

# Sharp Composition Bounds for Gaussian Differential Privacy via Edgeworth Expansion

Qinqing Zheng\*      Jinshuo Dong†      Qi Long‡      Weijie J. Su§

*University of Pennsylvania*

February 27, 2020

## Abstract

Datasets containing sensitive information are often sequentially analyzed by many algorithms. This raises a fundamental question in differential privacy regarding how the overall privacy bound degrades under composition. To address this question, we introduce a family of analytical and sharp privacy bounds under composition using the Edgeworth expansion in the framework of the recently proposed  $f$ -differential privacy. In contrast to the existing composition theorems using the central limit theorem, our new privacy bounds under composition gain improved tightness by leveraging the refined approximation accuracy of the Edgeworth expansion. Our approach is easy to implement and computationally efficient for any number of compositions. The superiority of these new bounds is confirmed by an asymptotic error analysis and an application to quantifying the overall privacy guarantees of noisy stochastic gradient descent used in training private deep neural networks.

## 1 Introduction

Machine learning, data mining, and statistical analysis are widely applied to various applications impacting our daily lives. While we celebrate the benefits brought by these applications, to an alarming degree, the algorithms are accessing datasets containing sensitive information such as individual behaviors on the web and health records. By simply tweaking the datasets and leveraging the output of algorithms, it is possible for an adversary to learn information about and even identify certain individuals [FJR15, SSSS17]. In particular, privacy concerns become even more acute when the same dataset is probed by a sequence of algorithms. With knowledge of the dataset from the prior algorithms' output, an adversary can adaptively analyze the dataset to cause additional privacy loss at each round. This reality raises one of the most fundamental problems in the area of private data analysis:

*How can we accurately and efficiently quantify the cumulative privacy loss under composition of private algorithms?*

---

\*Department of Statistics. Email: zhengqinqing@gmail.com.

†Graduate Group in Applied Mathematics and Computational Science. Email: jinshuo@sas.upenn.edu.

‡Department of Biostatistics, Epidemiology and Informatics. Email: qlong@penmedicine.upenn.edu.

§Department of Statistics. Email: suw@wharton.upenn.edu.

To address this important problem, one has to start with a formal privacy definition. To date, the most popular statistical privacy definition is  $(\epsilon, \delta)$ -differential privacy (DP) [DKM<sup>+</sup>06, DMNS06], with numerous deployments in both industrial applications and academic research [EPK14, ACG<sup>+</sup>16, PAE<sup>+</sup>16, DKY17, App17, Abo18]. Informally, this privacy definition requires an unnoticeable change in a (randomized) algorithm’s output due to the replacement of any individual in the dataset. More concretely, letting  $\epsilon \geq 0$  and  $0 \leq \delta \leq 1$ , an algorithm  $M$  is  $(\epsilon, \delta)$ -differentially private if for any pair of neighboring datasets  $S, S'$  (in the sense that  $S$  and  $S'$  differs in only one individual) and any event  $E$ ,

$$\mathbb{P}(M(S) \in E) \leq e^\epsilon \mathbb{P}(M(S') \in E) + \delta. \quad (1.1)$$

Unfortunately, this privacy definition comes with a drawback when handling composition. Explicitly,  $(\epsilon, \delta)$ -DP is *not closed* under composition in the sense that the overall privacy bound of a sequence of (at least two)  $(\epsilon, \delta)$ -DP algorithms cannot be precisely described by a single pair of the parameters  $\epsilon, \delta$  [KOV17]. Although the precise bound can be collectively represented by infinitely many pairs of  $\epsilon, \delta$ , [MV16] shows that it is computationally hard to find such a pair of privacy parameters.

The need for a better treatment of composition has triggered a surge of interest in proposing generalizations of  $(\epsilon, \delta)$ -DP, including divergence-based relaxations [DR16, BS16, Mir17, BDRS18] and, more recently, a hypothesis testing-based extension termed  $f$ -differential privacy ( $f$ -DP) [DRS19]. This privacy definition leverages the hypothesis testing interpretation of differential privacy, and characterizes the privacy guarantee using the trade-off between type I and type II errors given by the associated hypothesis testing problem. As an advantage over the divergence-based privacy definitions, among others,  $f$ -DP allows for a concise and sharp argument for privacy amplification by subsampling. More significantly,  $f$ -DP is accompanied with a technique powered by the central limit theorem (CLT) for analyzing privacy bounds under composition of a large number of private algorithms. Loosely speaking, the overall privacy bound asymptotically converges to the trade-off function defined by testing between two normal distributions. This class of trade-off functions gives rise to *Gaussian differential privacy* (GDP), a subfamily of  $f$ -DP guarantees.

In deploying differential privacy, however, the number of private algorithms under composition may be *moderate* or *small* (see such applications in private sparse linear regression [KST12] and personalized online advertising [LO11]). In this regime, the CLT phenomenon does not kick in and, as such, the composition bounds developed using CLT can be inaccurate [DRS19, BDLS19]. To address this practically important problem, in this paper we develop sharp and analytical composition bounds in  $f$ -DP without assuming a larger number of algorithms, by leveraging the Edgeworth expansion [Hal13]. The Edgeworth expansion is a technique for approximating probability distributions in terms of their cumulants. Compared with the CLT approach, in our setting, this technique enables a significant reduction of approximation errors for composition theorems.

In short, our Edgeworth expansion-powered privacy bounds have a number of appealing properties, which will be shown in this paper through both theoretical analysis and numerical examples.

- The Edgeworth expansion is a more general approach that subsumes the CLT-based approximation. Moreover, our new privacy bounds tighten the composition bounds that are developed in the prior art [DRS19, BDLS19].
- Our method is easy to implement and computationally efficient. In the case where all trade-off functions are identical under composition, the computational cost is constant regardless of the number of private algorithms. This case is not uncommon and can be found, for example, in

the privacy analysis of noisy stochastic gradient descent (SGD) used in training deep neural networks.

The remainder of the paper is organized as follows. Section 2 reviews the  $f$ -DP framework. Section 3 introduces our methods based on the Edgeworth expansion. Section 4 provides a two-parameter interpretation of the Edgeworth approximation-based privacy guarantees. Finally, we present experimental results to demonstrate the superiority of our approach in Section 5.

## 2 Preliminaries on $f$ -Differential Privacy

Let a randomized algorithm  $M$  take a dataset  $S$  as input. Leveraging the output of this algorithm, differential privacy seeks to measure the difficulty of identifying the presence or absence of any individual in  $S$ . The  $(\epsilon, \delta)$ -DP definition offers such a measure using the probabilities that  $M$  gives the same outcome for two neighboring datasets  $S$  and  $S'$ . A more concrete description is as follows. Let  $P$  and  $Q$  denote the probability distribution of  $M(S)$  and  $M(S')$ , respectively. To breach the privacy, in essence, an adversary performs the following hypothesis testing problem:

$$H_0 : \text{output} \sim P \text{ vs } H_1 : \text{output} \sim Q.$$

The privacy guarantee of  $M$  boils down to the extent to which the adversary can tell the two distributions apart. In the case of  $(\epsilon, \delta)$ -DP, the privacy guarantee is expressed via 1.1. The relationship between differential privacy and hypothesis testing is first studied in [WZ10, KOV17, LHC<sup>+</sup>19, BBG<sup>+</sup>19]. More recently, [DRS19] proposes to use the trade-off between type I and type II errors of the optimal likelihood ratio tests at level ranging from 0 to 1 as a measure of the privacy guarantee. Note that the optimal tests are given by the Neyman–Pearson lemma, and can be thought of as the most powerful adversary.

**Trade-off function.** Let  $\phi$  be a rejection rule for testing against  $H_0$  against  $H_1$ . The type I and type II error of  $\phi$  are  $\mathbb{E}_P(\phi)$  and  $1 - \mathbb{E}_Q(\phi)$ , respectively. The trade-off function  $T : [0, 1] \rightarrow [0, 1]$  between the two probability distributions  $P$  and  $Q$  is defined as

$$T(P, Q)(\alpha) = \inf_{\phi} \{1 - \mathbb{E}_Q(\phi) : \mathbb{E}_P(\phi) \leq \alpha\}.$$

That is,  $T[P, Q](\alpha)$  equals the minimum type II error that one can achieve at significance level  $\alpha$ . A larger trade-off function corresponds to a more difficult hypothesis testing problem, thereby implying more privacy of the associated private algorithm. When the two distributions are the same, the perfect privacy is achieved and the corresponding trade-off function is  $T(P, P)(\alpha) = 1 - \alpha$ . In the sequel, we denote this function by  $\text{Id}(\alpha)$ . With the definition of trade-off functions in place, [DRS19] introduces the following privacy definition (we say  $f \geq g$  if  $f(\alpha) \geq g(\alpha)$  for all  $0 \leq \alpha \leq 1$ ):

**Definition 2.1.** Let  $f$  be a trade-off function. An algorithm  $M$  is  $f$ -differentially private if  $T(M(S), M(S')) \geq f$  for any pair of neighboring datasets  $S$  and  $S'$ .

While the definition above considers a general trade-off function, it is worthwhile noting that  $f$  can always be assumed to be *symmetric*. Letting  $f^{-1}(\alpha) := \inf\{0 \leq t \leq 1 : f(t) \leq \alpha\}$  (note that  $f^{-1} = T(Q, P)$  if  $f = T(P, Q)$ ), a trade-off function  $f$  is said to be symmetric if  $f = f^{-1}$ . Due to the symmetry of the two neighboring datasets in the privacy definition, an  $f$ -DP algorithm must be

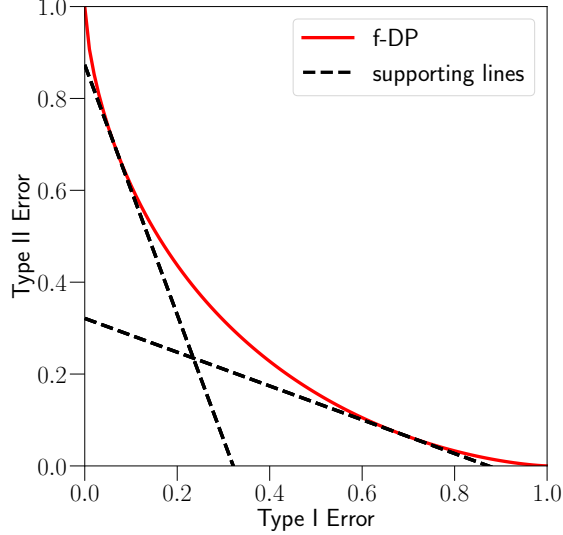


Figure 2.1: An example of a trade-off function and two supporting lines induced by the associated  $(\varepsilon, \delta(\varepsilon))$ -DP guarantees. These lines have slopes  $-e^{\pm\varepsilon}$ , respectively, and intercepts  $1 - \delta$ .

