Tight Differential Privacy for Discrete-Valued Mechanisms and for the Subsampled Gaussian Mechanism Using FFT

Koskela, Antti

Microtome Publishing
2021

Koskela , A , Jalko , J , Prediger , L & Honkela , A 2021 , Tight Differential Privacy for Discrete-Valued Mechanisms and for the Subsampled Gaussian Mechanism Using FFT . in A Banerjee & K Fukumizu (eds) , 24th International Conference on Artificial Intelligence and Statistics (AISTATS) . Proceedings of Machine Learning Research , vol. 130 , Microtome Publishing , 24th International Conference on Artificial Intelligence and Statistics (AISTATS) , 13/04/2021 .

http://hdl.handle.net/10138/336153

unspecified
publishedVersion

Tight Differential Privacy for Discrete-Valued Mechanisms and for the Subsampled Gaussian Mechanism Using FFT

Antti Koskela
University of Helsinki

Joonas Jälkö
Aalto University

Lukas Prediger
Aalto University

Antti Honkela
University of Helsinki

Abstract

We propose a numerical accountant for evaluating the tight (ε, δ)-privacy loss for algorithms with discrete one dimensional output. The method is based on the privacy loss distribution formalism and it uses the recently introduced fast Fourier transform based accounting technique. We carry out an error analysis of the method in terms of moment bounds of the privacy loss distribution which leads to rigorous lower and upper bounds for the true (ε, δ)-values. As an application, we present a novel approach to accurate privacy accounting of the subsampled Gaussian mechanism. This completes the previously proposed analysis by giving strict lower and upper bounds for the privacy parameters. We demonstrate the performance of the accountant on the binomial mechanism and show that our approach allows decreasing noise variance up to 75 percent at equal privacy compared to existing bounds in the literature. We also illustrate how to compute tight bounds for the exponential mechanism applied to counting queries.

1 Introduction

Differential privacy (DP) (Dwork et al., 2006) has been established as the standard approach for privacy-preserving machine learning. As DP algorithms have grown increasingly complex, accurately bounding the compound privacy loss has become more challenging as well. The moments accountant (Abadi et al., 2016) represented a major breakthrough in the accuracy of bounding the privacy loss in compositions of subsampled Gaussian mechanisms that are commonly used in DP stochastic gradient descent (DP-SGD). This has further been refined through the general development of Rényi differential privacy (RDP) (Mironov, 2017) as well as tighter RDP bounds for subsampled mechanisms (Balle et al., 2018; Wang et al., 2019; Zhu and Wang, 2019; Mironov et al., 2019). RDP enables tight analysis for compositions of Gaussian mechanisms, but this may be difficult for other mechanisms. Moreover, conversion of RDP guarantees back to more commonly used (ε, δ)-guarantees is lossy.

In this work, we focus on an alternative approach based on the privacy loss distribution (PLD) formalism introduced by Sommer et al. (2019). This work directly extends the recent Fourier accountant by Koskela et al. (2020) to discrete mechanisms. We provide a rigorous error analysis which leads to strict (ε, δ)-bounds. This analysis is further used to obtain strict bounds for the subsampled Gaussian mechanism.

The need to consider discrete mechanisms for rigorous DP on finite-precision computers was first pointed out by Mironov (2012). Agarwal et al. (2018) implement a communication efficient binomial mechanism cpSGD for neural network training which however cannot handle compositions. Agarwal et al. (2018) and Kairouz et al. (2019) note the need for a privacy accountant for the binomial mechanism as an important open problem, which we solve in this paper for the case where gradients are replaced with a sign approximation.

The outline of the paper is as follows. In Sections 2 and 3 we give the basic definitions and describe the PLD formalism used for our accountant. In Section 4 we describe the algorithm based on the fast Fourier transform (FFT) and in Section 5 we provide an error analysis. Section 6 concludes with experiments illustrating the efficiency and accuracy of the method.

Implementation of the methods is available in Github[1].

Our Contribution. We extend the work by Koskela et al. (2020) which considered an FFT based method

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

[1] https://github.com/DPBayes/PLD-Accountant/

for approximating the tight (ε, δ)-DP guarantees of the subsampled Gaussian mechanism, however without strict lower and upper bounds. The main contributions of this work are:

- A framework for computing tight (ε, δ)-DP guarantees of discrete-valued mechanisms.
- An error analysis of the proposed method using moment bounds of the mechanism at hand, which leads to strict lower and (ε, δ)-upper bounds.
- Accurate lower and upper bounds for (ε, δ)-DP of the subsampled Gaussian mechanism.

2 Differential Privacy

We first recall some basic definitions of DP (Dwork et al., 2006). We use the following notation. An input data set containing N data points is denoted as $X = (x_1, \dots, x_N) \in \mathcal{X}^N$, where $x_i \in \mathcal{X}$, $1 \leq i \leq N$.

Definition 1. *We say two data sets X and Y are neighbours in remove/add relation if we get one by removing/adding an element from/to the other and denote this with $\sim_R$. We say X and Y are neighbours in substitute relation if we get one by substituting one element in the other. We denote this with $\sim_S$.*

Definition 2. *Let $\varepsilon > 0$ and $\delta \in [0,1]$. Let $\sim$ define a neighbouring relation. Mechanism $\mathcal{M} : \mathcal{X}^N \to \mathcal{R}$ is $(\varepsilon, \delta, \sim)$-DP if for every $X \sim Y$ and every measurable set $E \subset \mathcal{R}$ we have that*

$$\Pr(\mathcal{M}(X) \in E) \leq \mathrm{e}^{\varepsilon} \Pr(\mathcal{M}(Y) \in E) + \delta.$$

When the relation is clear from context or irrelevant, we will abbreviate it as (ε, δ)-DP. We call $\mathcal{M}$ tightly $(\varepsilon, \delta, \sim)$-DP, if there does not exist $\delta' < \delta$ such that $\mathcal{M}$ is $(\varepsilon, \delta', \sim)$-DP.

3 Privacy Loss Distribution

We first introduce the basic tool for obtaining tight privacy bounds: the privacy loss distribution (PLD). The results in Subsection 3.1 are reformulations of the results given by Meiser and Mohammadi (2018) and Sommer et al. (2019). Proofs of the results of this section are given in the supplementary material.

3.1 Privacy Loss Distribution

We consider discrete-valued one-dimensional mechanisms $\mathcal{M}$ which can be seen as mappings from $\mathcal{X}^N$ to the set of discrete-valued random variables. The *generalised probability density functions* of $\mathcal{M}(X)$ and $\mathcal{M}(Y)$, denoted $f_X(t)$ and $f_Y(t)$, respectively, are given by

$$\begin{aligned} f_X(t) &= \sum\nolimits_i a_{X,i} \cdot \delta_{t_{X,i}}(t), \\ f_Y(t) &= \sum\nolimits_i a_{Y,i} \cdot \delta_{t_{Y,i}}(t), \end{aligned} \tag{1}$$

where $\delta_t(\cdot)$, $t \in \mathbb{R}$, denotes the Dirac delta function centered at t, and $t_{X,i}, t_{Y,i} \in \mathbb{R}$ and $a_{X,i}, a_{Y,i} \geq 0$. Equivalently, (1) means that for all i,

$$\begin{aligned} \mathbb{P}(\mathcal{M}(X) = t_{X,i}) &= a_{X,i}, \\ \mathbb{P}(\mathcal{M}(Y) = t_{Y,i}) &= a_{Y,i}. \end{aligned}$$

Thus, we have that

$$\begin{aligned} \int_{-\infty}^{\infty} f_X(t) \, \mathrm{d}t &= \sum\nolimits_i a_{X,i} = 1, \\ \int_{-\infty}^{\infty} f_Y(t) \, \mathrm{d}t &= \sum\nolimits_i a_{Y,i} = 1. \end{aligned}$$

If g is a function such that $g(X)$ determines a random variable, then

$$\begin{aligned} \mathbb{E}_{s \sim X}[g(s)] &= \int_{-\infty}^{\infty} g(t) f_X(t) \, \mathrm{d}t \\ &= \sum\nolimits_i a_{X,i} \cdot g(t_{X,i}). \end{aligned} \tag{2}$$

More generally, we define integrals over generalised probability density functions as in (2). We prefer using the integral notation as it simplifies the analysis.

