathbb{E}[\mathcal{M}(C^*_1)] + \sum_{i=1}^{n} f_i(C^*_1) - \sum_{i=1}^{n} f_i(C^*_1) + \sum_{i=1}^{n} f_i(C^*_1) - C^* M
\]
The first inequality comes from the proof of Lemma 4.2. We bound the terms separately,
\[
\mathbb{E}\left[\sum_{i=1}^{n} f_i(\hat{C}) - \sum_{i=1}^{n} f_i(C^*_1)\right] \leq n\sqrt{s}\mathbb{E}[|\hat{C} - C^*_1|] \leq n\sqrt{s} \frac{\Delta}{\varepsilon'} = \frac{4\sqrt{s} \sqrt{s}}{\lambda^{\varepsilon'}}.
\]
The remaining terms are bounded using the following fact
\[
\sum_{i=1}^{n} f_i(C^*_1) + C^*_1 M + \frac{\lambda}{2} (C^*_1)^2 \leq \sum_{i=1}^{n} f_i(C^*_1) + C^* M + \frac{\lambda C^*_2}{2}.
\]
Hence,
\[
\sum_{i=1}^{n} f_i(C^*_1) + C^*_1 M - \sum_{i=1}^{n} f_i(C^*_1) - C^* M \leq \frac{n\lambda C^*_2}{2} \leq \frac{n\lambda C_m^2}{2}.
\]
Combining equation 7 and 8 yields the desired result.

The proof of differential privacy follows from Lemma A.1 and the definition of Laplace mechanism. Setting $\lambda = \frac{2\sqrt{2\pi}}{C_m \sqrt{n\varepsilon}}$ in Lemma A.2 yields the error.

A.3 Proof of Theorem 5.1

Proof. Let $Y = \frac{1}{n} Y_i$. Then, when $N_i \sim \text{Poi}(\lambda_i)$,
\[
\mathbb{E}[Y] = \frac{1}{n} \sum_{i=1}^{n} h(\lambda_i)
\]
\[
= \frac{1}{n} \sum_{i=1}^{n} \left( \sum_{j=0}^{C-1} j \cdot \Pr[N_i = j] + \sum_{j=0}^{\infty} C \cdot \Pr[N_i = j] \right)
\]
\[
= \frac{1}{n} \sum_{i=1}^{n} \left( C - \sum_{j=0}^{C-1} (C - j) \Pr[N_i = j] \right)
\]
\[ \lambda = \frac{\mathrm{Lap}(\lambda, \lambda - 1)}{C/n} \]

where \( \lambda = h^{-1}(Y + Z) \). We first bound the error for estimating \( h(\lambda) \),

\[
\mathbb{E} \left[ (h(\lambda) - h(\hat{\lambda}))^2 \right] = \mathbb{E} \left[ (h(\lambda) - \mathbb{E}[Y] + \mathbb{E}[Y] - h(\hat{\lambda}))^2 \right] \\
= \mathbb{E}[h(\lambda) - \mathbb{E}[Y]^2] + (\mathbb{E}[Y] - h(\lambda))^2 \\
= \mathbb{E}[Z^2] + \mathbb{E}[(Y - \mathbb{E}Y)^2] + (\mathbb{E}[Y] - h(\lambda))^2 \\
\leq \frac{C^2}{n^2 \varepsilon^2} + \mathbb{E}[(Y - \mathbb{E}Y)^2] + (\mathbb{E}[Y] - h(\lambda))^2. \tag{9}
\]

We first bound \( \mathbb{E}[(Y - \mathbb{E}[Y])^2] \).

\[
\mathbb{E}[(Y - \mathbb{E}[Y])^2] = \frac{1}{n^2} \mathbb{E} \left[ \left( \sum_{i=1}^{n} (Y_i - \mathbb{E}[Y_i]) \right)^2 \right] \\
= \frac{1}{n^2} \sum_{i=1}^{n} \mathbb{V}[Y_i] \\
\leq \frac{1}{n^2} \sum_{i=1}^{n} \mathbb{V}[N_i] = \frac{1}{n^2} \sum_{i=1}^{n} \lambda_i. \tag{10}
\]

The final inequality is due to \( \mathbb{V}[Y_i] \leq \mathbb{V}[N_i] \). To see this we use the symmetrization trick. Let \( N'_i \) be an independent copy of \( N_i \) and \( Y'_i = \text{clip}(N'_i, C) \). Then by definition \( |Y_i - Y'_i| \leq |N_i - N'_i| \). Hence,

\[
\mathbb{V}[Y_i] = \mathbb{E}[(Y_i - Y'_i)^2/2] \leq \mathbb{E}[(N_i - N'_i)^2/2] = \mathbb{V}[N_i].
\]