$\max\{f, f^{-1}\}$ -DP. Compared to  $f$ , the new trade-off function  $\max\{f, f^{-1}\}$  is symmetric and gives a greater or equal privacy guarantee. For the special case where the lower bound in Definition 2.1 is a trade-off function between two Gaussian distributions, we say that the algorithm has *Gaussian differential privacy* (GDP):

**Definition 2.2.** Let  $G_\mu := T(\mathcal{N}(0, 1), \mathcal{N}(\mu, 1)) \equiv \Phi(\Phi^{-1}(1 - \alpha) - \mu)$  for some  $\mu \geq 0$ , where  $\Phi$  denotes the cumulative distribution function (CDF) of the standard normal distribution. An algorithm  $M$  gives  $\mu$ -GDP if  $T(M(S), M(S')) \geq G_\mu$  for any pair of neighboring datasets  $S$  and  $S'$ .

**Duality to  $(\varepsilon, \delta)$ -DP.** The  $f$ -DP framework has a dual relationship with  $(\varepsilon, \delta)$ -DP in the sense that  $f$ -DP is equivalent to an infinite collection of  $(\varepsilon, \delta)$ -DP guarantees via the convex conjugate of  $f$ . One can view  $f$ -DP as the primal representation of privacy, and accordingly, its dual representation is the collection of  $(\varepsilon, \delta)$ -DP guarantees. In this paper, the Edgeworth approximation addresses  $f$ -DP from the primal perspective. However, it is also instructive to check the dual presentation. The following propositions introduce how to convert the primal to the dual, and vice versa. Geometrically, each associated  $(\varepsilon, \delta)$ -DP guarantee defines two symmetric supporting linear functions to  $f$  (assuming  $f$  is symmetric). See Figure 2.1.

**Proposition 2.3 (Primal to Dual).** For a symmetric trade-off function  $f$ , let  $f^* : \mathbb{R} \rightarrow \mathbb{R}$  be its convex conjugate function  $f^*(y) = \sup_{0 \leq x \leq 1} yx - f(x)$ . A mechanism is  $f$ -DP if and only if it is  $(\varepsilon, \delta(\varepsilon))$ -DP for all  $\varepsilon \geq 0$  with  $\delta(\varepsilon) = 1 + f^*(-e^\varepsilon)$ .

**Proposition 2.4 (Dual to Primal).** Let  $I$  be an arbitrary index set such that each  $i \in I$  is associated with  $\varepsilon_i \in [0, \infty)$  and  $\delta_i \in [0, 1]$ . A mechanism is  $(\varepsilon_i, \delta_i)$ -DP for all  $i \in I$  if and only if it is  $f$ -DP with  $f = \sup_{i \in I} f_{\varepsilon_i, \delta_i}$ , where  $f_{\varepsilon, \delta}(\alpha) = \max\{0, 1 - \delta - e^\varepsilon \alpha, e^{-\varepsilon}(1 - \delta - \alpha)\}$  is the trade-off function corresponding to  $(\varepsilon, \delta)$ -DP.

Next, we introduce how  $f$ -DP guarantees degrade under composition. With regard to composition, SGD offers an important benchmark for testing a privacy definition. As a popular optimizer for training deep neural networks, SGD outputs a series of models that are generated from the *composition* of many gradient descent updates. Furthermore, each step of update is computed from a *subsampling* mini-batch of data points. While composition degrades the privacy, in contrast, subsampling amplifies the privacy as individuals uncollected in the mini-batch have perfect privacy. Quantifying these two operations under the  $f$ -DP framework is crucial for analyzing the privacy guarantee of deep learning models trained by noisy SGD.

**Composition.** Let  $f_1 = T(P_1, Q_1)$  and  $f_2 = T(P_2, Q_2)$ , [DRS19] defines a binary operator  $\otimes$  on trade-off functions such that  $f_1 \otimes f_2 = T(P_1 \times P_2, Q_1 \times Q_2)$ , where  $P_1 \times P_2$  is the distribution product. This operator is commutative and associative. The composition primitive refers to an algorithm  $M$  that consists of  $n$  algorithms  $M_1, \dots, M_n$ , where  $M_i$  observes both the input dataset and output from all previous algorithms<sup>1</sup>. In [DRS19], it is shown that if  $M_i$  is  $f_i$ -DP for  $1 \leq i \leq n$ , then the composed algorithm  $M$  is  $f_1 \otimes \dots \otimes f_n$ -DP. The authors further identify a central limit theorem-type phenomenon of the overall privacy loss under composition. Loosely speaking, the privacy guarantee asymptotically converges to GDP in the sense that  $f_1 \otimes \dots \otimes f_n \rightarrow G_\mu$  as  $n \rightarrow \infty$  under certain conditions. The privacy parameter  $\mu$  depends on the trade-off functions  $f_1, \dots, f_n$ .

**Subsampling.** Consider the operator  $\text{Sample}_p$  that includes each individual in the dataset with probability  $p$  independently. Let  $M \circ \text{Sample}_p$  denote the algorithm where  $M$  is applied to the subsampled dataset. In the subsampling theorem for  $f$ -DP, [DRS19] proves that if  $M$  is  $f$ -DP, then  $M \circ \text{Sample}_p(S)$  is  $\tilde{f}$ -DP if  $\tilde{f} \leq f_p$  and  $\tilde{f} \leq f_p^{-1}$ , where  $f_p = pf + (1-p)\text{Id}$ . As such, we can take  $\tilde{f} = \min\{f_p, f_p^{-1}\}$ , which is not convex in general however. This issue can be resolved by using  $\min\{f_p, f_p^{-1}\}^{**}$  in place of  $\min\{f_p, f_p^{-1}\}$ , where  $f^{**}$  denotes the double conjugate of  $f$ . Indeed, [DRS19] shows that the subsampled algorithm is  $\min\{f_p, f_p^{-1}\}^{**}$ -DP.

**Noisy SGD.** Let  $M_\sigma$  denote the noisy gradient descent update, where  $\sigma$  is the scale of the Gaussian noise added to the gradient. The noisy SGD update can essentially be represented as  $M_\sigma \circ \text{sample}_p$ . Exploiting the above results for composition and subsampling, [BDLS19] shows that  $M_\sigma \circ \text{Sample}_p$  is  $\min\{h, h^{-1}\}$ -DP, where  $h = pG_{1/\sigma} + (1-p)\text{Id}$ . Recognizing that noisy SGD with  $n$  iterations is the  $n$ -fold composition of  $M_\sigma \circ \text{Sample}_p(S)$ , the overall privacy lower bound is  $\min\{g, g^{-1}\}$ -DP, where  $g = h^{\otimes n} = (pG_{1/\sigma} + (1-p)\text{Id})^{\otimes n}$ . To evaluate the composition bound, [BDLS19] uses a central limit theorem-type result in the asymptotic regime where  $p\sqrt{n}$  converges to a positive constant as  $n \rightarrow \infty$ : in this regime, one can show  $g \rightarrow G_{p\sqrt{n(e^{1/\sigma^2}-1)}}$  and consequently  $\min\{g, g^{-1}\}^{**} \rightarrow G_{p\sqrt{n(e^{1/\sigma^2}-1)}}$  as well.

### 3 Edgeworth Approximation

In this section, we introduce the Edgeworth expansion-based approach to computing the privacy bound under composition. The development of this approach builds on [DRS19], with two crucial modifications.

---

<sup>1</sup>In this paper,  $n$  denotes the number of private algorithms under composition, as opposed to the number of individuals in the dataset. This is to be consistent with the literature on central limit theorems.

Consider the hypothesis testing problem  $H_0 : \mathbf{x} \sim P_1 \times \cdots \times P_n$  vs  $H_1 : \mathbf{x} \sim Q_1 \times \cdots \times Q_n$ . Let  $P$  denote the distribution  $P_1 \times \cdots \times P_n$ , and  $p_i(\cdot)$  denote the probability density functions of  $P_i$ . Correspondingly, we define  $Q$  and  $q_i$  in the same way. Letting  $L_i(\cdot) = \log \frac{q_i(\cdot)}{p_i(\cdot)}$ , the likelihood ratio

test statistic is given by  $T_n(\mathbf{x}) = \log \frac{\prod_{i=1}^n q_i(x_i)}{\prod_{i=1}^n p_i(x_i)} = \sum_{i=1}^n L_i(x_i)$ . The Neyman–Pearson lemma states that the most powerful test at a given significance level  $\alpha$  must be a thresholding function of  $T_n(\mathbf{x})$ . As a result, the optimal rejection rule would reject  $H_0$  if  $T_n(\mathbf{x}) > \eta$ , where  $\eta$  is determined by  $\alpha$ . An equivalent rule is to apply thresholding to the standardized statistic:  $H_0$  is rejected if

$$\frac{T_n(\mathbf{x}) - \mathbb{E}_P[T_n(\mathbf{x})]}{\sqrt{\text{Var}_P[T_n(\mathbf{x})]}} > h(\alpha), \quad (3.1)$$

where the threshold  $h$  is determined by  $\alpha$ .

In the sequel, for notational simplicity we shall use  $T_n$  to denote  $T_n(\mathbf{x})$ , though it is a function of  $\mathbf{x}$ . Let  $F_n(\cdot)$  be the CDF of  $\frac{T_n - \mathbb{E}_P[T_n]}{\sqrt{\text{Var}_P[T_n]}}$  when  $\mathbf{x}$  is drawn from  $P$ . That is,  $F_n(h) = \mathbb{P}_P\left(\frac{T_n - \mathbb{E}_P[T_n]}{\sqrt{\text{Var}_P[T_n]}} \leq h\right)$ . By the Lyapunov CLT, the standardized statistic  $\frac{T_n - \mathbb{E}_P[T_n]}{\sqrt{\text{Var}_P[T_n]}}$  converges in distribution to the standard normal random variable. In other words, it holds that

$$\frac{T_n - \mathbb{E}_P[T_n]}{\sqrt{\text{Var}_P[T_n]}} \xrightarrow{d} \mathcal{N}(0, 1), \quad F_n(\cdot) \rightarrow \Phi(\cdot)$$

as  $n \rightarrow \infty$ . Likewise, we write  $\tilde{F}_n(h) = \mathbb{P}_Q\left(\frac{T_n - \mathbb{E}_Q[T_n]}{\sqrt{\text{Var}_Q[T_n]}} \leq h\right)$  with  $\mathbf{x} \sim Q$  and get

$$\frac{T_n - \mathbb{E}_Q[T_n]}{\sqrt{\text{Var}_Q[T_n]}} \xrightarrow{d} \mathcal{N}(0, 1), \quad \tilde{F}_n(\cdot) \rightarrow \Phi(\cdot).$$

With these notations in place, one can write the type I error of the rejection rule (3.1) as

$$\alpha = \mathbb{P}_P\left(\frac{T_n - \mathbb{E}_P[T_n]}{\sqrt{\text{Var}_P[T_n]}} > h\right) = 1 - F_n(h). \quad (3.2)$$

The type II error of this test, which is  $f(\alpha)$  by definition, is given by

$$\begin{aligned} f(\alpha) &= \mathbb{P}_Q\left(\frac{T_n - \mathbb{E}_P[T_n]}{\sqrt{\text{Var}_P[T_n]}} \leq h\right) \\ &= \mathbb{P}_Q\left(\frac{T_n - \mathbb{E}_Q[T_n]}{\sqrt{\text{Var}_P[T_n]}} \leq h - \frac{\mathbb{E}_Q[T_n] - \mathbb{E}_P[T_n]}{\sqrt{\text{Var}_P[T_n]}}\right). \end{aligned}$$

In [DRS19], the authors assume that  $f$  is symmetric and therefore derive the identity  $\text{Var}_Q(T_n) = \text{Var}_P(T_n)$ . As a consequence,  $f(\alpha)$  can be written as  $\tilde{F}_n(h - \mu_n)$ , where  $\mu_n = (\mathbb{E}_Q[T_n] - \mathbb{E}_P[T_n])/\sqrt{\text{Var}_Q[T_n]}$ . Taken together, the equations above give rise to  $f(\alpha) = \tilde{F}_n(F_n^{-1}(1 - \alpha) - \mu_n)$ . Leveraging this expression of  $f$ , [DRS19] proves a CLT-type asymptotic convergence result under certain conditions:

$$f(\alpha) \rightarrow G_\mu(\alpha) = \Phi(\Phi^{-1}(1 - \alpha) - \mu) \quad (3.3)$$

as  $n \rightarrow \infty$ , where  $\mu$  is the limit of  $\mu_n$ .