We define the privacy loss distribution as follows.

Definition 3. *Let $\mathcal{M} : \mathcal{X}^N \to \mathcal{R}$, $\mathcal{R} \subset \mathbb{R}$, be a discrete-valued randomised mechanism and let $f_X(t)$ and $f_Y(t)$ be probability density functions of the form (1). We define the privacy loss distribution $\omega_{X/Y}$ as*

$$\omega_{X/Y}(s) = \sum\nolimits_{t_{X,i} = t_{Y,j}} a_{X,i} \cdot \delta_{s_i}(s), \tag{3}$$

where $s_i = \log\left(\frac{a_{X,i}}{a_{Y,j}}\right)$.

Notice that this definition differs slightly from the one given by Sommer et al. (2019, Def. 4.2): we do not include the symbol ∞ in ω. Thus, if $f_X(t)$ and $f_Y(t)$ do not have equal supports, we have $\int_{\mathbb{R}} \omega_{X/Y}(s) \, \mathrm{d}s < 1$. This situation is included in our Lemma 4 and Theorem 5, and the analysis of Section 5 also applies then. We remark that Def. 3 is related to the KL divergence, as $\mathrm{KL}(f_X || f_Y) = \mathbb{E}[\omega_{X/Y}] = \int_{-\infty}^{\infty} s \cdot \omega_{X/Y}(s) \, \mathrm{d}s$ in case f_X and f_Y have equal supports.

Evaluating (ε, δ)-bounds using the PLD formalism is essentially based on a result (Supplements) which states that the mechanism $\mathcal{M}$ is tightly (ε, δ)-DP with

$$\begin{aligned} \delta(\varepsilon) = \max_{X \sim Y} \Bigg\{ &\int_{\mathbb{R}} \max\{f_X(t) - \mathrm{e}^{\varepsilon} f_Y(t), 0\} \, \mathrm{d}t, \\ &\int_{\mathbb{R}} \max\{f_Y(t) - \mathrm{e}^{\varepsilon} f_X(t), 0\} \, \mathrm{d}t \Bigg\}. \end{aligned} \tag{4}$$

This relation holds for both continuous and discrete output mechanisms, and a more general version of this result using so called f-divergences is given by Barthe and Olmedo (2013). In case f_X and f_Y are generalised probability density functions of the form (1), i.e.,

$$f_X(t) - \mathrm{e}^{\varepsilon} f_Y(t) = \sum\nolimits_i c_i \cdot \delta_{t_i}(t)$$

for some coefficients $c_i, t_i \in \mathbb{R}$, then in (4) we denote

$$\max\{f_X(t) - \mathrm{e}^{\varepsilon} f_Y(t), 0\} = \sum\nolimits_i \max\{c_i, 0\} \cdot \delta_{t_i}(t).$$

For the discrete-valued mechanisms, the relation (4) was originally given by Sommer et al. (2019, Lemmas 5 and 10). Assuming the PLD distribution is of the form (3), the relation (4) directly gives the following representation for $\delta(\varepsilon)$.

Lemma 4. *$\mathcal{M}$ is tightly (ε, δ)-DP for*

$$\delta(\varepsilon) = \max_{X \sim Y} \{\delta_{X/Y}(\varepsilon), \delta_{Y/X}(\varepsilon)\},$$

where

$$\begin{aligned} &\delta_{X/Y}(\varepsilon) = \delta_{X/Y}(\infty) + \int_{\varepsilon}^{\infty} (1 - \mathrm{e}^{\varepsilon - s})\, \omega_{X/Y}(s)\, \mathrm{d}s, \\ &\delta_{X/Y}(\infty) = \\ &\sum\nolimits_{\{t_i \,:\, \mathbb{P}(\mathcal{M}(X)=t_i)>0,\, \mathbb{P}(\mathcal{M}(Y)=t_i)=0\}} \mathbb{P}(\mathcal{M}(X) = t_i), \end{aligned} \tag{5}$$

and similarly for $\delta_{Y/X}(\varepsilon)$.

We remark that finding the outputs $\mathcal{M}(X)$ and $\mathcal{M}(Y)$ that give the maximum $\delta(\varepsilon)$ is application specific and has to be carried out individually for each case, similarly as, e.g., in the case of RDP (Mironov, 2017).

3.2 Example: The Randomised Response

To illustrate the formalism described above, consider the randomised response mechanism (Warner, 1965) which is described as follows. Suppose F is a function $F : \mathcal{X} \to \{0, 1\}$. Define the randomised mechanism $\mathcal{M}$ for input $X \in \mathcal{X}$ by

$$\mathcal{M}(X) = \begin{cases} F(X), & \text{with probability } p \\ 1 - F(X), & \text{with probability } 1 - p, \end{cases}$$

where $0 < p < 1$. The mechanism is ε-DP for $\varepsilon = \log \frac{p}{1-p}$ (Dwork and Roth, 2014). Let $X \sim Y$ and let $F(X) = 1$ and $F(Y) = 0$. As these are the only possible outputs, X and Y represent the worst case in Lemma 4 and give the tight $\delta(\varepsilon)$. We see that the density functions of $\mathcal{M}(X)$ and $\mathcal{M}(Y)$ are given by

$$\begin{aligned} f_X(t) &= p \cdot \delta_1(t) + (1-p) \cdot \delta_0(t), \\ f_Y(t) &= (1-p) \cdot \delta_1(t) + p \cdot \delta_0(t). \end{aligned}$$

From (3) we see that

$$\begin{aligned} \omega_{X/Y}(s) &= p \cdot \delta_{c_p}(s) + (1-p) \cdot \delta_{-c_p}(s), \\ \omega_{Y/X}(s) &= p \cdot \delta_{-c_p}(s) + (1-p) \cdot \delta_{c_p}(s), \end{aligned}$$

where $c_p = \log \frac{p}{1-p}$. Assume $\frac{1}{2} < p < 1$. Then by Lemma 4 we see that

$$\delta(\varepsilon) = \begin{cases} p\,(1 - \mathrm{e}^{\varepsilon - c_p}), & \text{if } \varepsilon \leq c_p \\ 0, & \text{else.} \end{cases}$$

As $\varepsilon \to^{-} c_p$, we see that $\delta \to 0$ as expected.

3.3 Tight (ε, δ)-Bounds for Compositions

Let X and Y be random variables described by generalised probability density functions f_X and f_Y of the form (1). We define the convolution $f_X * f_Y$ as

$$(f_X * f_Y)(t) = \sum\nolimits_{i,j} a_{X,i}\, a_{Y,j} \cdot \delta_{t_{X,i} + t_{Y,j}}(t).$$

Notice that $f_X * f_Y$ describes the probability density of the random variable $X + Y$. The following theorem shows that the tight (ε, δ)-bounds for compositions of non-adaptive mechanisms are obtained using convolutions of PLDs (see also Sommer et al., 2019, Thm. 1).

