Privacy Loss of Noise Perturbation via Concentration Analysis of A Product Measure

SHUAINAN LIU, Texas Tech University, USA
TIANXI JI*, Texas Tech University, USA
ZHONGSHUO FANG, Texas Tech University, USA
LU WEI, Texas Tech University, USA
PAN LI, Hangzhou Dianzi University, China

Noise perturbation is one of the most fundamental approaches for achieving (ϵ, δ)-differential privacy (DP) guarantees when releasing the result of a query or function $f(\cdot) \in \mathbb{R}^M$ evaluated on a sensitive dataset $\boldsymbol{x}$. In this approach, calibrated noise $\mathbf{n} \in \mathbb{R}^M$ is used to obscure the difference vector $f(\boldsymbol{x}) - f(\boldsymbol{x}')$, where $\boldsymbol{x}'$ is known as a neighboring dataset. A DP guarantee is obtained by studying the tail probability bound of a privacy loss random variable (PLRV), defined as the Radon-Nikodym derivative between two distributions. When $\mathbf{n}$ follows a multivariate Gaussian distribution, the PLRV is characterized as a specific univariate Gaussian. In this paper, we propose a novel scheme to generate $\mathbf{n}$ by leveraging the fact that the perturbation noise is typically spherically symmetric (i.e., the distribution is rotationally invariant around the origin). The new noise generation scheme allows us to investigate the privacy loss from a geometric perspective and express the resulting PLRV using a product measure, $W \times U$; measure W is related to a radius random variable controlling the magnitude of $\mathbf{n}$, while measure U involves a directional random variable governing the angle between $\mathbf{n}$ and the difference $f(\boldsymbol{x}) - f(\boldsymbol{x}')$. We derive a closed-form moment bound on the product measure to prove (ϵ, δ)-DP. Under the same (ϵ, δ)-DP guarantee, our mechanism yields a smaller expected noise magnitude than the classic Gaussian noise in high dimensions, thereby significantly improving the utility of the noisy result $f(\boldsymbol{x}) + \mathbf{n}$. To validate this, we consider privacy-preserving convex and non-convex empirical risk minimization (ERM) problems in high dimensional space. We propose leveraging our developed noise in output perturbation, objective perturbation, and gradient perturbation to establish DP guarantees when solving ERMs. Experiments on multiple datasets show that our method achieves significant utility improvements for convex ERM models (e.g., regression and SVM) under the same privacy guarantees. For non-convex models (e.g., neural networks), it provides substantially stronger privacy guarantees under comparable utility.[1]

CCS Concepts: • **Theory of computation → Randomness, geometry and discrete structures**; • **Security and privacy → Data anonymization and sanitization**.

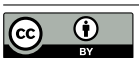
Additional Key Words and Phrases: Differential Privacy, Measure Concentration, Optimization

ACM Reference Format:
Shuainan Liu, Tianxi Ji, Zhongshuo Fang, Lu Wei, and Pan Li. 2026. Privacy Loss of Noise Perturbation via Concentration Analysis of A Product Measure. *Proc. ACM Manag. Data* 4, 1 (SIGMOD), Article 66 (February 2026), 27 pages. https://doi.org/10.1145/3786680

*Corresponding author.

[1]Code Repository: https://github.com/issleepgroup/Product-Noise-ERM

Authors' Contact Information: Shuainan Liu, Shuainan.Liu@ttu.edu, Texas Tech University, Lubbock, Texas, USA; Tianxi Ji, Texas Tech University, Lubbock, Texas, USA, tiji@ttu.edu; Zhongshuo Fang, Texas Tech University, Lubbock, Texas, USA, zhonfang@ttu.edu; Lu Wei, Texas Tech University, Lubbock, Texas, USA, luwei@ttu.edu; Pan Li, Hangzhou Dianzi University, Hangzhou, Zhejiang, China, lipan@ieee.org.



ACM 2836-6573/2026/2-ART66
https://doi.org/10.1145/3786680

1 Introduction

Differential privacy (DP) [17] is widely applied to protect the privacy of a sensitive dataset $\boldsymbol{x}$ when releasing computation results, denoted as $f(\boldsymbol{x}) \in \mathbb{R}^M$. An important privacy guarantee is (ϵ, δ)-DP. It means that, with the exception of a small probability δ (e.g., $< 10^{-5}$), the inclusion or exclusion of any single data record in $\boldsymbol{x}$ cannot change the probability of observing a specific computation result by more than a multiplicative factor of e^{ϵ}. (ϵ, δ)-DP enables a unified and comparable privacy notion for privacy-preserving algorithms, and has been widely applied to quantify privacy loss in various privacy-sensitive tasks, such as covariance matrix estimation in principal component analysis (PCA) [11, 35], recommendation systems [36, 46], data mining [21, 33], graph data publication [28, 30], machine learning and artificial intelligence [1, 10, 26].