Now, we discard the symmetry assumption and just rewrite

$$\begin{aligned}
f(\alpha) &= \mathbb{P}_Q \left( \frac{T_n - \mathbb{E}_P[T_n]}{\sqrt{\text{Var}_P[T_n]}} \leq h \right) \\
&= \mathbb{P}_Q \left( \frac{T_n - \mathbb{E}_Q[T_n]}{\sqrt{\text{Var}_P[T_n]}} \leq h - \frac{\mathbb{E}_Q[T_n] - \mathbb{E}_P[T_n]}{\sqrt{\text{Var}_P[T_n]}} \right) \\
&= \mathbb{P}_Q \left( \frac{T_n - \mathbb{E}_Q[T_n]}{\sqrt{\text{Var}_Q[T_n]}} \leq \left[ h - \frac{\mathbb{E}_Q[T_n] - \mathbb{E}_P[T_n]}{\sqrt{\text{Var}_P[T_n]}} \right] \sqrt{\frac{\text{Var}_P[T_n]}{\text{Var}_Q[T_n]}} \right) \\
&= \tilde{F}_n \left( (h - \mu_n) \sqrt{\frac{\text{Var}_P[T_n]}{\text{Var}_Q[T_n]}} \right).
\end{aligned} \tag{3.4}$$

Plugging Equation 3.2 into 3.4, we obtain

$$f(\alpha) = \tilde{F}_n \left( (F_n^{-1}(1 - \alpha) - \mu_n) \sqrt{\frac{\text{Var}_P[T_n]}{\text{Var}_Q[T_n]}} \right). \tag{3.5}$$

In the special case  $f$  is symmetric, the factor  $\sqrt{\frac{\text{Var}_P[T_n]}{\text{Var}_Q[T_n]}}$  is equal to one and we recover the result in [DRS19].

To obtain the composition bound, the exact computing of Equation 3.5 is not trivial. In Section 5.1 we present a numerical method to compute it directly; however, this method is computationally daunting and could not scale to a large number of compositions. The CLT estimator (Equation 3.3) can be computed quickly, however it can be loose for a small or moderate number of compositions. More importantly, in practice, we observe that the CLT estimator does not handle the composition of asymmetric trade-off functions well. To address these issues, we propose a two-sided approximation method, where the Edgeworth expansion is applied to both  $F_n$  and  $\tilde{F}_n$  in Equation 3.5. Our method leads to more accurate description of  $f(\alpha)$ , as justified in Section 5.

### 3.1 Technical Details

In Equation 3.5, we need to evaluate  $\tilde{F}_n$  and  $F_n^{-1}$ . Our methods for addressing each of them are described below.

**Approximate  $\tilde{F}_n$ .** Assume  $x \sim Q$ . Denote

$$X_Q = \frac{T_n - \mathbb{E}_Q[T_n]}{\sqrt{\text{Var}_Q(T_n)}} = \frac{\sum_{i=1}^n (L_i - \mu_i)}{\sqrt{\sum_{i=1}^n \sigma_i^2}}, \tag{3.6}$$

where  $\mu_i$  and  $\sigma_i^2$  are the mean and variance of  $L_i$  under distribution  $Q_i$ . Recall  $\tilde{F}_n$  is the CDF of  $X_Q$ , and we can apply the Edgeworth expansion to approximate  $\tilde{F}_n$  directly, following the techniques introduced in [Hal13]. It provides a family of series that approximate  $\tilde{F}_n$  in terms of the cumulants of  $X_Q$ , which can be derived from the cumulants of  $L_i$ s under distribution  $Q_i$ s. See Definition 3.1 for the notion of cumulant and Appendix B for how to compute them.



**Definition 3.1.** For a random variable  $X$ , let  $K(t)$  be the natural logarithm of the moment-generating function of  $X$ :  $K(t) = \log \mathbb{E}(e^{tX})$ . The cumulants of  $X$ , denoted by  $\kappa_0, \dots, \kappa_r, \dots$  for integer  $r > 0$ , are the coefficients in the Taylor expansion of  $K(t)$  about the origin:  $K(t) = \log \mathbb{E}(e^{tX}) = \sum_{r=0}^{\infty} \kappa_r t^r / r!$ .

The key idea of the Edgeworth approximation is to write the characteristic function of the distribution of  $X_Q$  as the following form:

$$\chi_n(t) = \left( 1 + \sum_{j=1}^{\infty} \frac{P_j(it)}{n^{j/2}} \right) \exp(-t^2/2),$$

where  $P_j$  is a polynomial with degree  $3j$ , and then truncate the series up to a fixed number of terms. The corresponding CDF approximation is obtained by the inverse Fourier transform of the truncated series. The Edgeworth approximation of degree  $j$  amounts to truncating the above series up to terms of order  $n^{-\frac{j}{2}}$ .

Let  $\tilde{\kappa}_r(L_i)$  be the  $r$ -th cumulant of  $L_i$  under  $Q_i$ , and  $\tilde{\kappa}_r = \sum_{i=1}^n \tilde{\kappa}_r(L_i)$ . Let  $\sigma_n = \sqrt{\sum_{i=1}^n \sigma_i^2}$ . Denoted by  $\tilde{F}_{\text{EW}}(h)$ <sup>2</sup>, the degree-2 Edgeworth approximation of  $\tilde{F}_n(h)$  is given by

$$\begin{aligned} \tilde{F}_{\text{EW}}(h) = & \underbrace{\Phi(h) - \sigma_n^{-3} \cdot \frac{1}{6} \tilde{\kappa}_3 (h^2 - 1) \phi(h)}_{T_1} \\ & - \underbrace{\left( \frac{\sigma_n^{-4}}{24} \tilde{\kappa}_4 (h^3 - 3h) + \frac{\sigma_n^{-6}}{72} \tilde{\kappa}_3^2 (h^5 - 10h^3 + 15h) \right)}_{T_2} \phi(h). \end{aligned} \quad (3.7)$$

The term  $T_1$  is of order  $n^{-\frac{1}{2}}$  and  $T_2$  is of order  $n^{-1}$ . See Appendix A for detailed derivations. Our framework covers the CLT approximation introduced in [DRS19]. The CLT approximation is equivalent to the degree-0 Edgeworth approximation, whose approximation error of order  $n^{-\frac{1}{2}}$ .

**Approximate  $F_n^{-1}$ .** In Equation 3.5 we need to compute the  $F_n^{-1}(1 - \alpha)$ . This is the  $1 - \alpha$  quantile of the distribution of  $\frac{T_n(\mathbf{x}) - \mathbb{E}_P[T_n(\mathbf{x})]}{\sqrt{\text{Var}_P[T_n(\mathbf{x})]}}$ , where  $\mathbf{x}$  is  $P$  distributed. We consider two approaches to deal with it.

*Method (i):* First compute the degree-2 Edgeworth approximation  $F_{\text{EW}}(\cdot)$  of  $F_n(\cdot)$ :

$$\begin{aligned} F_{\text{EW}}(h) = & \Phi(h) - \sigma_n^{-3} \cdot \frac{1}{6} \kappa_3 (h^2 - 1) \phi(h) \\ & - \sigma_n^{-4} \cdot \frac{1}{24} \kappa_4 (h^3 - 3h) \phi(h) \\ & - \sigma_n^{-6} \cdot \frac{1}{72} \kappa_3^2 (h^5 - 10h^3 + 15h) \phi(h), \end{aligned} \quad (3.8)$$

where  $\kappa_r = \sum_{i=1}^n \kappa_r(L_i)$  and  $\kappa_r(L_i)$  is the  $r$ -th cumulant of  $L_i$  under  $P_i$ . Next, numerically solve equation  $F_{\text{EW}}(h) - 1 - \alpha = 0$ .

---

<sup>2</sup>We note that  $\tilde{F}_{\text{EW}}(h)$  is not guaranteed to be a valid CDF in general, however it is a numerical approximation to  $F_n(h)$  with improved error bounds compared to the CLT approximation.



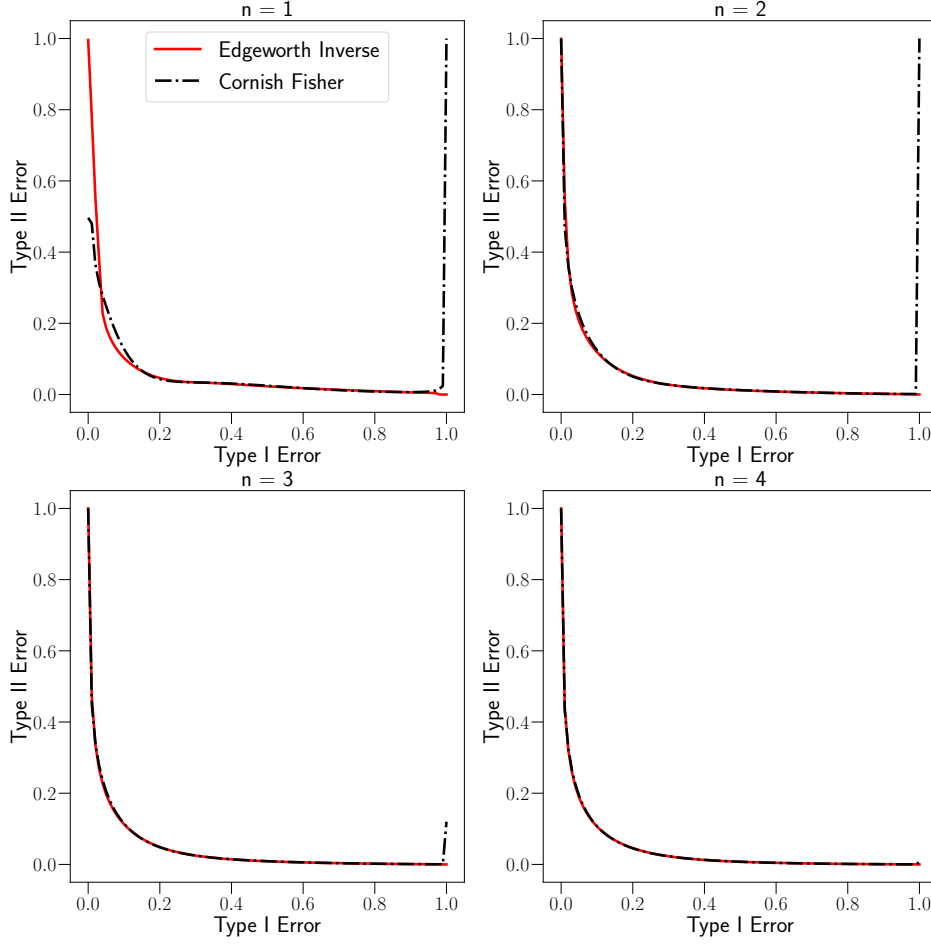


Figure 3.1: Comparison of  $f(\alpha)$  obtained when approximating  $F_n^{-1}(\cdot)$  using Method (i) and (ii). The approximation of  $\tilde{F}_n(\cdot)$  is fixed as in Equation 3.7.