Theorem 5. *Consider a k-fold non-adaptive composition of a mechanism $\mathcal{M}$. The composition is tightly (ε, δ)-DP for $\delta(\varepsilon)$ given by*

$$\delta(\varepsilon) = \max\{\delta_{X/Y}(\varepsilon), \delta_{Y/X}(\varepsilon)\},$$

where

$$\begin{aligned} \delta_{X/Y}(\varepsilon) = 1 - \big(1 - \delta_{X/Y}(\infty)\big)^k + \\ \int_{\varepsilon}^{\infty} (1 - \mathrm{e}^{\varepsilon - s}) \left(\omega_{X/Y} *^k \omega_{X/Y}\right)(s)\, \mathrm{d}s, \end{aligned}$$

where $\delta_{X/Y}(\infty)$ is as defined in (5) and $\omega_{X/Y} *^k \omega_{X/Y}$ denotes the k-fold convolution of the density function $\omega_{X/Y}$ (an analogous expression holds for $\delta_{Y/X}(\varepsilon)$).

We remark that our approach also allows computing tight privacy bounds for a composite mechanism $\mathcal{M}_1 \circ \ldots \circ \mathcal{M}_k$, where the PLDs of the mechanisms $\mathcal{M}_i$ vary (see the supplementary material).

3.4 Subsampling Amplification

The subsampling amplification can be analysed similarly as by Koskela et al. (2020) in the case of the Gaussian mechanism. For example, considering the $\sim_R$-neighbouring relation and using the Poisson subsampling with subsampling ratio $0 < q < 1$ leads to considering the pair of density functions

$$q \cdot f_X + (1-q) \cdot f_Y \quad \text{and} \quad f_Y,$$

where the density function f_X corresponds to a subsample including the additional data element. Subsampling without and with replacement using $\sim_S$-neighbouring relation can be analysed with mixture distributions analogously (Koskela et al., 2020).

4 Fourier Accountant for Discrete-Valued Mechanisms

We next describe the numerical method for computing tight DP guarantees for discrete one-dimensional distributions using the PLD formalism. We will apply the fast Fourier transform to numerically evaluate the PLD convolutions of Theorem 5.

4.1 Fast Fourier Transform

Let

$$x = \big[x_0, \ldots, x_{n-1}\big]^{\mathrm{T}}, \; w = \big[w_0, \ldots, w_{n-1}\big]^{\mathrm{T}} \in \mathbb{R}^n.$$

The discrete Fourier transform $\mathcal{F}$ and its inverse $\mathcal{F}^{-1}$ are defined as (Stoer and Bulirsch, 2013)

$$\begin{aligned}(\mathcal{F}x)_k &= \sum\nolimits_{j=0}^{n-1} x_j \mathrm{e}^{-\mathrm{i}\, 2\pi kj/n},\\ (\mathcal{F}^{-1}w)_k &= \frac{1}{n}\sum\nolimits_{j=0}^{n-1} w_j \mathrm{e}^{\mathrm{i}\, 2\pi kj/n},\end{aligned}$$

where $\mathrm{i} = \sqrt{-1}$. Evaluating $\mathcal{F}x$ and $\mathcal{F}^{-1}w$ naively takes $O(n^2)$ operations, however evaluation using the Fast Fourier Transform (FFT) (Cooley and Tukey, 1965) reduces the running time complexity to $O(n \log n)$.

For our purposes FFT will be useful as it enables evaluating the discrete convolutions efficiently. The so-called convolution theorem (Stockham Jr, 1966) states that for periodic discrete convolutions it holds that

$$\sum\nolimits_{i=0}^{n-1} v_i w_{k-i} = \mathcal{F}^{-1}(\mathcal{F}v \odot \mathcal{F}w), \tag{6}$$

where $\odot$ denotes the elementwise product and the summation indices are modulo n. Using (6), repeated convolutions are evaluated efficiently.

4.2 Grid Approximation

In order to harness the FFT, we place the PLD on a grid

$$X_n = \{x_0, \ldots, x_{n-1}\}, \quad n \in \mathbb{Z}^+, \tag{7}$$

where

$$x_i = -L + i\Delta x, \quad \Delta x = 2L/n.$$

Suppose the distribution ω of the PLD is of the form

$$\omega(s) = \sum\nolimits_{i=0}^{n-1} a_i \cdot \delta_{s_i}(s),$$

where $a_i \geq 0$ and $-L \leq s_i \leq L - \Delta x$, $0 \leq i \leq n-1$. We define the grid approximations

$$\begin{aligned}\omega^{\mathrm{L}}(s) &:= \sum\nolimits_{i=0}^{n-1} a_i \cdot \delta_{s_i^{\mathrm{L}}}(s),\\ \omega^{\mathrm{R}}(s) &:= \sum\nolimits_{i=0}^{n-1} a_i \cdot \delta_{s_i^{\mathrm{R}}}(s),\end{aligned} \tag{8}$$

where

$$\begin{aligned}s_i^{\mathrm{L}} &= \max\{x \in X_n \,:\, x \leq s_i\},\\ s_i^{\mathrm{R}} &= \min\{x \in X_n \,:\, x \geq s_i\},\end{aligned}$$

i.e., s_i^L and s_i^R refer to the closest left and right grid approximation points to s_i. We note that as s_i's correspond to the log ratios of probabilities of individual events, often a moderate L is sufficient for the condition $-L \leq s_i \leq L - \Delta x$ to hold for all i. In the Supplements we provide analysis also for the case where this assumption does not hold. From (8) we have:

Lemma 6. *Let $\delta(\varepsilon)$ be given by the integral formula of Lemma 4 and let $\delta^{\mathrm{L}}(\varepsilon)$ and $\delta^{\mathrm{R}}(\varepsilon)$ be determined analogously by ω^{L} and ω^{R}. Then for all $\varepsilon > 0$:*

$$\delta^{\mathrm{L}}(\varepsilon) \leq \delta(\varepsilon) \leq \delta^{\mathrm{R}}(\varepsilon).$$

Lemma 6 directly generalises to convolutions. The following bounds for the moment generating functions will be used in the error analysis.

Lemma 7. *Let ω, ω^{R} and ω^{L} also denote the random variables determined by the density functions defined above, and let $0 < \lambda < (\Delta x)^{-1}$. Then*

$$\mathbb{E}[\mathrm{e}^{\lambda \omega^{\mathrm{L}}}] \leq \mathbb{E}[\mathrm{e}^{\lambda \omega}], \quad \mathbb{E}[\mathrm{e}^{-\lambda \omega^{\mathrm{L}}}] \leq \tfrac{1}{1-\lambda \Delta x} \mathbb{E}[\mathrm{e}^{-\lambda \omega}]$$

and

$$\mathbb{E}[\mathrm{e}^{-\lambda \omega^{\mathrm{R}}}] \leq \mathbb{E}[\mathrm{e}^{-\lambda \omega}], \quad \mathbb{E}[\mathrm{e}^{\lambda \omega^{\mathrm{R}}}] \leq \tfrac{1}{1-\lambda \Delta x} \mathbb{E}[\mathrm{e}^{\lambda \omega}].$$