The classic Gaussian mechanism [16] is the most common way to achieve (ϵ, δ)-DP. It perturbs the computation result $f(\boldsymbol{x}) \in \mathbb{R}^M$ using Gaussian noise $\mathbf{n} \in \mathbb{R}^M$ and releases the noisy result $\mathbf{s}$, i.e.,

$$\mathbf{s} = f(\boldsymbol{x}) + \mathbf{n}. \tag{1}$$

There are quite a few variants of this classic mechanism, e.g., [2, 8]. In general, they all establish (ϵ, δ)-DP guarantee by investigating the tail probability bound of the **privacy loss random variable** (PLRV), defined as the Radon-Nikodym derivatives (cf. Definition 2 in Section 3). In essence, the PLRV quantifies the ratio between the probability density functions (PDFs) of two randomized outputs generated from slightly different inputs.

Looking from the lens of PLRV, all existing DP mechanisms using Gaussian noise share common characteristic: when Gaussian noise with variance σ^2 is used to perturb $f(\boldsymbol{x})$ with l_2 sensitivity $\Delta_2 f$,[2] the resulted PLRV is characterized as another Gaussian distribution, $\mathcal{N}\left(\frac{(\Delta_2 f)^2}{2\sigma^2}, 2\frac{(\Delta_2 f)^2}{2\sigma^2}\right)$ (cf. [44, Proposition 3] and [2]). Then, a certain (ϵ, δ)-DP guarantee is constructed by analyzing the tail probability of such Gaussian distributed PLRV. Other DP variants that leverage Gaussian noise, such as Concentrated DP [6, 18] or Rényi DP [37] handle the PLRV in much the same way; the considered PLRV is still bounded using the measure concentration of a Gaussian random variable. For example, to compose the privacy loss in DP-stochastic gradient descent, Abadi et al. [1] use the Cramér-Chernoff method to bound the moment generation function of a Gaussian-type PLRV.

From the perspective of the noise itself, the multivariate Gaussian noise in (1) is known to be spherically symmetric [19, 39], which means that the noise can be represented as a random unit vector $\boldsymbol{h} \in \mathbb{R}^M$ scaled by a non-negative random scalar R and a privacy related constant σ, i.e., $\mathbf{n} = \sigma R \boldsymbol{h}$ (the details are deferred to Section 4.1). Since $\|\boldsymbol{h}\|_2 = 1$, we have $\|\mathbf{n}\|_2 \propto R$, thus, a straightforward approach to control the perturbation noise in (1) is to make the non-negative random scalar R as small as possible. Additionally, to control the event where a query result $f(\boldsymbol{x})$ is overwhelmed by large noise, we wish extremely large values of $\|\mathbf{n}\|_2$ (or outliers) are less likely to happen. Statistically speaking, we prefer $\|\mathbf{n}\|_2$ (or equivalently R) to have large kurtosis and skewness [7, 49].

Unfortunately, as we will rigorously show in Section 4.1, the commonly adopted multivariate Gaussian noise leads to two intrinsic limitations: (i) a large R, distributed as χ_M with degrees of freedom equal to the output dimension of $f(\boldsymbol{x}) \in \mathbb{R}^M$, and (ii) minimal kurtosis and skewness that scale inversely with the dimension, i.e., $\boldsymbol{O}(\frac{1}{M})$. To overcome these drawbacks, we introduce a new product noise construction, where the non-negative scalar R is instead drawn from χ_1.

As one might expect, there exists a trade-off in the new noise construction as the degree of freedom of R is reduced from M to 1. Specifically, this trade-off manifests in the privacy-related constant, which becomes dependent on the dimension M, and we denote it as σ_M in our noise (a counterpart of σ in the Gaussian mechanism). Nevertheless, we will develop advanced techniques

[2] $\Delta_2 f = \sup_{\boldsymbol{x} \sim \boldsymbol{x}'} ||f(\boldsymbol{x}) - f(\boldsymbol{x}')||_2$, where $\boldsymbol{x}'$ differs with $\boldsymbol{x}$ by only one data record.

to control this trade-off, ensuring the new constant σ_M grows only sublinearly with respect to M. Our novel techniques have two key components:

• We construct a **non-Gaussian type** PLRV by interpreting noise perturbation in (1) as a random triangle instance in $\mathbb{R}^M$. Under this geometric view, the new form of PLRV is a product of two measures; one (denoted as W) is related to a random variable controlling the magnitude of our noise, and the other (indicated by U) involves a directional random variable governing the angle between our noise and the sensitive vector that needs to be obfuscated.

• To control the value of the constructed PLRV (i.e., $W \times U$), we adopt the moment bound [40] to directly constrain its tail probability, which can be represented in the form of $\Pr\left[WU \geq \frac{\sigma_M}{\Delta_2 f}\epsilon\right] \leq \delta$. Moment bound is always tighter than the Cramér–Chernoff bound [40], which is widely used by existing mechanisms [1, 6, 37] to analyze the tail probability of a PLRV through bounding its moment generating function. As a result, for a given dimension M, we can make the required σ_M scale sublinearly with M.