*Method (ii):* Apply the closely related *Cornish-Fisher Expansion* [CF38, FC60], an asymptotic expansion used to approximate the quantiles of a probability distribution based on its cumulants, to approximate  $F_n^{-1}(\cdot)$  directly. Let  $z = \Phi^{-1}(1 - \alpha)$  be the  $1 - \alpha$  quantile of the standard normal distribution. The degree-2 Cornish-Fisher approximation of the  $1 - \alpha$  quantile of  $\frac{T_n - \mathbb{E}_P[T_n]}{\sqrt{\text{Var}_P(T_n)}}$  is given by

$$\begin{aligned}
 F_n^{-1}(1 - \alpha) &\approx z + \sigma_n^{-3} \frac{1}{6} \kappa_3 (z^2 - 1) \\
 &\quad + \sigma_n^{-4} \frac{1}{24} \kappa_4 (z^3 - 3z) - \sigma_n^{-6} \frac{1}{36} \kappa_3^2 (2z^3 - 5z).
 \end{aligned}
 \tag{3.9}$$

Both approaches have pros and cons. The Cornish-Fisher approximation has closed form solution, yet Figure 3.1 shows that it is unstable at the boundary when the number of compositions is small. For our experiments in Section 5, we use the numerical inverse approach throughout all the runs.

### 3.2 Error Analysis

Here we provide an error bound for approximating the overall privacy level  $f$  using Edgeworth expansion. For simplicity, we assume that  $f$  is symmetric and the log-likelihood ratios  $L_i$ 's are iid distributed with the common distribution having sufficiently light tails for the convergence of the Edgeworth expansion, under both  $P$  and  $Q$ .

The Edgeworth expansion of degree 2 satisfies both  $F_n(h) - F_{\text{EW}}(h) = O(n^{-\frac{3}{2}})$  and  $\tilde{F}_n(h) - \tilde{F}_{\text{EW}}(h) = O(n^{-\frac{3}{2}})$ . Conversely, the inverse satisfies  $F_n^{-1}(\alpha) - F_{\text{EW}}^{-1}(\alpha) = O(n^{-\frac{3}{2}})$  and  $\tilde{F}_n^{-1}(\alpha) - \tilde{F}_{\text{EW}}^{-1}(\alpha) = O(n^{-\frac{3}{2}})$  for  $\alpha$  that is bounded away from 0 and 1. Making use of these approximation bounds, we get

$$\begin{aligned} f_{\text{EW}}(\alpha) &= \tilde{F}_{\text{EW}}(F_{\text{EW}}^{-1}(1 - \alpha) - \mu) \\ &= \tilde{F}_{\text{EW}}(F_n^{-1}(1 - \alpha) + O(n^{-\frac{3}{2}}) - \mu) \\ &= \tilde{F}_n(F_n^{-1}(1 - \alpha) + O(n^{-\frac{3}{2}}) - \mu) + O(n^{-\frac{3}{2}}) \\ &= f(\alpha) + O(n^{-\frac{3}{2}}). \end{aligned}$$

As a caveat, we remark that the analysis above does not extend the error bound  $O(n^{-\frac{3}{2}})$  to a type I error that is close to 0 or 1. The result states that the approximation error of using the Edgeworth expansion quickly tends to 0 at the rate of  $O(n^{-\frac{3}{2}})$ . This error rate can be improved at the expense of a higher order expansion of the Edgeworth series. For comparison, our analysis can be carried over to show that the approximation error of using CLT is  $O(n^{-\frac{1}{2}})$ .

### 3.3 Computational Cost

The computational cost of the Edgeworth approximation can be broken down as follows. We first need to compute the cumulants of  $L_i$  under  $P_i$  and  $Q_i$  up to a certain order, for  $i = 1, \dots, n$ . Next, we need to compute  $F_{\text{EW}}^{-1}(1 - \alpha)$ . The Cornish-Fisher approximation (Equation 3.9) costs constant time. If we choose to compute the inverse numerically, we need to evaluate Equation 3.8 then solve a one-dimensional root-finding problem. The former has a constant cost, and the latter can be extremely efficiently computed by various types of iterative methods, for which we can consider the cost as constant too. Finally, it costs constant time to evaluate  $\tilde{F}_{\text{EW}}(\cdot)$  using Equation 3.7. Therefore, the only cost that might scale with the number of compositions  $n$  is the computation of cumulants. However, in many applications including computing the privacy bound for noisy SGD, all the  $L_i$ s are iid distributed. Under such condition, we only need to compute the cumulants once. The total cost is thus a constant independent of  $n$ . This is verified by the runtime comparison in Section 5.

## 4 A Two-Parameter Privacy Interpreter

Let  $M_1$  and  $M_2$  be two private algorithms that are associated with trade-off functions  $f_1$  and  $f_2$ , respectively. The algorithm  $M_2$  will be more private than  $M_1$  if  $f_2$  upper bounds  $f_1$ . For the family of Gaussian differentially private algorithms, this property can be reflected by the parameter  $\mu \geq 0$  directly, where a smaller value of  $\mu$  manifests a more private algorithm.

Here we provide a two-parameter description  $(\mu^*, \gamma)$  for the Edgeworth approximation, through which the privacy guarantee between two different approximations can be directly compared.

Given an approximation  $f_{\text{EW}}$ , let  $\alpha^*$  be its fixed point such that  $f_{\text{EW}}(\alpha^*) = \alpha^*$ . Let  $\mu^*$  be the parameter of GDP for which  $G_{\mu^*}$  admits the same fixed point as  $f_{\text{EW}}$ :  $G_{\mu^*}(\alpha^*) = \alpha^*$ . Such  $\mu^*$  can

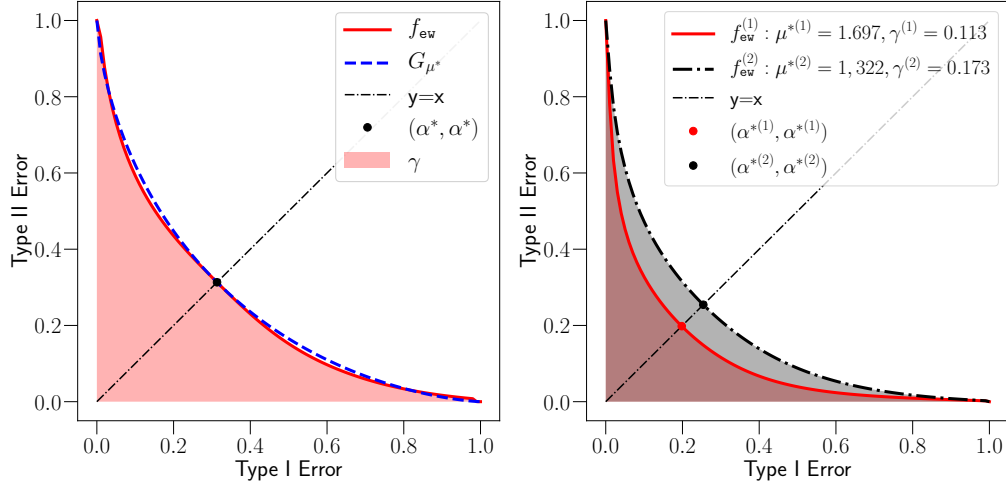


Figure 4.1: Left: An illustration of the  $(\mu^*, \gamma)$  parameterization. Right:  $f_{\text{EW}}^{(2)}$  is more private than  $f_{\text{EW}}^{(1)}$ .

be computed in closed form:

$$\mu^* = \Phi^{-1}(1 - \alpha^*) - \Phi^{-1}(\alpha^*).$$

Let  $\gamma = \int_0^1 f_{\text{EW}}(\alpha) d\alpha$  be the area under the curve of  $f_{\text{EW}}$ .

Two symmetric Edgeworth approximations<sup>3</sup>  $f_{\text{EW}}^{(1)}$  and  $f_{\text{EW}}^{(2)}$  can be compared in the sense that  $f_{\text{EW}}^{(2)}$  is more private than  $f_{\text{EW}}^{(1)}$  if their associated parameters  $(\mu^{*(1)}, \gamma^{(1)})$  and  $(\mu^{*(2)}, \gamma^{(2)})$  satisfy the following condition:

$$\mu^{*(2)} < \mu^{*(1)} \text{ and } \gamma^{(2)} > \gamma^{(1)}.$$

The left panel of Figure 4.1 provides the geometric interpretation of the above parameterization. The Edgeworth approximation  $f_{\text{EW}}$ , the CLT approximation  $G_{\mu^*}$ , and the line  $y = x$  intersect at the point  $(\alpha^*, \alpha^*)$ .

The right panel compares two Edgeworth approximations  $f_{\text{EW}}^{(1)}$  and  $f_{\text{EW}}^{(2)}$ . It is easy to see that in this case  $f_{\text{EW}}^{(2)}$  upper bounds  $f_{\text{EW}}^{(1)}$  and thus it is more private than  $f_{\text{EW}}^{(1)}$ . There are two important properties. First, its intersection with the line  $y = x$  is further away from the original point than  $f_{\text{EW}}^{(1)}$ . Consider the geometric interpretation shown in the left panel. This implies that  $\mu^{*(2)} < \mu^{*(1)}$ . Second, the approximation  $f_{\text{EW}}^{(2)}$  also has a larger area under the curve than  $f_{\text{EW}}^{(1)}$ , which is essentially  $\gamma^{(2)} > \gamma^{(1)}$ .

This parameterization defines a partial order over the set  $\{(\mu, \gamma) : \mu \geq 0, 0 \leq \gamma \leq \frac{1}{2}\}$ <sup>4</sup>. It is also applicable to general trade-off functions.

## 5 Experiments

In this section, we present numerical experiments to compare the Edgeworth approximation and the CLT approximation. Before we proceed, we introduce a numerical method to directly compute

<sup>3</sup>If  $f$  is asymmetric, we can always symmetrize it by taking  $\min\{f, f^{-1}\}^{**}$ .

<sup>4</sup>The perfect privacy is attained when the trade-off function is  $\text{Id}(\alpha) = 1 - \alpha$ , whose area under the curve is  $\frac{1}{2}$ .

the true composition bound in Section 5.1. This method is not scalable and hence merely serves as a reference for our comparison. We use the Edgeworth approximation of degree 2 for all the experiments. In the sequel, we refer to those methods as **Edgeworth**, **CLT**, and **Numerical**, respectively. All the methods are implemented in Python<sup>5</sup> and all the experiments are carried out on a MacBook with 2.5GHz processor and 16GB memory.