4.3 Truncation of Convolutions and Periodisation

The FFT assumes that inputs are periodic over a finite range. We describe truncation of convolutions and periodisation of distribution functions to meet this assumption. Suppose ω is defined such that

$$\omega(s) = \sum\nolimits_i a_i \cdot \delta_{s_i}(s), \tag{9}$$

where $a_i \geq 0$ and $s_i = i\Delta x$. The convolutions can then be written as

$$\begin{aligned}(\omega * \omega)(s) &= \sum\nolimits_{i,j} a_i a_j \cdot \delta_{s_i + s_j}(s)\\ &= \sum\nolimits_i \Big(\sum\nolimits_j a_j a_{i-j}\Big) \cdot \delta_{s_i}(s).\end{aligned}$$

Let $L > 0$. We truncate these convolutions to the interval $[-L, L]$ such that

$$\begin{aligned}(\omega * \omega)(s) &\approx \sum\nolimits_i \Big(\sum\nolimits_{-L \leq s_j < L} a_j a_{i-j}\Big) \cdot \delta_{s_i}(s)\\ &=: (\omega \circledast \omega)(s).\end{aligned}$$

We define $\widetilde{\omega}$ to be a $2L$-periodic extension of ω, i.e., $\widetilde{\omega}$ is of the form

$$\widetilde{\omega}(s) = \sum\nolimits_{m\in\mathbb{Z}} \sum\nolimits_i a_i \cdot \delta_{s_i+m\cdot 2L}(s).$$

We further approximate

$$(\omega \circledast \omega) \approx (\widetilde{\omega} \circledast \widetilde{\omega}).$$

In case the distribution ω is defined on an equidistant grid, FFT can be used to evaluate $\widetilde{\omega} \circledast \widetilde{\omega}$ as follows:

Lemma 8. *Let ω be of the form (9), such that n is even, $L > 0$, $\Delta x = 2L/n$ and $s_i = -L + i\Delta x$, $0 \leq i \leq n-1$. Define*

$$\boldsymbol{a} = \begin{bmatrix} a_0 & \dots & a_{n-1} \end{bmatrix}^{\mathrm{T}} \quad \text{and} \quad D = \begin{bmatrix} 0 & I_{n/2} \\ I_{n/2} & 0 \end{bmatrix} \in \mathbb{R}^{n\times n}.$$

Then,

$$(\widetilde{\omega} \circledast^k \widetilde{\omega})(s) = \sum\nolimits_{i=0}^{n-1} b_i^k \cdot \delta_{s_i}(s),$$

where

$$b_i^k = \left[D\, \mathcal{F}^{-1}\big(\mathcal{F}(D\boldsymbol{a})^{\odot k}\big)\right]_i,$$

and $^{\odot k}$ denotes the elementwise power of vectors.

4.4 Approximation of the $\delta(\varepsilon)$-Integral

Finally, using the truncated and periodised convolutions we approximate the integral formula in Lemma 4 for the tight δ-value as

$$\begin{aligned} &\int_\varepsilon^\infty (1-\mathrm{e}^{\varepsilon - s})(\omega *^k \omega)(s)\, \mathrm{d}s \\ \approx &\int_\varepsilon^L (1-\mathrm{e}^{\varepsilon - s})(\widetilde{\omega} \circledast^k \widetilde{\omega})(s)\, \mathrm{d}s \\ = &\sum\nolimits_{\ell=\ell_\varepsilon}^{n-1} \big(1 - \mathrm{e}^{\varepsilon-(-L+\ell\Delta x)}\big)\, b_\ell^k, \end{aligned} \tag{10}$$

where $\ell_\varepsilon = \min\{\ell \in \mathbb{Z} \,:\, -L+\ell\Delta x > \varepsilon\}$ and the vector $b^k \in \mathbb{R}^n$ is given by Lemma 8. We describe the method in the pseudocode of Algorithm 1. In the following section we give an error bound for the approximation with respect to the parameter L.

***Remark* 9.** *To evaluate ε as a function of δ, Newton's method can be used (Koskela et al., 2020). Suppose ω is continuous and $\delta(\varepsilon)$ given by the integral (10). Then, $\delta'(\varepsilon) = -\int_\varepsilon^\infty \mathrm{e}^{\varepsilon-s}(\omega *^k \omega)(s)\, \mathrm{d}s$ and Newton's method applied to the function $\delta(\varepsilon) - \bar{\delta}$ gives the iteration*

$$\varepsilon_{\ell+1} = \varepsilon_\ell - \frac{\delta(\varepsilon_\ell) - \bar{\delta}}{\delta'(\varepsilon_\ell)}. \tag{11}$$

Similarly to (10) this naturally translates to the case of discrete distributions. We use as a stopping criterion $\big|\delta(\varepsilon_\ell) - \bar{\delta}\big| \leq \tau$ for some prescribed tolerance parameter τ and an initial value $\varepsilon_0 = 0$. In experiments, for an equal stopping criterion τ, the iteration (11) gave more than twice as fast convergence as the binary search algorithm.

Algorithm 1 Fourier Accountant Algorithm for Discrete-Valued Mechanisms

```
Input: distribution $\omega$ of the form (9), such that $n$ is even and $s_i = -L + i\Delta x$, $0 \leq i \leq n-1$, $\Delta x = 2L/n$, number of compositions $k$.
Set
    $\boldsymbol{a} = \begin{bmatrix} a_0 & \dots & a_{n-1} \end{bmatrix}^{\mathrm{T}}, \quad D = \begin{bmatrix} 0 & I_{n/2} \\ I_{n/2} & 0 \end{bmatrix}.$
Evaluate the convolutions using Lemma 8 and FFT:
    $\boldsymbol{b}^k = \left[ D\, \mathcal{F}^{-1}\big(\mathcal{F}(D\boldsymbol{a})^{\odot k}\big)\right],$
Determine the starting point of the integral interval:
    $\ell_\varepsilon = \min\{\ell \in \mathbb{N} \,:\, -L + \ell\Delta x > \varepsilon\},$
Approximate $\delta(\varepsilon)$ using Lemma 4:
    $\delta(\varepsilon) \approx 1 - (1-\delta_{X/Y}(\infty))^k$
        $+ \sum_{\ell=\ell_\varepsilon}^{n-1} \big(1 - \mathrm{e}^{\varepsilon-(-L+\ell\Delta x)}\big)\, b_\ell^k.$
```

5 Error Analysis

We next give a bound for the error induced by Algorithm 1 which is determined by the parameter L. The total error consists of (see the supplementary material)

1. The tail integral $\int_L^\infty (\omega *^k \omega)(s)\, \mathrm{d}s$.
2. The error arising from periodisation of ω and truncation of the convolutions.