We then bound \( |\mathbb{E}[Y] - h(\hat{\lambda})| \). We compute Taylor expansion of \( \mathbb{E}[Y] \) at \( \hat{\lambda} \) with the Lagrangian remainder,

\[
\mathbb{E}[Y] = \frac{1}{n} \sum_{i=1}^{n} \left( h(\lambda) + h'(\lambda)(\lambda_i - \lambda) + \frac{h''(\xi_i)}{2}(\lambda - \lambda_i)^2 \right) \\
= h(\lambda) + \frac{1}{n} \sum_{i=1}^{n} \frac{h''(\xi_i)}{2}(\lambda - \lambda_i)^2,
\]

where \( \xi_i \in (\min\{\lambda, \lambda_i\}, \max\{\bar{\lambda}, \lambda_i\}) \). We move on to compute \( h''(\lambda) \),

\[
h''(\lambda) = e^{-\lambda} \left( - \sum_{j=0}^{C-1} \frac{C - j}{j!} \lambda^j + 2 \sum_{j=0}^{C-2} \frac{C - (j + 1)}{j!} \lambda^j - \sum_{j=0}^{C-3} \frac{C - (j + 2)}{j!} \lambda^j \right) \\
= -e^{-\lambda} \frac{\lambda^{C-1}}{(C - 1)!}.
\]

We can verify that for \( C \geq 1 \), \( |h''(\lambda)| \) increases when \( \lambda \leq C - 1 \) and decreases when \( \lambda \geq C - 1 \). Therefore by Sterling’s approximation,

\[
|h''(\xi_i)| \leq \frac{(C - 1)^{C-1}}{e^{C-1}(C - 1)!} \leq \frac{1}{\sqrt{2\pi}(C - 1)}.
\]

Thus,

\[
|\mathbb{E}[Y] - h(\bar{\lambda})| \leq \frac{1}{2\sqrt{2\pi}(C - 1)} \cdot \frac{1}{n} \sum_{i=1}^{n} (\lambda_i - \bar{\lambda})^2.
\]
Combining all the parts we have
\[
\mathbb{E}[|h(\hat{\lambda}) - h(\bar{\lambda})|] \leq \frac{C}{n\varepsilon} + \frac{1}{n} \sum_{i=1}^{n} \lambda_i + \frac{1}{2\sqrt{2\pi(C-1)}} \cdot \frac{1}{n} \sum_{i=1}^{n} (\lambda_i - \bar{\lambda})^2.
\]

To bound the estimation error \(\mathbb{E}[|\hat{\lambda} - \bar{\lambda}|]\), we first compute \(h'(\lambda)\),
\[
h'(\lambda) = -e^{-\lambda} \left( \sum_{j=0}^{C-2} \frac{C-(j+1)}{j!} \lambda^j - \sum_{j=0}^{C-1} \frac{C-j}{j!} \lambda^j \right) = e^{-\lambda} \sum_{j=0}^{C-1} \frac{\lambda^j}{j!}.
\]

Let \(X \sim \text{Poi}(1)\). We note that for \(\lambda \in [0, 1]\),
\[
|h'(\lambda)| = |h'(1)| = 1 - \Pr[X \geq C] = \frac{1}{\gamma_C} \geq 1 - e^{-\frac{(C-1)^2}{2}}.
\]

Hence, we can proceed to bound the error for estimating \(\bar{\lambda}\),
\[
\mathbb{E}[(\hat{\lambda} - \bar{\lambda})^2] \leq \sup_{\lambda \in [0,1]} \frac{1}{|h'(\lambda)|^2} \mathbb{E}[(h(\hat{\lambda}) - h(\bar{\lambda}))^2]
\]
\[
\leq \frac{C^2}{n^2\varepsilon^2} + \frac{1}{n^2} \sum_{i=1}^{n} \lambda_i + \frac{1}{8\pi(C-1)} \left( \frac{1}{n} \sum_{i=1}^{n} (\lambda_i - \bar{\lambda})^2 \right)^2.
\]

If we assume that \(\lambda_i \leq 1\) for all \(i\) we can bound \(|h''(\xi_i)|\) as
\[
|h''(\xi_i)| \leq \max_{\lambda \in [0,1]} |h''(\lambda)| \leq \frac{1}{(C-1)!}.
\]

Hence the last term can be bounded as
\[
|\mathbb{E}[Y] - h(\lambda)| \leq \frac{1}{2(C-1)!} \cdot \frac{1}{n} \sum_{i=1}^{n} (\lambda_i - \lambda)^2.
\]

\[A.4\text{ Improved bound of Theorem 5.1}\]

We can also obtain a more refined analysis with a better dependence on \(C\). Suppose that \(C \geq 3\). Let \(\eta = 1/2\).