## 5.1 A Numerical Method

Consider the problem of computing  $f^{\otimes n}$  numerically. We know that we can find  $P, Q$  such that  $f = T(P, Q)$  and  $f^{\otimes n} = T(P^n, Q^n)$ . However, computing  $T(P^n, Q^n)$  directly involves high-dimensional testing, which can be challenging. We show this difficulty can be avoided by going from the primal representation to the dual representation. Let  $(\varepsilon, \delta_k(\varepsilon))$  be the dual representation associated with  $f^{\otimes k}$ . The method contains three steps to obtain  $\delta_n(\varepsilon)$  for  $f^{\otimes n}$ .

1. Convert  $f$  to  $\delta_1(\varepsilon)$ . This step can be done implicitly via  $P$  and  $Q$ , see Lemma 5.1.
2. Iteratively compute  $\delta_{k+1}(\varepsilon)$  from  $\delta_k(\varepsilon)$  using Lemma 5.2.
3. Convert  $\delta_n(\varepsilon)$  to  $f^{\otimes n}$  using Proposition 2.4.

Next, we explain how to compute  $\delta_{k+1}(\varepsilon)$  from  $\delta_k(\varepsilon)$ . First, we need a lemma that relates  $\delta_1(\varepsilon)$  with  $P, Q$ .

**Lemma 5.1.** *Suppose  $f = T(P, Q)$  and  $P, Q$  have densities  $p, q$  with respect to a dominating measure  $\mu$ . Then the dual representation  $\delta_1(\varepsilon) = 1 + f^*(-e^\varepsilon)$  satisfies  $\delta_1(\varepsilon) = \int (q - e^\varepsilon p)_+ d\mu$ .*

Suppose we are given  $f_1 = T(P_1, Q_1), f_2 = T(P_2, Q_2)$ . Let  $\delta_1, \delta_2$  and  $\delta_\otimes$  be the dual representations of  $f_1, f_2$  and  $f_1 \otimes f_2$  respectively. The following lemma shows how to evaluate  $\delta_\otimes$  from  $\delta_1$  and  $\delta_2$ . To simplify notations, we assume  $P_i, Q_i$  are distributions on the real line and have corresponding densities  $p_i, q_i$  for  $i = 1, 2$  with respect to Lebesgue measure. Generalization to abstract measurable space is straightforward.

**Lemma 5.2.** *Let  $L_2(y) = \log \frac{q_2(y)}{p_2(y)}$ . Then  $\delta_\otimes(\varepsilon) = \int_{\mathbb{R}} \delta_1(\varepsilon - L_2(y)) q_2(y) dy$ .*

In particular, it yields a recursive formula to compute  $f^{\otimes n}$  when  $f = T(P, Q)$ . Again we assume  $P, Q$  has densities  $p, q$  on the real line. Let  $L(x) = \log \frac{q(x)}{p(x)}$ . We have

$$\begin{aligned} \delta_0(\varepsilon) &= \max\{1 - e^\varepsilon, 0\} \\ \delta_{k+1}(\varepsilon) &= \int \delta_k(\varepsilon - L(x)) dx \\ f^{\otimes n}(\alpha) &= \sup_{\varepsilon \in \mathbb{R}} 1 - \delta_n(\varepsilon) - e^\varepsilon \alpha. \end{aligned}$$

We remark here that if  $f$  is asymmetric, then the dual involves negative  $\varepsilon$ , which is why the conversion to  $f^{\otimes n}$  involves the whole real line. The proof of the above lemmas is deferred to Appendix D.

<sup>5</sup>The code is available at <https://github.com/enosair/gdp-edgeworth>.

In practice, it is more efficient to store  $\delta_n$  in the memory than to perform the computation on the fly, so we need to discretize the domain and store the function value on this grid. Consider an abstract grid  $\{\varepsilon_j\}_{j=1}^N \subseteq \mathbb{R}$ , the recursion goes as follows:

$$\begin{aligned}\delta_0(\varepsilon_j) &= \max\{1 - e^{\varepsilon_j}, 0\}, j = 1, \dots, N \\ \delta_{k+1}(\varepsilon_j) &= \int \delta_k(\lfloor \varepsilon_j - L(x) \rfloor) dx, j = 1, \dots, N \\ f^{\otimes n}(\alpha) &= \sup_{1 \leq j \leq N} 1 - \delta_n(\varepsilon_j) - e^{\varepsilon_j} \alpha.\end{aligned}$$

where  $\lfloor \varepsilon_j - L(x) \rfloor = \max\{\varepsilon_l : \varepsilon_l \leq \varepsilon_j - L(x)\}$ . This rounding step can be replaced by an interpolation as well.

The major challenge in making this numerical method practical for computing composition product of trade-off functions is that it is slow in computation as it involves  $nN$  numerical integrations.

## 5.2 A Moderate Number of Compositions

Section 3 shows that the approximation error of **Edgeworth** is  $O(n^{-\frac{3}{2}})$ , and for **CLT** the error is  $O(n^{-\frac{1}{2}})$ . We thus expect for small or moderate values of  $n$ , **Edgeworth** will produce non-negligible outperformance to **CLT**. To verify this, we investigate their performance on a toy problem for testing order- $n$  compositions of Laplace distributions<sup>6</sup>:  $P = \text{Lap}(0, 1)^{\otimes n}$  vs  $Q = \text{Lap}(\theta, 1)^{\otimes n}$ .

We let the number of compositions  $n$  vary from 1 to 10. Since the privacy guarantee decays as  $n$  increases and the resulting curves would be very close to the axes, we set  $\theta = 3/\sqrt{n}$  for the sake of better visibility. Figure 5.1 plots the estimated trade-off functions for four representative cases  $n = 1, 3, 5, 10$ . For each of the methods, we also compute the associated  $(\varepsilon, \delta)$ -DP (see Proposition 2.3) and plot  $\delta$  as a function of  $\varepsilon$  in Figure 5.2. From both the primal and dual views, **Edgeworth** coincides better with **Numerical** in all the cases. When the number of compositions is 10, even though the difference between **Edgeworth** and **CLT** is small in the primal visualization (Figure 5.1), the  $(\varepsilon, \delta)$  presentation still clearly distinguishes them. In addition, due to the heavy tail of the Laplace distribution, we shall have  $T(\{\text{Lap}(0, 1)\}^{\otimes n}, \{\text{Lap}(\theta, 1)\}^{\otimes n}) \geq f_{\varepsilon, 0}$  for  $\varepsilon \geq n\theta$  (see Definition 2.4 for the exact form of  $f_{\varepsilon, \delta}$ ). Therefore, the ground truth has the property that  $\delta(\varepsilon) = 0$  for  $\varepsilon \geq n\theta$ . Figure 5.2 shows that **Edgeworth** also outperforms **CLT** for predicting this changing point.

	$n = 2$	$n = 4$	$n = 6$	$n = 8$	$n = 10$
CLT	0.0004	0.0004	0.0004	0.0004	0.0005
Edgeworth	0.2347	0.2341	0.2391	0.2222	0.2234
Numerical	3.6834	7.3361	12.055	16.729	21.3575

Table 1: Runtime for estimating the trade-off function for testing  $\text{Lap}(0, 1)^{\otimes n}$  vs  $\text{Lap}(3/\sqrt{n}, 1)^{\otimes n}$ .

Table 1 reports the runtime of the above experiment. **CLT** takes constant computing time at the scale of 1e-4 second. Due to the homogeneity of the component distributions under composition, the runtime of **Edgeworth** is also invariant of the composition number, which is at the scale of 0.1 second. **Numerical** is computationally heavy. Its runtime is much larger and grows linearly as the number of compositions.

<sup>6</sup> The density function of  $\text{Lap}(\theta, b)$  is  $\frac{1}{2b} \exp(-|x - \theta|/b)$ .

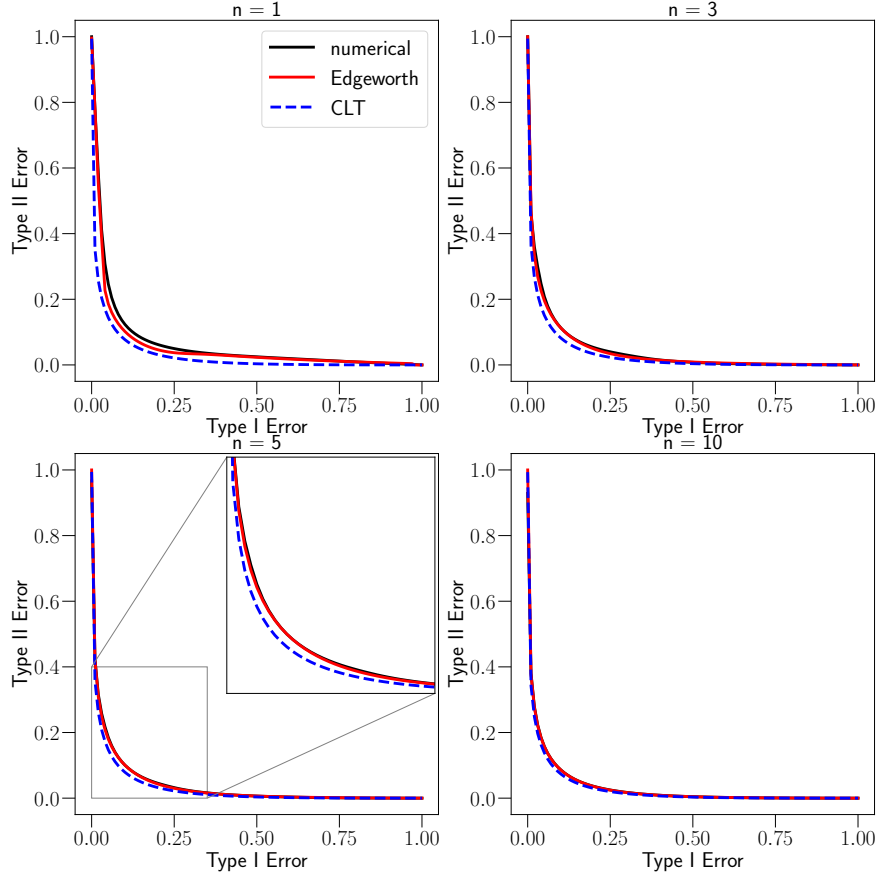


Figure 5.1: The estimated trade-off functions for testing  $\text{Lap}(0, 1)^{\otimes n}$  vs  $\text{Lap}(3/\sqrt{n}, 1)^{\otimes n}$ .