We obtain bounds for these two error sources using the Chernoff bound (Wainwright, 2019)

$$\mathbb{P}[X \geq t] \leq \frac{\mathbb{E}[\mathrm{e}^{\lambda X}]}{\mathrm{e}^{\lambda t}}$$

which holds for any random variable X and all $\lambda > 0$. Suppose $\omega_{X/Y}$ is of the form

$$\omega_{X/Y}(s) = \sum\nolimits_{i=0}^{n-1} a_{X,i} \cdot \delta_{s_i}(s), \tag{12}$$

where $s_i = \log\left(\frac{a_{X,i}}{a_{Y,i}}\right)$ and $a_{X,i}, a_{Y,i} > 0$. Then, the moment generating function of $\omega_{X/Y}$ is given by

$$\begin{aligned} \mathbb{E}[\mathrm{e}^{\lambda \omega_{X/Y}}] &= \int_{-\infty}^\infty \mathrm{e}^{\lambda s} \omega(s)\, \mathrm{d}s \\ &= \sum\nolimits_{i=0}^{n-1} \mathrm{e}^{\lambda s_i} \cdot a_{X,i} \\ &= \sum\nolimits_{i=0}^{n-1} \left(\frac{a_{X,i}}{a_{Y,i}}\right)^\lambda a_{X,i}. \end{aligned} \tag{13}$$

5.1 Connection to RDP

Suppose $f_X(t) = \sum_i a_{X,i} \cdot \delta_{t_i}(t)$, $f_Y(t) = \sum_i a_{Y,i} \cdot \delta_{t_i}(t)$ for some coefficients $a_{X,i}, a_{Y,i}$, and suppose $\omega_{X/Y}$ is of the form (12). Then, we have that

$$\mathbb{E}[\mathrm{e}^{\lambda \omega_{X/Y}}] = \lambda \cdot D_{\lambda+1}(f_X, f_Y),$$

where D_α denotes the Rényi divergence of order α (Mironov, 2017). Further, defining

$$\alpha(\lambda) := \log(\mathbb{E}[\mathrm{e}^{\lambda \omega_{X/Y}}]),$$

we see that $\alpha(\lambda)$ is exactly the logarithm of the moment generating function of the privacy loss function as defined, e.g., by Abadi et al. (2016) and Mironov et al. (2019). Thus existing Rényi differential privacy estimates for $\alpha(\lambda)$ could be used to bound the moment generating function of $\omega_{X/Y}$.

5.2 Tail Bound

Denote $S_k := \sum_{i=1}^k \omega_i$, where ω_i denotes the PLD random variable of the ith mechanism. If ω_i's are independent, we have that

$$\mathbb{E}[\mathrm{e}^{\lambda S_k}] = \prod_{i=1}^k \mathbb{E}[\mathrm{e}^{\lambda \omega_i}].$$

Then, if ω_i's are i.i.d. and distributed as ω, the Chernoff bound shows that for any $\lambda > 0$

$$\begin{aligned} \int_L^\infty (\omega *^k \omega)(s) \, \mathrm{d}s &= \mathbb{P}[S_k \geq L] \\ &\leq \prod_{i=1}^k \mathbb{E}[\mathrm{e}^{\lambda \omega_i}] \, \mathrm{e}^{-\lambda L} \\ &\leq \mathrm{e}^{k\alpha(\lambda)} \mathrm{e}^{-\lambda L}, \end{aligned} \tag{14}$$

where $\alpha(\lambda) = \log(\mathbb{E}[\mathrm{e}^{\lambda \omega}])$.

5.3 Total Error

We define $\alpha^+(\lambda)$ and $\alpha^-(\lambda)$ via the moment generating function of the PLD as

$$\alpha^+(\lambda) = \log\Big(\mathbb{E}[\mathrm{e}^{\lambda\omega}]\Big), \quad \alpha^-(\lambda) = \log\Big(\mathbb{E}[\mathrm{e}^{-\lambda\omega}]\Big).$$

Using the analysis given in the supplementary material, we bound the errors arising from the periodisation of the distribution and truncation of the convolutions. As a result, combining with (14), we obtain the following bound for the total error incurred by Algorithm 1.

Theorem 10. *Let ω be defined on the grid X_n as described above, let $\delta(\varepsilon)$ give the tight (ε, δ)-bound for ω and let $\widetilde{\delta}(\varepsilon)$ be the result of Algorithm 1. Then, for all $\lambda > 0$*

$$\begin{aligned} \left|\delta(\varepsilon) - \widetilde{\delta}(\varepsilon)\right| \leq &\Bigg(\frac{2\mathrm{e}^{(k+1)\alpha^+(\lambda)} - \mathrm{e}^{k\alpha^+(\lambda)} - \mathrm{e}^{\alpha^+(\lambda)}}{\mathrm{e}^{\alpha^+(\lambda)} - 1} \\ &+ \frac{\mathrm{e}^{(k+1)\alpha^-(\lambda)} - \mathrm{e}^{\alpha^-(\lambda)}}{\mathrm{e}^{\alpha^-(\lambda)} - 1} \Bigg) \frac{\mathrm{e}^{-L\lambda}}{1 - \mathrm{e}^{-L\lambda}}. \end{aligned}$$

Given a discrete-valued PLD distribution ω, we get strict lower and upper $\delta(\varepsilon)$-DP bounds as follows. Using parameter values $L > 0$ and $n \in \mathbb{Z}^+$, we form a grid X_n as defined in (7) and place ω on X_n to obtain ω^{L} and ω^{R} as defined in (8). We then approximate $\delta^L(\varepsilon)$ and $\delta^R(\varepsilon)$ using Algorithm 1. We estimate the error incurred by the approximation using Thm. 10 and the expressions given by Lemma 7. By subtracting this error from the approximation of $\delta^L(\varepsilon)$ and adding it to the approximation of $\delta^R(\varepsilon)$ and using Lemma 6, we obtain strict lower and upper bounds for $\delta(\varepsilon)$.

To obtain $\alpha^+(\lambda)$ and $\alpha^-(\lambda)$, we evaluate the moment generating functions $\mathbb{E}[\mathrm{e}^{\lambda\omega}]$ and $\mathbb{E}[\mathrm{e}^{-\lambda\omega}]$ using the finite sum (13). We use $\lambda = L/2$ in all experiments.

We emphasise that the error analysis is given in terms of the parameter L. The parameter n can be increased in case the resulting lower and upper bounds for $\delta(\varepsilon)$ are too far from each other.

6 Examples

6.1 The Exponential Mechanism

Consider the exponential mechanism $\mathcal{M}$ with quality score $u : \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$ and parameter $\widetilde{\varepsilon}$, i.e., an outcome y is sampled with probability

$$\mathbb{P}(\mathcal{M}(X) = y) = \frac{\mathrm{e}^{\widetilde{\varepsilon} u(X,y)}}{\sum_y \mathrm{e}^{\widetilde{\varepsilon} u(X,y)}}.$$

Consider the neighbouring relation $\sim_R$. Let u be a counting query, i.e.,

$$u(X, y) = \sum_{x \in X} \mathbf{1}(x = y),$$

and let $\mathcal{Y} = \{0, 1\}$. Denote by m the number of elements in X which equal 0. Let $Y \in \mathcal{X}^{n-1}$, $X \sim Y$, be such that $m-1$ elements equal 0. Then, the logarithmic ratio at $y = 0$ is given by

$$\begin{aligned} s_0 &:= \log\left(\frac{\mathbb{P}(\mathcal{M}(X) = 0)}{\mathbb{P}(\mathcal{M}(Y) = 0)}\right) \\ &= \log\left(\frac{\mathrm{e}^{\widetilde{\varepsilon} m}}{\mathrm{e}^{\widetilde{\varepsilon}(m-1)}} \frac{\mathrm{e}^{\widetilde{\varepsilon}(m-1)} + \mathrm{e}^{\widetilde{\varepsilon}(n-m)}}{\mathrm{e}^{\widetilde{\varepsilon} m} + \mathrm{e}^{\widetilde{\varepsilon}(n-m)}}\right) \end{aligned}$$

and similarly $s_1 = \log\left(\frac{\mathbb{P}(\mathcal{M}(X)=1)}{\mathbb{P}(\mathcal{M}(Y)=1)}\right)$. Using the values of $\mathbb{P}(\mathcal{M}(X) = i)$ and s_i, $i = 0, 1$, we obtain the PLD. We set $\widetilde{\varepsilon} = 0.05$ and $m = 50$. Figure 1 shows the $\delta(\varepsilon)$-values for $\varepsilon = 1.0$, when computed using Algorithm 1 for $\mathcal{M}(X)$ and $\mathcal{M}(Y)$ and the optimal bound (Dong et al., 2020, Thm. 2). The corresponding compute times are shown in Figure 2. The evaluation of the expression in (Dong et al., 2020, Thm. 2) is optimised using the logarithmic gamma function.