We bound \(|\mathbb{E}[Y] - h(\bar{\lambda})|\) by analyzing \(\lambda_i\) close to \(C-1\) and far from \(C-1\) separately.
\[
|\mathbb{E}[Y] - h(\bar{\lambda})| = \frac{1}{n} \sum_{i=1}^{n} \frac{|h''(\xi_i)|}{2} (\bar{\lambda} - \lambda_i)^2
\]
\[
= \frac{1}{n} \sum_{\xi_i: \frac{\lambda_i}{\bar{\lambda} - \lambda_i} \in (\eta,1/\eta)} \frac{|h''(\xi_i)|}{2} (\bar{\lambda} - \lambda_i)^2 + \frac{1}{n} \sum_{\xi_i: \frac{\lambda_i}{\bar{\lambda} - \lambda_i} \notin (\eta,1/\eta)} \frac{|h''(\xi_i)|}{2} (\bar{\lambda} - \lambda_i)^2.
\]
If $\lambda_i \leq \eta(C - 1)$ and $C \geq 3$, then we also have $\xi_i \leq \eta(C - 1)$. In this case,

$$|h''(\xi_i)| \leq |h''(\eta(C - 1))| = \frac{(\eta(C - 1))^{C - 1}}{e^{\eta(C - 1)}(C - 1)!} \leq \frac{1}{\sqrt{2\pi(C - 1)}} \left( \frac{83}{e \eta(C - 1)} \right)^{C - 1} \leq \frac{1}{\sqrt{2\pi(C - 1)}} \cdot 0.83^{C - 1}. $$

The last inequality follows by $ex/e^x$ is increasing for $x \in [0, 1]$. Using a similar argument, we can verify that the above inequality also holds when $\lambda_i \geq (C - 1)/\eta$. Therefore,

$$|\mathbb{E}[Y] - h(\bar{\lambda})| \leq \frac{1}{2\sqrt{2\pi(C - 1)}} \cdot \frac{1}{n} \sum_{i: \lambda_i \notin \{1/2, 2\}} 0.83^{C - 1}(\lambda_i - \bar{\lambda})^2 + \sum_{i: \lambda_i \in \{1/2, 2\}} (\lambda_i - \bar{\lambda})^2. $$

For convenience let $\Sigma = \frac{1}{n} \sum_{i=1}^{n} (\lambda_i - \bar{\lambda})^2$. We can further bound the bias by

$$|\mathbb{E}[Y] - h(\bar{\lambda})| \leq 0.83^{C - 1} \Sigma + \frac{\sqrt{2}(C - 1)^{3/2}}{\sqrt{\pi}} \frac{1}{n} \sum_{i=1}^{n} 1_{\lambda_i > (C - 1)/2}. \tag{13}$$

We can see that the bias term depends on $C$, $\Sigma$, and $\sum_{i=1}^{n} 1_{\lambda_i > 2(C - 1)}$ (which depends on the distribution of $\{\lambda_i\}_{i=1}^{n}$). To ensure an $O(1/n)$ rate, the two terms should be at most $O(1/\sqrt{n})$.

$$0.83^{C - 1} \Sigma \leq \frac{1}{\sqrt{n}} \frac{4(C - 1)^{3/2}}{2\sqrt{2\pi}} \frac{1}{n} \sum_{i=1}^{n} 1_{\lambda_i > (C - 1)/2} \leq \frac{1}{\sqrt{n}}. $$

Note that in the worst case

$$\frac{1}{n} \sum_{i=1}^{n} 1_{\lambda_i > (C - 1)/2} \leq O \left( \frac{\Sigma}{(C - 1)^2} \right),$$

which recovers (1). The bound could have a better dependence on $C$ if $\lambda_i$s are more concentrated. For example, if $\lambda_i \leq 1$, then the above quantity is 0 as long as $C > 3$, and we can choose $C = 3 + O(\log(1 + n\Sigma))$ to achieve $O(1/n)$ mean squared error.

More generally, if $\lambda_i$s are from a distribution with exponential tail, i.e. $\Pr[\lambda_i \geq x] = O(\exp(-\Omega(x)))$, then

$$\frac{1}{n} \sum_{i=1}^{n} 1_{\lambda_i > 2(C - 1)} \approx \Pr[\lambda_i \geq (C - 1)/2] = O(\exp(-\Omega(C - 1))).$$

Choosing $C = 3 + O(\log n)$ gives $O(1/n)$ mean squared error.