### 5.3 Privacy Guarantees for Noisy SGD

We inspect the performance of Edgeworth and CLT on estimating the privacy guarantee for  $n$ -step noisy SGD. As introduced in Section 2, the privacy bound is of form  $\min\{g, g^{-1}\}^{**}$  where  $g = (pG_{1/\sigma} + (1-p)\text{Id})^{\otimes n}$ , and the CLT estimation is  $G_\mu$  with  $\mu = p\sqrt{n}(e^{1/\sigma^2} - 1)$ . For Edgeworth, note that  $pG_{1/\sigma} + (1-p)\text{Id}$  is the trade-off function of testing the standard normal distribution  $\mathcal{N}(0, 1)$  versus the mixture model  $p\mathcal{N}(\frac{1}{\sigma}, 1) + (1-p)\mathcal{N}(0, 1)$  (see Appendix C for the proof). It follows that  $g = T(\mathcal{N}(0, 1)^{\otimes n}, \{p\mathcal{N}(1/\sigma, 1) + (1-p)\mathcal{N}(0, 1)\}^{\otimes n})$ . As a result, Edgeworth can be applied by exploiting the cumulants of  $\mathcal{N}(0, 1)$  and  $p\mathcal{N}(1/\sigma, 1) + (1-p)\mathcal{N}(0, 1)$ .

	$n = 1$	$n = 50$	$n = 100$	$n = 200$	$n = 500$
CLT	0.0004	0.0004	0.0003	0.0003	0.0003
Edgeworth	0.2504	0.2198	0.2087	0.2257	0.3048
Numerical	35.7	2138.9	4980.7	7784.8	14212.1

Table 2: Runtime for computing  $0.5/n^{\frac{1}{4}}(G_1 + \text{Id})^{\otimes n}$ .

Here we present the result when  $p = 0.5/n^{\frac{1}{4}}$ . Since the convergence of CLT requires the

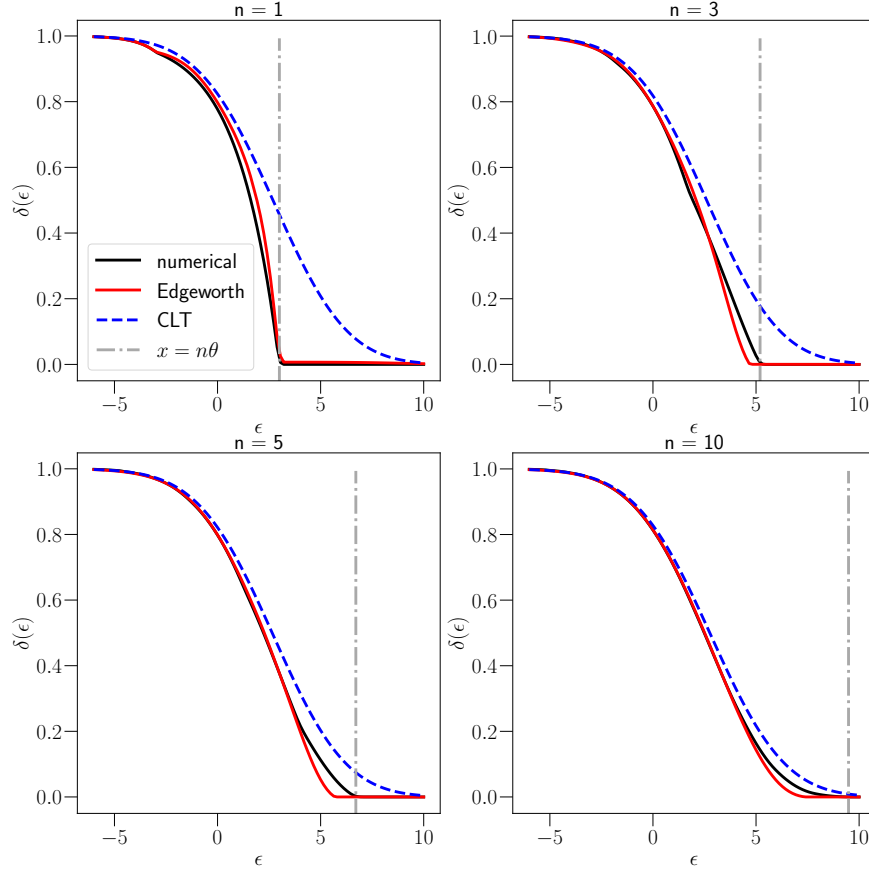


Figure 5.2: The associated  $(\varepsilon, \delta(\varepsilon))$ -DP of the estimated trade-off functions for testing  $\text{Lap}(0, 1)^{\otimes n}$  vs  $\text{Lap}(3/\sqrt{n}, 1)^{\otimes n}$ .

assumption  $p\sqrt{n} \rightarrow \nu > 0$  [BDLS19], this is the regime that the performance of CLT does not have theoretical guarantees, and we shall see in the following experiment that CLT significantly deviates **Numerical**. Nevertheless, even in this case, **Edgeworth** enjoys a nearly perfect match with **Numerical**. We also investigated the case  $p = 0.5/n^{-\frac{1}{2}}$ , where the convergence of CLT is guaranteed. In this regime, we shall see that **Edgeworth** still outperforms CLT. The results are deferred to Appendix E.

We let  $n$  vary from 1 to 500. The noise scale  $\sigma$  is set to 1. Figure 5.3 presents the estimated  $g$  for two representative cases  $n = 5$  and  $n = 500$ . Figure 5.4 shows the final estimation after symmetrization, for a few selected values of  $n$ . Obviously, **Edgeworth** is very close to **Numerical** for all the cases. Unlike the previous experiment, where CLT performs reasonably well when  $n$  reaches 10, here the difference between CLT and the other two remains large even when  $n = 500$ .

Table 2 presents the runtime comparison. Clearly, **Numerical** is not scalable. It takes more than two hours to compute 500 compositions, whereas CLT and **Edgeworth** can be computed within one second. The runtime of **Numerical** has grown three orders of magnitude from  $n = 1$  to  $n = 500$ , yet the runtime of CLT and **Edgeworth** remains constant for all the runs.



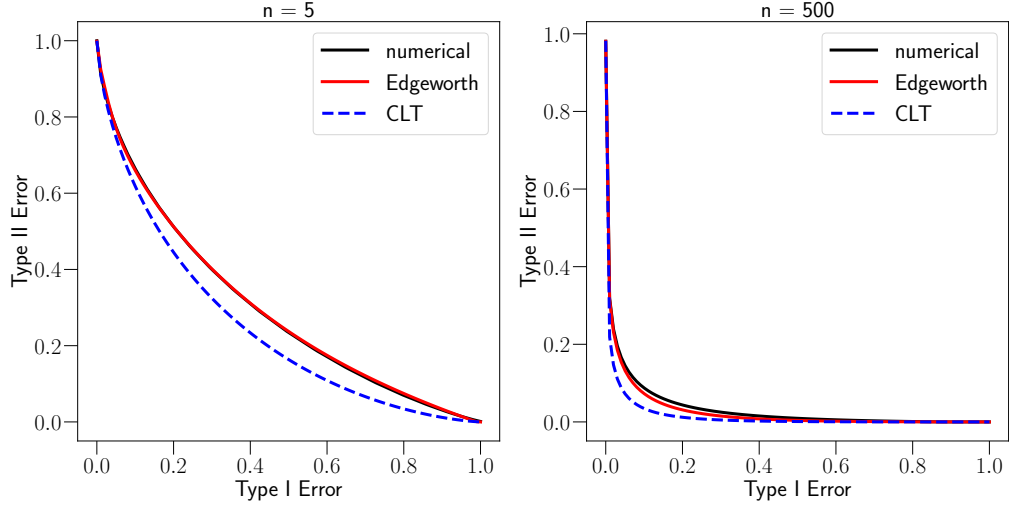


Figure 5.3: The estimation of  $0.5/n^{\frac{1}{4}}(G_1 + \text{Id})^{\otimes n}$ .

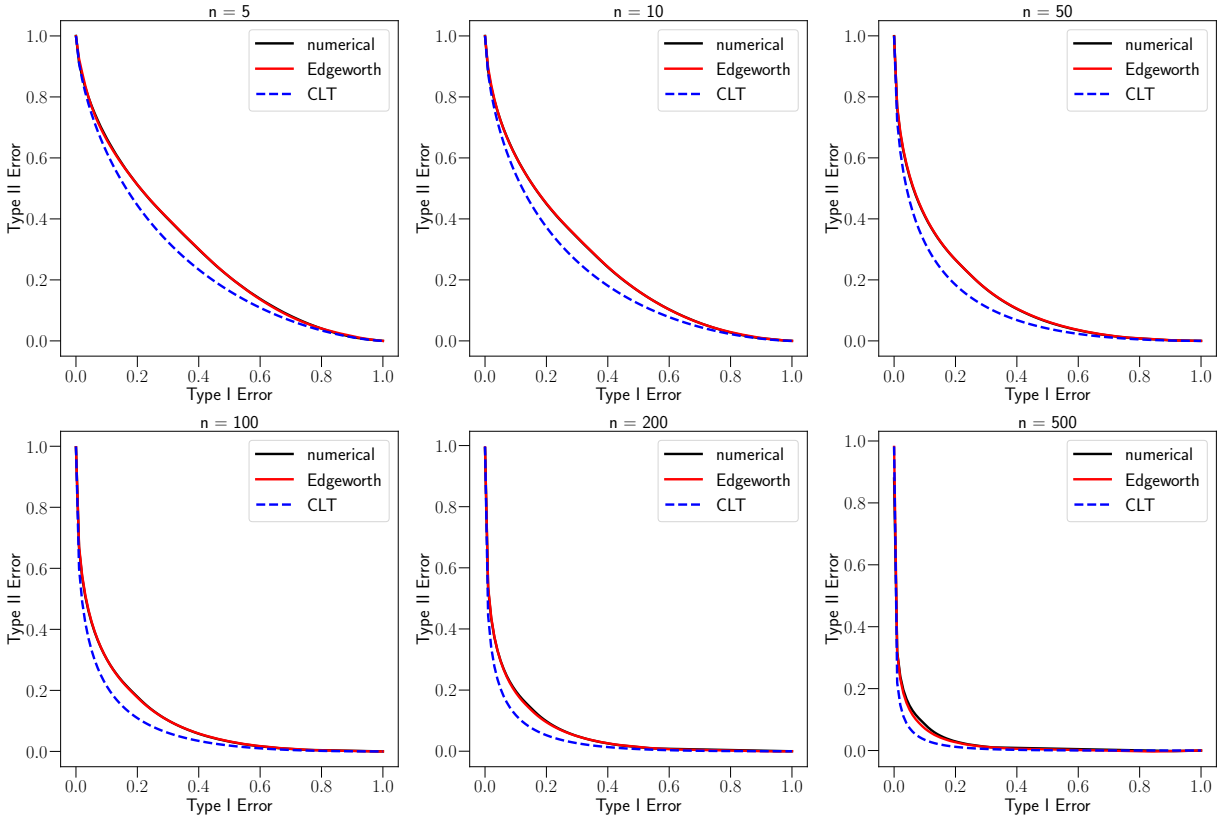


Figure 5.4: The estimation of the privacy bound for  $n$ -step noisy SGD. The sampling rate is  $p = 0.5/n^{\frac{1}{4}}$  and the noise scale is  $\sigma = 1$ .

## Acknowledgments

We are grateful to Yicong Jiang for stimulating discussions in the early stages of this work. This work was supported in part by NSF through CAREER DMS-1847415, CCF-1763314, and CCF-1934876,

the Wharton Dean’s Research Fund, and NIH through R01GM124111.