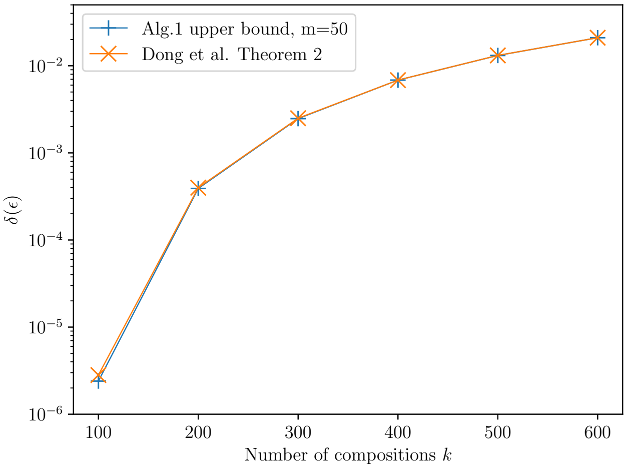

Figure 1: Exponential mechanism with a counting query quality score and parameter value $\varepsilon = 1.0$. We compute $\delta(\varepsilon)$ using Algorithm 1 and the optimal bound given by Dong et al. (2020, Thm. 2), for $\widetilde{\varepsilon} = 0.1$.

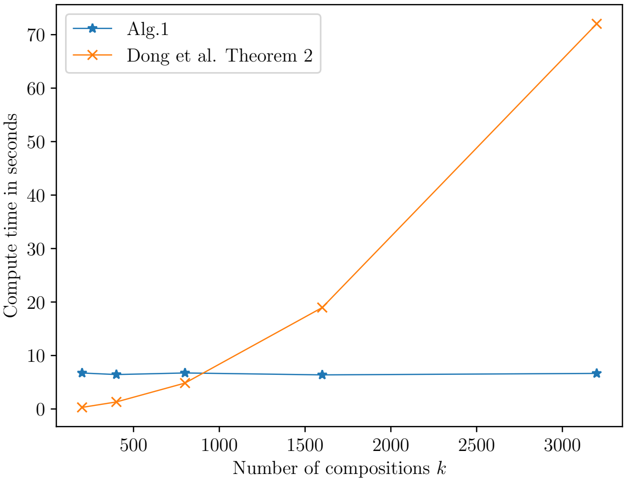

Figure 2: Compute times for different number of compositions k, using Algorithm 1 and the expression of Dong et al. (2020, Thm. 2) $\delta(\varepsilon)$. The evaluation of the expression by Dong et al. (2020, Thm. 2) is optimised using the logarithmic gamma function.

6.2 The Binomial Mechanism

The binomial mechanism by Agarwal et al. (2018) adds binomially distributed noise Z with parameters $n \in \mathbb{N}$ and $0 < p < 1$ to the output of a query f with output space $\mathbb{Z}^d$ as

$$\mathcal{M}(X) = f(X) + (Z - np) \cdot s,$$

where $s = 1/j$ for some $j \in \mathbb{N}$ and where for each coordinate i, $Z_i \sim \text{Bin}(n,p)$ and Z_i's are independent.

As described in the proof of Thm. 1 of Agarwal et al. (2018), for the privacy analysis of the binomial mechanism it is sufficient to consider the neighbouring binomial distributions centred at 0 and Δ. If, for example, $d = 1$, it is sufficient to consider the neighbouring binomial distributions

$$\begin{aligned} f_X(t) &= \sum\nolimits_{i=0}^n \binom{n}{i} p^i (1-p)^{n-i} \delta_{i+\Delta}(t), \\ f_Y(t) &= \sum\nolimits_{i=0}^n \binom{n}{i} p^i (1-p)^{n-i} \delta_i(t). \end{aligned}$$

Then, the privacy loss distribution $\omega_{X/Y}$ is of the form

$$\begin{aligned} \omega_{X/Y}(s) &= \sum\nolimits_{i=\Delta}^{n-\Delta} a_i \cdot \delta_{s_i}(s), \\ a_i &= \binom{n}{i} p^{i-\Delta} (1-p)^{n-i+\Delta}, \\ s_i &= \log \left(\frac{\binom{n}{i}}{\binom{n}{i-\Delta}} \left(\frac{1-p}{p} \right)^{\Delta} \right). \end{aligned}$$

Moreover,

$$\omega_{X/Y}(\infty) = \sum\nolimits_{i=n-\Delta+1}^n \binom{n}{i} p^i (1-p)^{n-i},$$

and determining the privacy loss distribution $\omega_{Y/X}$ can be done analogously.

The (ε, δ)-analysis of the multivariate binomial mechanism can be carried out via one-dimensional distributions using the following observation.

Theorem 11. *Consider a function $f : \mathcal{X}^N \to \mathbb{R}^d$ and a randomised mechanism $\mathcal{M}$ of the form $\mathcal{M}(X) = f(X) + Z$, where Z_i's are independent random variables. Suppose the data sets X and Y lead to the $\delta(\varepsilon)$-upper bound, and denote $\Delta = f(X) - f(Y)$. Then, the tight (ε, δ)-bound for $\mathcal{M}$ is given by the tight (ε, δ)-bound for the non-adaptive compositions of one-dimensional random variables*

$$\Delta_i + Z_i \quad \textit{and} \quad Z_i, \quad 1 \leq i \leq d.$$

Figure 3 illustrates how Algorithm 1 gives tighter bounds than the bound of Agarwal et al. (2018, Thm. 1), and also how the (ε, δ)-bound given by Algorithm 1 is close to the tight bound of the Gaussian mechanism for the corresponding variance (Analytical Gaussian mechanism by Balle and Wang, 2018). We use an example analogous to Agarwal et al. (2018, Fig. 1): we set $\Delta = \left[\frac{1}{10}, \ldots, \frac{1}{10}\right]^{\mathrm{T}} \in \mathbb{R}^{100}$, $p = 0.5$ and vary the parameters n and s. Using Thm. 11, we obtain tight (ε, δ)-bounds by considering a 100-fold compositions of one-dimensional mechanisms

$$\mathcal{M}(X) = \tfrac{1}{10} + (Z - np) \cdot s, \quad \mathcal{M}(Y) = (Z - np),$$

and thus we can use Algorithm 1 to obtain tight (ε, δ)-bounds for a single call of $\mathcal{M}(X)$.

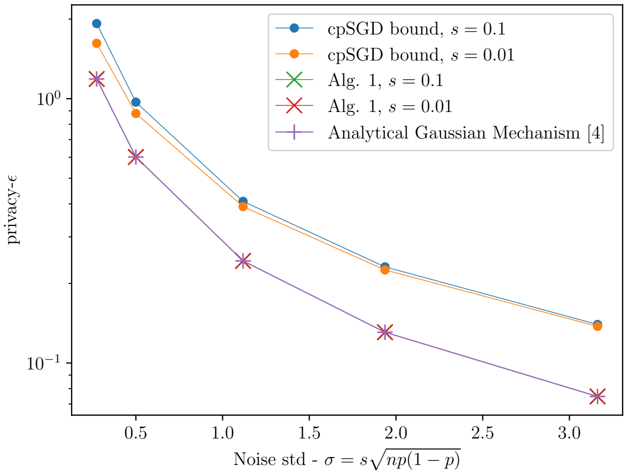

Figure 3: Comparison of the cpSGD bound (Agarwal et al., 2018, Thm. 1) and the upper bound given by Algorithm 1 ($\delta = 10^{-4}$, $p = 0.5$). The bound given by Algorithm 1 is close to that of the Analytical Gaussian mechanism (Balle and Wang, 2018).