### A.5 Proof of Theorem 5.2

Before we proceed to the proof, we first characterize the error of Algorithm 1 with Laplace noise.

**Lemma A.3.** Let $\hat{\lambda}_L = \hat{N}_L/n$ where $\hat{N}_L$ is the output of Algorithm 1 with Laplace noise. Then

$$\mathbb{E} \left[ (\bar{\lambda} - \hat{\lambda}_L)^2 \right] = \left( \bar{\lambda} - \frac{1}{n} \sum_{i=1}^{n} h(\lambda_i) \right)^2 + \frac{1}{n^2} \sum_{i=1}^{n} \text{Var}(Y_i) + \frac{C^2}{n^2 \varepsilon^2}. $$

**Proof.**

$$\mathbb{E} \left[ (\bar{\lambda} - \hat{\lambda}_L)^2 \right] = \mathbb{E} \left[ (\bar{\lambda} - \mathbb{E}[\hat{\lambda}_L] + \mathbb{E}[\hat{\lambda}_L] - \hat{\lambda}_L)^2 \right] $$

$$= \mathbb{E} \left[ (\bar{\lambda} - \mathbb{E}[\hat{\lambda}_L])^2 \right] + \mathbb{E} \left[ (\mathbb{E}[\hat{\lambda}_L] - \hat{\lambda}_L)^2 \right] $$

19
\[
(\bar{\lambda} - \frac{1}{n} \sum_{i=1}^{n} h(\lambda_i))^2 + \frac{1}{n^2} \sum_{i=1}^{n} \text{Var}(Y_i) + \frac{C^2}{n^2 \varepsilon^2}.
\]

Now we have all the ingredients to complete Theorem 5.2

Proof. Combining \((9), (10), \) and \((11)\) we have

\[
E[(\bar{\lambda} - \hat{\lambda})^2] \leq \gamma^2 C \left( \frac{C^2}{n^2 \varepsilon^2} + \frac{1}{n^2} \sum_{i=1}^{n} \text{Var}[Y_i] + \left( h(\bar{\lambda}) - \frac{1}{n} \sum_{i=1}^{n} h(\lambda_i) \right)^2 \right).
\]

Let \(\bar{h} = -\frac{1}{n} \sum_{i=1}^{n} h(\lambda_i).\) Therefore,

\[
E[(\bar{\lambda} - \hat{\lambda})^2] - E[(\bar{\lambda} - \lambda)^2] = -\left( \gamma^2 - 1 \right) \left( \frac{C^2}{n^2 \varepsilon^2} + \frac{1}{n^2} \sum_{i=1}^{n} \text{Var}[Y_i] \right) + (\bar{\lambda} - \bar{h})^2 - \gamma^2 (h(\bar{\lambda}) - \bar{h})^2
\]

We bound the terms separately. Note that \(\text{Var}[Y_i] \leq \text{Var}[N_i] = \lambda_i.\) Hence,

\[
(\gamma^2 - 1) \left( \frac{C^2}{n^2 \varepsilon^2} + \frac{1}{n^2} \sum_{i=1}^{n} \text{Var}[Y_i] \right) \leq (\gamma^2 - 1) \left( \frac{C^2}{n^2 \varepsilon^2} + \frac{1}{n} \right).
\]

Recall the assumption that \(\bar{h} \geq h_{\min} := h(\bar{\lambda}) - \frac{\bar{\lambda} - h(\bar{\lambda})}{\gamma C - 1},\) and write \(\bar{h} = h_{\min} + \alpha \frac{\bar{\lambda} - h(\bar{\lambda})}{\gamma C - 1}.\) The remaining part simplifies to

\[
(\bar{\lambda} - \bar{h})^2 - \gamma^2 (h(\bar{\lambda}) - \bar{h})^2 = \frac{\gamma C - \alpha}{\gamma C - 1} \left( \frac{\gamma C - \alpha}{\gamma C - 1} - \alpha \right) (\bar{\lambda} - h(\bar{\lambda}))^2
\]

Combining the two parts completes the proof.

\[\square\]

### B Additional experiments

#### B.1 Algorithm \[\text{Log}\] with fixed \(C_m = 150\)

Fixing \(C_m\) performs better than \(C_m = DP-M\text{-quantile},\) because in the latter case, we need to split half of the privacy budget to estimate \(DP-M\text{-quantile}.\) We note that the output perturbation method has large standard deviation for the SNAP dataset and investing the reason behind it is an interesting future direction.
Figure 4: Histogram estimation. Left: Sentiment140 dataset. Right: SNAP dataset