## References

- [Abo18] John M Abowd. The us census bureau adopts differential privacy. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2867–2867, 2018.
- [ACG<sup>+</sup>16] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318, 2016.
- [App17] D Apple. Learning with privacy at scale. *Apple Machine Learning Journal*, 1(8), 2017.
- [BBG<sup>+</sup>19] Borja Balle, Gilles Barthe, Marco Gaboardi, Justin Hsu, and Tetsuya Sato. Hypothesis testing interpretations and renyi differential privacy. *arXiv preprint arXiv:1905.09982*, 2019.
- [BDLS19] Zhiqi Bu, Jinshuo Dong, Qi Long, and Weijie J Su. Deep learning with Gaussian differential privacy. *arXiv preprint arXiv:1911.11607*, 2019.
- [BDRS18] Mark Bun, Cynthia Dwork, Guy N Rothblum, and Thomas Steinke. Composable and versatile privacy via truncated cdp. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 74–86, 2018.
- [BS16] Mark Bun and Thomas Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography Conference*, pages 635–658. Springer, 2016.
- [CF38] Edmund A Cornish and Ronald A Fisher. Moments and cumulants in the specification of distributions. *Revue de l’Institut international de Statistique*, pages 307–320, 1938.
- [DKM<sup>+</sup>06] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Proceedings of the 24th Annual International Conference on The Theory and Applications of Cryptographic Techniques*, EUROCRYPT’06, pages 486–503, Berlin, Heidelberg, 2006. Springer-Verlag.
- [DKY17] Bolin Ding, Janardhan Kulkarni, and Sergey Yekhanin. Collecting telemetry data privately. In *Advances in Neural Information Processing Systems*, pages 3571–3580, 2017.
- [DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Third Conference on Theory of Cryptography*, TCC’06, pages 265–284, Berlin, Heidelberg, 2006. Springer-Verlag.
- [DR16] Cynthia Dwork and Guy N Rothblum. Concentrated differential privacy. *arXiv preprint arXiv:1603.01887*, 2016.

- [DRS19] Jinshuo Dong, Aaron Roth, and Weijie J Su. Gaussian differential privacy. *arXiv preprint arXiv:1905.02383*, 2019.
- [EPK14] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 1054–1067, 2014.
- [FC60] Sir Ronald A Fisher and EA Cornish. The percentile points of distributions having known cumulants. *Technometrics*, 2(2):209–225, 1960.
- [FJR15] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1322–1333, 2015.
- [Hal13] Peter Hall. *The bootstrap and Edgeworth expansion*. Springer Science & Business Media, 2013.
- [KOV17] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. The composition theorem for differential privacy. *IEEE Transactions on Information Theory*, 63(6):4037–4049, 2017.
- [KST12] Daniel Kifer, Adam Smith, and Abhradeep Thakurta. Private convex empirical risk minimization and high-dimensional regression. In *Conference on Learning Theory*, pages 25–1, 2012.
- [LHC<sup>+</sup>19] Changchang Liu, Xi He, Thee Chanyaswad, Shiqiang Wang, and Prateek Mittal. Investigating statistical privacy frameworks from the perspective of hypothesis testing. *Proceedings on Privacy Enhancing Technologies*, 2019(3):233–254, 2019.
- [LO11] Yehuda Lindell and Eran Omri. A practical application of differential privacy to personalized online advertising. *IACR Cryptology EPrint Archive*, 2011:152, 2011.
- [Mir17] Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE, 2017.
- [MV16] Jack Murtagh and Salil Vadhan. The complexity of computing the optimal composition of differential privacy. In *Theory of Cryptography Conference*, pages 157–175. Springer, 2016.
- [PAE<sup>+</sup>16] Nicolas Papernot, Martín Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. In *Proceedings of the 5th International Conference on Learning Representations*, 2016.
- [SSSS17] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE, 2017.
- [WZ10] Larry Wasserman and Shuheng Zhou. A statistical framework for differential privacy. *Journal of the American Statistical Association*, 105(489):375–389, 2010.

## A The Edgeworth Approximation

We can apply Edgeworth expansion to approximate  $\tilde{F}_n$  directly, following the techniques introduced in [Hal13]. Let us assume  $x \sim Q$ . Denote

$$X_Q = \frac{T_n - \mathbb{E}_Q[T_n]}{\sqrt{\text{Var}_Q(T_n)}} = \frac{\sum_{i=1}^n (L_i - \mu_i)}{\sqrt{\sum_{i=1}^n \sigma_i^2}}, \quad (\text{A.1})$$

where  $\mu_i$  and  $\sigma_i^2$  are the mean and variance of  $L_i$  under the distribution  $Q_i$ . The characteristic function of  $X_Q$  is

$$\chi_n(t) = \exp \left( \sum_{i=1}^{\infty} \tilde{\kappa}_r(X_Q) \frac{(it)^r}{r!} \right),$$

where  $\tilde{\kappa}_r(X_Q)$  is the  $r$ -th cumulant of  $X_Q$ . Details of how to compute the cumulants are summarized in Appendix B. Let  $\sigma_n = \sqrt{\sum_{i=1}^n \sigma_i^2}$ . Particularly we have

$$\begin{aligned} \tilde{\kappa}_1(X_Q) &= \mathbb{E}_Q(X_Q) = 0, \\ \tilde{\kappa}_2(X_Q) &= \text{Var}_Q(X_Q) = 1, \\ &\vdots \\ \tilde{\kappa}_r(X_Q) &= \tilde{\kappa}_r \left( \sigma_n^{-1} \sum_{i=1}^n (L_i - \mu_i) \right) \\ &= \sigma_n^{-r} \sum_{i=1}^n \tilde{\kappa}_r(L_i), \quad \forall r > 2. \end{aligned} \quad (\text{A.2})$$

We will denote the sum of  $n$  cumulants by  $\tilde{\kappa}_r = \sum_{i=1}^n \tilde{\kappa}_r(L_i)$ . Under the series expansion of the exponential function, we will have

$$\begin{aligned} \chi_n(t) &= \exp \left( -\frac{t^2}{2} \right) \exp \left( \sum_{r=3}^{\infty} \frac{\sigma_n^{-r}}{r!} \tilde{\kappa}_r (it)^r \right) \\ &\approx \exp \left( -\frac{t^2}{2} \right) \exp \left( \sum_{r=3,4} \frac{\sigma_n^{-r}}{r!} \tilde{\kappa}_r (it)^r \right) \\ &\approx \exp \left( -\frac{t^2}{2} \right) \left( 1 + \sigma_n^{-3} \cdot \overbrace{\frac{1}{6} \tilde{\kappa}_3(it)^3}^{r_1(it)} \right. \\ &\quad \left. + \sigma_n^{-4} \cdot \overbrace{\frac{1}{24} \tilde{\kappa}_4(it)^4}^{r_2(it)} + \sigma_n^{-6} \cdot \overbrace{\frac{1}{72} \tilde{\kappa}_3^2(it)^6}^{r_3(it)} \right). \end{aligned} \quad (\text{A.3})$$

Since  $\chi_n(t) = \int e^{ith} d\tilde{F}_n(h)$  and  $e^{-t^2/2} = \int e^{ith} d\Phi(h)$ , we can obtain the corresponding “inverse” expansion:

$$\tilde{F}_n(h) \approx \Phi(h) + \sigma_n^{-3} \cdot R_1(h) + \sigma_n^{-4} \cdot R_2(h) + \sigma_n^{-6} \cdot R_3(h), \quad (\text{A.4})$$

and  $R_j(h)$  is a function whose Fourier-Stieljes transform equals  $r_j(it)e^{-t^2/2}$ :

$$\int_{-\infty}^{\infty} e^{ith} dR_j(h) = r_j(it)e^{-t^2/2}.$$

Let  $D$  denote the differential operator  $d/dh$ . We have

$$e^{-t^2/2} = (-it)^{-j} \int_{-\infty}^{\infty} e^{ith} d\{D^j \Phi(h)\}$$

and hence

$$\int_{-\infty}^{\infty} e^{ith} d\{(-D)^j \Phi(h)\} = (it)^j e^{-t^2/2}.$$

Let us interpret  $r_j(-D)$  as a polynomial in  $D$ , we then obtain

$$\int_{-\infty}^{\infty} e^{ith} d\{r_j(-D)\Phi(h)\} = r_j(it)e^{-t^2/2}.$$

Consequently,

$$R_j(h) = r_j(-D)\phi(h). \tag{A.5}$$

It is well known that for  $j \geq 1$ ,

$$(-D)^j \Phi(h) = -He_{j-1}(h)\phi(h) \tag{A.6}$$

and  $He_j$ s are the Hermite polynomials:

$$\begin{aligned} He_0(h) &= 1, \\ He_1(h) &= h, \\ He_2(h) &= h^2 - 1, \\ He_3(h) &= h^3 - 3h, \\ He_4(h) &= h^4 - 6h^2 + 3, \\ He_5(h) &= h^5 - 10h^3 + 15h, \\ He_6(h) &= h^6 - 15h^4 + 45h^2 - 15, \\ He_7(h) &= h^7 - 21h^5 + 105h^3, \\ &\dots \end{aligned} \tag{A.7}$$

Combine equations [A.4](#), [A.5](#), [A.6](#) and [A.7](#) we can deduce the final result:

$$\begin{aligned} \tilde{F}_n(h) &\approx \Phi(h) + \sigma_n^{-3} \cdot -\frac{1}{6} \tilde{\kappa}_3(h^2 - 1)\phi(h) \\ &\quad + \sigma_n^{-4} \cdot -\frac{1}{24} \tilde{\kappa}_4(h^3 - 3h)\phi(h) \\ &\quad + \sigma_n^{-6} \cdot -\frac{1}{72} \tilde{\kappa}_3^2(h^5 - 10h^3 + 15h)\phi(h). \end{aligned} \tag{A.8}$$

In [A.3](#), the truncation happens in both the second and third line. In the second line, we truncated terms where  $r \geq 5$ . In the following line, we apply the series expansion to the exponential function, and we stopped after taking  $t_1 := \sigma_n^{-3} \cdot \frac{1}{6} \tilde{\kappa}_3(it)^3$ ,  $t_2 := \sigma_n^{-4} \cdot \frac{1}{24} \tilde{\kappa}_4(it)^4$  and the square of  $t_1$ .

The error stems from truncating  $r \geq 5$  terms in the second line will be dominated by  $\frac{1}{120}\sigma_n^{-5}\tilde{\kappa}_5(it)^5$  in the series expansion. The error stems from truncating the expansion of  $r = 3, 4$  terms in the following line will be dominated by the square of  $t_2$ :  $\sigma_n^{-8} \cdot \frac{1}{576}\tilde{\kappa}_4^2(it)^8$ .