Figure 4 shows results for an MNIST classification task, where we use a three-layer feedforward network with ReLUs and a hidden layer of width 60. DP-SGD approximation of the gradients is carried out such that for each per example gradient we use a sign approximation: the 200 largest elements (by magnitude) of the input layer are approximated by their sign and the rest are set to zero and similarly the 20 largest of the hidden layer and the largest one of the output layer. Elementwise zero centred binomial noise with parameters n and $p = 0.5$ is then added to the averaged gradients. By Thm. 11 and subsampling amplification (Sec. 3.4), the (ε, δ)-bound can be obtained by running Algorithm 1 for the PLD determined by the distributions

$$q \cdot f_X + (1-q) \cdot f_Y, \quad \text{and} \quad f_Y,$$

where f_X and f_Y are the density functions of the random variables

$$X \sim \mathbf{1} + (Z - np) \quad \text{and} \quad Y \sim (Z - np),$$

where $\mathbf{1} = \begin{bmatrix}1, \ldots, 1\end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{221}$ and for each i, $Z_i \sim \mathrm{Bin}(n,p)$ and Z_i's are independent. Here q denotes the subsampling ratio, i.e., $q = |B|/M$, where $|B|$ is the minibatch size and M the total size of the training data. We obtain tight (ε, δ)-bounds for the training of the network as follows (details in the Supplements). We obtain the PLD ω determined by the distributions $q \cdot f_X + (1-q) \cdot f_Y$ and f_Y from the PLD determined by f_X and f_Y (that is obtained using Thm. 11 and Alg. 1, as in the example of Figure 3). We then apply Algorithm 1 to ω, for a given number of compositions. The results of Figure 4 are averages of 5 runs. We set the initial learning rate $\eta = 0.02$. We linearly decrease the learning rate η after each epoch such that it is zero at the end of the training (when $|B| = 500$ starting from epoch 13, and when $|B| = 300$ starting from epoch 5). We compare this method to cpSGD (Agarwal et al., 2018) applied to Infinite MNIST data set which has the same test data set as MNIST. The results for cpSGD are extracted from Agarwal et al. (2018, Fig. 2). For $\varepsilon = 2.0$ we extract the result where each element of the gradient requires 8 bits and for $\varepsilon = 4.0$ the one requiring 16 bits. We note that when $n = 3000$ our method requires 12 bits per element.

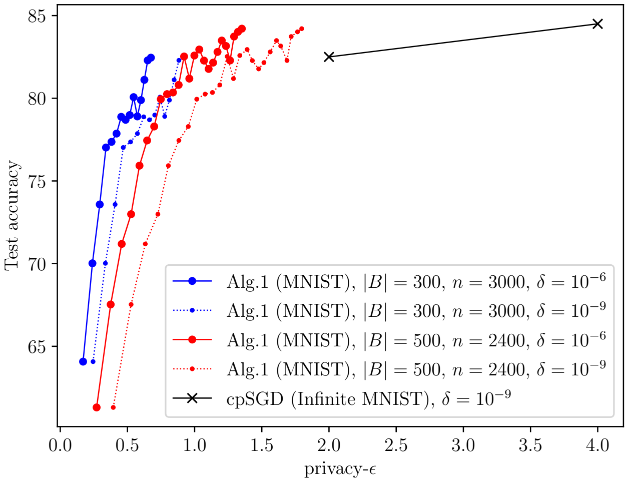

Figure 4: A small feedforward model run on MNIST ($M = 6 \cdot 10^4$) using Algorithm 1 and on Infinite MNIST ($M = 2.5 \cdot 10^8$) using cpSGD (Agarwal et al., 2018). Algorithm 1 takes into account the subsampling amplification (Sec. 3.4).

6.3 The Subsampled Gaussian Mechanism

We next show how to compute rigorous DP bounds for the subsampled Gaussian mechanism using the method presented here. We consider the Poisson subsampling and $\sim_R$-neighbouring relation. For a subsampling ratio q and noise level σ, the continuous PLD is given by Koskela et al. (2020)

$$\omega(s) = \begin{cases} f(g(s))g'(s), & \text{if } s > \log(1-q), \\ 0, & \text{otherwise,} \end{cases} \tag{15}$$

where

$$f(t) = \frac{1}{\sqrt{2\pi\sigma^2}} \left[q\mathrm{e}^{\frac{-(t-1)^2}{2\sigma^2}} + (1-q)\mathrm{e}^{-\frac{t^2}{2\sigma^2}}\right]$$

and

$$g(s) = \sigma^2 \log\left(\frac{\mathrm{e}^s - (1-q)}{q}\right) + \frac{1}{2}.$$

Let $L > 0$, $n \in \mathbb{Z}^+$, $\Delta x = 2L/n$ and $s_i = -L + i\Delta x$ for all $i \in \mathbb{Z}$. We define

$$\begin{aligned} \omega_{\text{min}}(s) &= \sum\nolimits_{i=0}^{n-1} c_i^- \cdot \delta_{s_i}(s), \\ \omega_{\text{max}}(s) &= \sum\nolimits_{i=0}^{n-1} c_i^+ \cdot \delta_{s_i}(s), \end{aligned} \tag{16}$$

where

$$\begin{aligned} c_i^- &= \Delta x \cdot \min\nolimits_{s \in [s_i, s_{i+1}]} \omega(s), \\ c_i^+ &= \Delta x \cdot \max\nolimits_{s \in [s_{i-1}, s_i]} \omega(s). \end{aligned}$$

Furthermore, we define

$$\begin{aligned} \omega_{\text{min}}^\infty(s) &= \sum_{i \in \mathbb{Z}} c_i^- \cdot \delta_{s_i}(s), \\ \omega_{\text{max}}^\infty(s) &= \sum_{i \in \mathbb{Z}} c_i^+ \cdot \delta_{s_i}(s). \end{aligned} \tag{17}$$

We find that ω as defined in (15) has one stationary point which we determine numerically. Using this fact, the numerical values of c_i^- and c_i^+ can be straightforwardly computed.