Since all  $L_i$ 's are identically distributed, the cumulants of  $L_1, \dots, L_n$  take the same value for any fixed order. Therefore,  $\sigma_1 = \dots = \sigma_n = \sigma$  and  $\tilde{\kappa}_r = \tilde{\kappa}_r(L_1) = \dots = \tilde{\kappa}_r(L_n)$ . As a consequence, we have  $\sigma_n = \sqrt{n}\sigma$  and  $\tilde{\kappa}_r = n\tilde{\kappa}_r$ . This leads to

$$\begin{aligned}\sigma_n^{-3} \cdot \tilde{\kappa}_3(it)^3 &\sim n^{-1/2}(it)^3, \\ \sigma_n^{-4} \cdot \tilde{\kappa}_4(it)^4 &\sim n^{-1}(it)^4, \\ \sigma_n^{-6} \cdot \tilde{\kappa}_3^2(it)^6 &\sim n^{-1}(it)^6, \\ \sigma_n^{-8} \cdot \tilde{\kappa}_4^2(it)^8 &\sim n^{-2}(it)^8, \\ \sigma_n^{-5} \cdot \tilde{\kappa}_5(it)^5 &\sim n^{-3/2}(it)^5.\end{aligned}\tag{A.9}$$

Hence the error for approximating  $\chi_n(t)$  is upper bounded by  $O(n^{-2}(it)^8 + n^{-3/2}(it)^5)$ . Next, we connect the characteristic function to CDF  $\tilde{F}_n(h)$ . From equations A.5 and A.6, we know the error term will be transformed into  $O(n^{-2}He_7(h) + n^{-3/2}He_4(h))$  as approximating  $\tilde{F}_n(h)$ , which is  $O(n^{-2}h^7 + n^{-3/2}h^3)$ .

## B Computing Cumulants From Moments

The cumulants of a random variable  $X$  are defined using the cumulant-generating function  $K(t)$ . It is the natural logarithm of the moment-generating function:

$$K(t) = \log \mathbb{E}(e^{tX}),$$

and the cumulants are the coefficients in the Taylor expansion of  $K(t)$  about the origin:

$$K(t) = \log \mathbb{E}(e^{tX}) = \sum_{r=0}^{\infty} \kappa_r t^r / r!.$$

For any integer  $r \geq 0$ , the  $r$ -th order non-central moment of  $X$  is  $\mu_r = \mathbb{E}(X^r)$ . Recall the Taylor expansion of the moment-generating function  $M(t)$  about the origin

$$M(t) = \mathbb{E}(e^{tX}) = \sum_{r=0}^{\infty} \mu_r t^r / r! = \exp(K(t)).$$

The cumulants can be recovered in terms of the moments and vice versa. In general,

$$\kappa_r = \sum_{k=1}^r (-1)^{k-1} (k-1)! B_{r,k}(\mu_1, \dots, \mu_{r-k+1})$$

where  $B_{n,k}$  are Bell polynomials. The relationship between the first few cumulants and moments is as the following:

$$\begin{aligned}\kappa_0 &= 0, \\ \kappa_1 &= \mu_1, \\ \kappa_2 &= \mu_2 - \mu_1^2, \\ \kappa_3 &= \mu_3 - 3\mu_2\mu_1 + 2\mu_1^3, \\ \kappa_4 &= \mu_4 - 4\mu_3\mu_1 - 3\mu_2^2 + 12\mu_2\mu_1^2 - 6\mu_1^4.\end{aligned}$$

## C $\mathcal{N}(0, 1)$ vs $p\mathcal{N}(\mu, 1) + (1-p)\mathcal{N}(0, 1)$

Let  $P$  be the standard normal distribution  $\mathcal{N}(0, 1)$  and  $Q$  be a mixture model  $p\mathcal{N}(\mu, 1) + (1-p)\mathcal{N}(0, 1)$  with  $\mu \geq 0$ . We now show that

**Lemma C.1.**

$$T(P, Q) = pG_\mu + (1-p)\text{Id}.$$

*Proof.* The likelihood ratio between  $Q$  and  $P$  is

$$pe^{-\frac{1}{2}(x-\mu)^2 + \frac{1}{2}x^2} + 1 - p = pe^{\mu x - \frac{1}{2}\mu^2} + 1 - p.$$

Since  $\mu \geq 0$ , likelihood ratio tests are thresholding, i.e.,  $\{x : x > h\}$ . The type I and type II errors are

$$\begin{aligned}\alpha &= P\{x : x > h\} = 1 - \Phi(h), \\ \beta &= Q\{x : x \leq h\} \\ &= p\mathbb{E}_{x \sim \mathcal{N}(\mu, 1)}[\mathbf{1}_{\{x \leq h\}}] + (1-p)\mathbb{E}_{x \sim \mathcal{N}(0, 1)}[\mathbf{1}_{\{x \leq h\}}] \\ &= p\Phi(h - \mu) + (1-p)\Phi(h).\end{aligned}$$

Inverting the first formula, we have  $h = \Phi^{-1}(1 - \alpha)$ . So

$$\beta = p\Phi(h - \mu) + (1-p)\Phi(h) = p\Phi(\Phi^{-1}(1 - \alpha) - \mu) + (1-p)(1 - \alpha)$$

Making use of the known expression  $G_\mu(\alpha) = \Phi(\Phi^{-1}(1 - \alpha) - \mu)$  and  $\text{Id}(\alpha) = 1 - \alpha$ , we have

$$T(P, Q)(\alpha) = \beta = pG_\mu(\alpha) + (1-p)\text{Id}(\alpha).$$

□

## D Details of the Numerical Method

### D.1 Proof of Lemma 5.1

*Proof.* By definition of convex conjugacy,  $\delta \geq \delta_1(\varepsilon)$  if and only if  $f(x) \geq 1 - \delta - e^\varepsilon x$  for all  $x \in [0, 1]$ . Since  $f = T(P, Q)$  characterizes optimal testing rules,  $f(x) \geq 1 - \delta - e^\varepsilon x$  for any  $x \in [0, 1]$  if and only if for any event  $E$ ,  $Q[E] \leq e^\varepsilon P[E] + \delta$ . That is,

$$\delta_1(\varepsilon) = \min\{\delta : Q[E] \leq e^\varepsilon P[E] + \delta, \forall E\}$$



$$\begin{aligned}
&= \max_E Q[E] - e^\varepsilon P[E] \\
&= \max_E \int_E [q(x) - e^\varepsilon p(x)] d\mu(x).
\end{aligned}$$

Obviously, the maximum is attained at the event that the integrand being non-negative. That is,  $E = \{x : q(x) - e^\varepsilon p(x) \geq 0\}$ . Therefore,

$$\delta_1(\varepsilon) = \int (q - e^\varepsilon p)_+ d\mu.$$

□

## D.2 Proof of Lemma 5.2

*Proof.* By definition of  $\otimes$  and Lemma 5.1, we have

$$\begin{aligned}
\delta_{\otimes}(\varepsilon) &= 1 + (f_1 \otimes f_2)^*(-e^\varepsilon) \\
&= 1 + (T(P_1 \times P_2, Q_1 \times Q_2))^*(-e^\varepsilon) && \text{(Def of } \otimes \text{)} \\
&= \iint (q_1(x)q_2(y) - e^\varepsilon p_1(x)p_2(y))_+ dx dy && \text{(Lemma 5.1)} \\
&= \iint q_2(y) \cdot \left( q_1(x) - e^\varepsilon p_1(x) \cdot \frac{p_2(y)}{q_2(y)} \right)_+ dx dy && (q_2(y) \geq 0) \\
&= \iint q_2(y) \cdot \left( q_1(x) - e^{\varepsilon - L_2(y)} p_1(x) \right)_+ dx dy && \text{(Def of } L_2 \text{)} \\
&= \int q_2(y) \cdot \left[ \int (q_1(x) - e^{\varepsilon - L_2(y)} p_1(x))_+ dx \right] dy && \text{(Fubini)} \\
&= \int q_2(y) \cdot \delta_1(\varepsilon - L_2(y)) dy. && \text{(Lemma 5.1 on } \delta_1 \text{)}
\end{aligned}$$

□

## E Privacy Guarantees for Noisy SGD with Sampling Rate $p = \frac{0.5}{\sqrt{n}}$

In Section 5.3 we present the result when the sampling rate  $p = 0.5/n^{1/4}$ . Since the convergence of CLT requires the assumption  $p\sqrt{n} \rightarrow \nu > 0$  [BDLS19], that is a regime where the performance of CLT does not have theoretical guarantees. Here we present the results when  $p = 0.5/n^{1/2}$ , where the convergence of CLT is guaranteed. However, we still observe that **Edgeworth** outperforms CLT. See Figure E.1 and E.2 for the comparison.

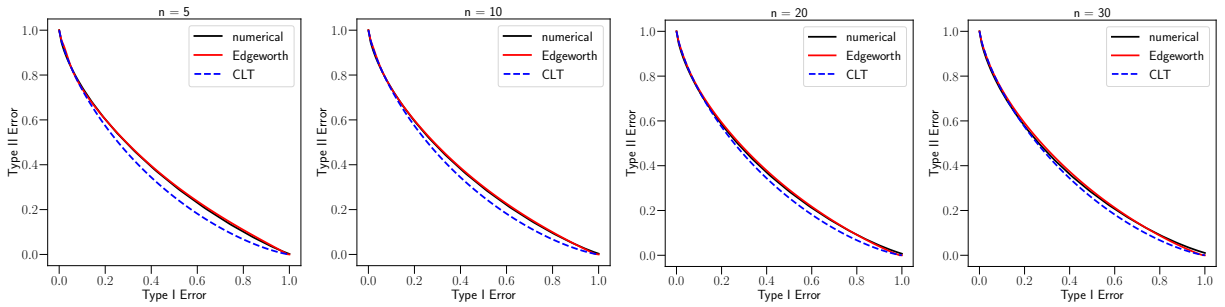


Figure E.1: The estimation of  $0.5/n^{\frac{1}{2}}(G_1 + \text{Id})^{\otimes n}$ .

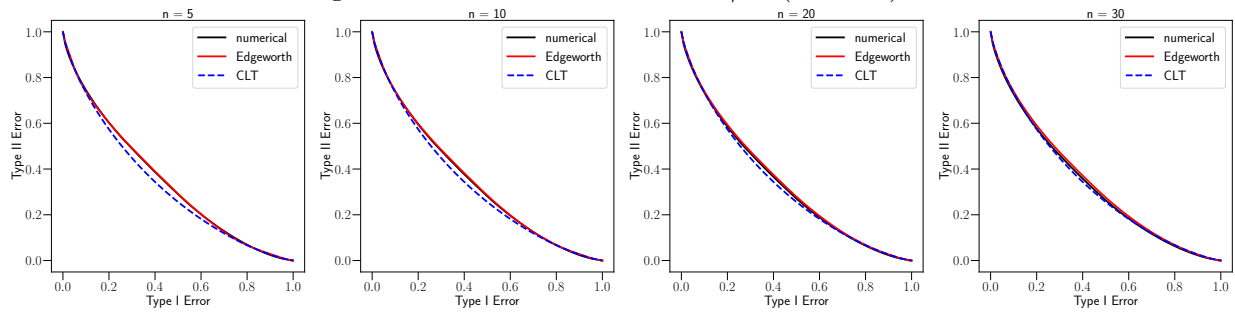


Figure E.2: The estimation of the privacy bound for  $n$ -step noisy SGD. The sampling rate is  $p = 0.5/n^{\frac{1}{2}}$  and the noise scale is  $\sigma = 1$ .