We obtain approximations for the lower and upper bounds $\delta_{\text{min}}(\varepsilon)$ and $\delta_{\text{max}}(\varepsilon)$ by running Algorithm 1 for $\omega_{\text{min}}^\infty$ and $\omega_{\text{max}}^\infty$ using some prescribed parameter values n and L:

Lemma 12. *Let $\delta(\varepsilon)$ be given by the integral formula of Thm. 5 for some privacy loss distribution ω. Let $\delta_{\text{min}}^\infty(\varepsilon)$ and $\delta_{\text{max}}^\infty(\varepsilon)$ be defined analogously by $\omega_{\text{min}}^\infty$ and $\omega_{\text{max}}^\infty$. Then for all $\varepsilon > 0$ we have*

$$\delta_{\text{min}}^\infty(\varepsilon) \leq \delta(\varepsilon) \leq \delta_{\text{max}}^\infty(\varepsilon).$$

Proof. Supplements. □

Running Alg. 1 for $\omega_{\text{min}}^\infty$ and $\omega_{\text{max}}^\infty$ is equivalent to running it for the truncated distributions ω_{min} and ω_{max}. However, to obtain the bounds of Thm. 5 (and subsequently strict bounds for $\delta(\varepsilon)$), the analysis has to be carried out for $\omega_{\text{min}}^\infty$ and $\omega_{\text{max}}^\infty$. To this end, we need bounds for the moment generating functions of $-\omega_{\text{min}}^\infty$, $\omega_{\text{min}}^\infty$ $-\omega_{\text{max}}^\infty$ and $\omega_{\text{max}}^\infty$ (where $-\omega(s) := \sum_i a_i \cdot \delta_{-s_i}(s)$ if $\omega(s) = \sum_i a_i \cdot \delta_{s_i}(s)$). We can bound the moment generating function of $\omega_{\text{max}}^\infty$ as follows. We note that $\mathbb{E}[\mathrm{e}^{\lambda \omega_{\text{max}}}]$ can be evaluated numerically.

Lemma 13. *Let $0 < \lambda \leq L$ and assume $\sigma \geq 1$ and $\Delta x \leq c \cdot L$, $0 < c < 1$. Let ω_{max} and $\omega_{\text{max}}^\infty$ be defined as in (16) and (17). The moment generating function of $\omega_{\text{max}}^\infty$ can be bounded as*

$$\mathbb{E}[\mathrm{e}^{\lambda \omega_{\text{max}}^\infty}] \leq \mathbb{E}[\mathrm{e}^{\lambda \omega_{\text{max}}}] + \text{err}(\lambda, L, \sigma),$$

where

$$\text{err}(\lambda, L, \sigma) = \\ \mathrm{e}^{c\lambda L} \frac{2}{\sqrt{\pi}} \mathrm{e}^{-\frac{\lambda(2C-\lambda)}{2\sigma^2}} \text{erfc}\left(\frac{(1-c)\sigma^2 L + C - \lambda}{\sqrt{2}\sigma} \right)$$

and $C = \sigma^2 \log(\frac{1}{2q}) - \frac{1}{2}$.

Proof. Supplements. □

An analogous bound holds for the moment generating functions of $-\omega_{\text{min}}^\infty$, $\omega_{\text{min}}^\infty$ and $-\omega_{\text{max}}^\infty$ (see the Supplements). In the experiments, the effect of the error term $\text{err}(\lambda, L, \sigma)$ was found to be negligible.

Figure 5 illustrates the convergence of the bound given by Lemma 12 as n grows and L is fixed. For comparison, we also show the numerical values given by Tensorflow moments accountant (Abadi et al., 2016).

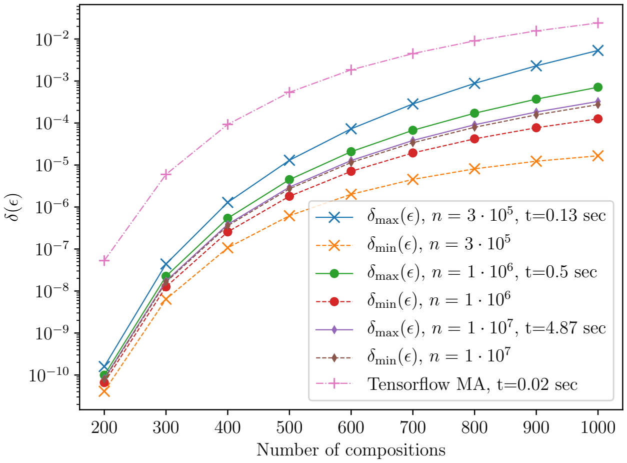

Figure 5: The subsampled Gaussian mechanism and bounds for $\delta(\varepsilon)$ computed using Algorithm 1, when $\varepsilon = 1.0$, $q = 0.02$, $\sigma = 2.0$ and $L = 8.0$. Here n denotes the number of discretisation points. Compute times are for each curve.

7 Conclusions

We have presented a novel approach for computing privacy bounds for discrete-valued mechanisms. The method provides tools for moments-accountant-like techniques for evaluating privacy bounds for discrete output DP-SGD algorithms. More specifically, we have shown how to accurately bound the $\delta(\varepsilon)$-DP for the subsampled binomial mechanism, when the gradients are replaced with a sign approximation. Moreover, as the example of Section 6.3 shows, accurate (ε, δ)-bounds for continuous mechanisms can also be obtained using the proposed method. Due to the rigorous error analysis the reported (ε, δ)-bounds are strict lower and upper privacy bounds.

Acknowledgements

This work has been supported by the Academy of Finland [Finnish Center for Artificial Intelligence FCAI and grants 319264, 325572, 325573].

References

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

Agarwal, N., Suresh, A. T., Yu, F. X. X., Kumar, S., and McMahan, B. (2018). cpSGD: Communication-efficient and differentially-private distributed SGD. In *Advances in Neural Information Processing Systems*, pages 7564–7575.

Balle, B., Barthe, G., and Gaboardi, M. (2018). Privacy amplification by subsampling: Tight analyses via couplings and divergences. In *Advances in Neural Information Processing Systems*, pages 6277–6287.

Balle, B. and Wang, Y.-X. (2018). Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *International Conference on Machine Learning*, pages 394–403.

Barthe, G. and Olmedo, F. (2013). Beyond differential privacy: composition theorems and relational logic for f-divergences between probabilistic programs. In *Proceedings of the 40th international conference on Automata, Languages, and Programming-Volume Part II*, pages 49–60.

Cooley, J. W. and Tukey, J. W. (1965). An algorithm for the machine calculation of complex Fourier series. *Mathematics of computation*, 19(90):297–301.

Dong, J., Durfee, D., and Rogers, R. (2020). Optimal differential privacy composition for exponential mechanisms. In *International Conference on Machine Learning*.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In *Proc. TCC 2006*, pages 265–284.

Dwork, C. and Roth, A. (2014). The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3–4):211–407.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al. (2019). Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*.

Koskela, A., Jälkö, J., and Honkela, A. (2020). Computing tight differential privacy guarantees using FFT. In *The 23rd International Conference on Artificial Intelligence and Statistics*.

Meiser, S. and Mohammadi, E. (2018). Tight on budget?: Tight bounds for r-fold approximate differential privacy. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 247–264. ACM.

Mironov, I. (2012). On significance of the least significant bits for differential privacy. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 650–661.

Mironov, I. (2017). Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275.

Mironov, I., Talwar, K., and Zhang, L. (2019). Rényi differential privacy of the sampled Gaussian mechanism. *arXiv preprint arXiv:1908.10530*.

Sommer, D. M., Meiser, S., and Mohammadi, E. (2019). Privacy loss classes: The central limit theorem in differential privacy. *Proceedings on Privacy Enhancing Technologies*, 2019(2):245–269.

Stockham Jr, T. G. (1966). High-speed convolution and correlation. In *Proceedings of the April 26-28, 1966, Spring joint computer conference*, pages 229–233. ACM.

Stoer, J. and Bulirsch, R. (2013). *Introduction to numerical analysis*, volume 12. Springer Science & Business Media.

Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press.

Wang, Y.-X., Balle, B., and Kasiviswanathan, S. P. (2019). Subsampled Rényi differential privacy and analytical moments accountant. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1226–1235.

Warner, S. L. (1965). Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69.

Zhu, Y. and Wang, Y.-X. (2019). Poisson subsampled Rényi differential privacy. In *International Conference on Machine Learning*, pages 7634–7642.