Since the aforementioned trade-off is effectively managed, under the same (ϵ, δ)-DP guarantee, the expected magnitude of the required noise is significantly reduced compared with the classic Gaussian noise in high-dimensional applications. In fact, our empirical guidance suggests that when fixing $\delta = 10^{-5}$, it becomes preferable to adopt our proposed product noise rather than the classic Gaussian noise whenever $M \geq 14$.

1.1 Contributions

Our main contributions are summarized as follows:

(1) New noise and new interpretation. We propose a novel product noise that facilitates the interpretation of DP through noise perturbation from the perspective of random geometry and offers an innovative characterization of PLRV as a product measure.

(2) Theory. We first rigorously derive the moment bound of the product measure to establish (ϵ, δ)-DP guarantee for our product noise. Building on this theory, we provide an approach to set the noise magnitude and conduct a systematic comparison with Gaussian noise in both low and high-dimensional settings, yielding actionable guidance for noise selection. Under the same DP guarantee, our noise has significantly lower magnitude compared with the classic Gaussian noise. We also theoretically analyze the privacy and utility guarantees when the product noise is utilized to learn machine learning models (formulated as empirical risk minimization, ERM) in a differentially private manner.

(3) Application. We apply the proposed product noise to differentially private ERM. We consider both convex ERM (logistic regression and support vector machines) and non-convex ERM (deep neural networks). For the convex scenario, we solve the ERM via both output perturbation [10, 50] and objective perturbation [26]. experiment results show that, under the same or even stricter privacy guarantees, perturbation using our proposed noise can achieve higher utility (i.e., testing accuracy) compared with other noise. We use gradient perturbation [1, 5] to solve the non-convex ERMs. Experiment results demonstrate that, to achieve comparable utility, our proposed noise requires smaller privacy parameters (ϵ and δ) compared with classic Gaussian noise, thereby providing a stronger privacy guarantee.

Roadmap. Section 2 reviews the related work. Section 3 presents the preliminaries. Section 4 provides the main results, guidance and simulation of this work. Section 5 theoretically explores the feasibility of applying the product noise to differentially private ERMs. Section 6 empirically evaluates the privacy and utility trade-off of the product noise on various ERM tasks and datasets. Finally, Section 7 concludes the paper.

2 Related Work

In this section, we review some representative works in various domains related to this work.

Noise design. Many works have attempted to develop variants of Gaussian noise to achieve (ϵ, δ)-DP. For example, the Analytic Gaussian Mechanism [2] determines noise variance using the exact Gaussian cumulative distribution function (CDF). The MVG mechanism [8] addresses matrix-valued queries with directional noise by using prior knowledge on the data to determine noise distribution parameters. Our work generates a novel noise by leveraging the polar decomposition of spherically symmetric distributions to decompose multivariate noise into the product of two independent random components, i.e., a magnitude (radius) of the noise and its direction.

Geometry insights. Our noise generation scheme allows a geometric interpretation of the PLRV in noise perturbation. In particular, we show that PLRV can be upper bounded by the random diameter of a circumcircle of a random triangle formed by two noise instances $\mathbf{n}$ and $\mathbf{n}'$ and the vector $f(\boldsymbol{x}) - f(\boldsymbol{x}')$. This further implies a product measure [14] and requires advanced measure concentration analysis to study the privacy guarantee.

The random triangle interpretation was first observed by Ji et al. [27], yet they equire a strong independence and nearly orthogonality assumption between $\mathbf{n}$ and $\mathbf{n}'$, which are used to perturb $f(\boldsymbol{x})$ and $f(\boldsymbol{x}')$, respectively. They proposed a DP mechanism that is only valid for extremely restricted privacy regimes (e.g., $\epsilon < \frac{1}{M}$). In contrast, we analyze a random triangle by lifting this less practical assumption of $\mathbf{n}$ and $\mathbf{n}'$ being independent and nearly orthogonal. This allows our proposed noise perturbation mechanism to work for all $\epsilon > 0$.

Some other works also use geometric properties, yet their considered geometry is completely different from ours. In particular, they focus on the structure and error lower bounds of DP database queries. For example, Hardt and Talwar [24] established lower bounds for linear queries by estimating the volume of the ℓ_1 unit ball under query mappings and proposed optimal mechanisms based on K-norm sampling. Nikolov et al. [38] extended this approach to (ϵ, δ)-DP by introducing minimum enclosing ellipsoids and singular value-based projections. These works share a common design principle; leveraging the geometric structure of the **query space** to guide noise mechanism design. In contrast, we focus on the random geometry in **noise space**. Reimherr et al. [41] introduce an elliptical perturbation mechanism by generalizing the Gaussian distribution to have a non-spherical covariance, enabling anisotropic noise tailored to directions of varying sensitivity. [47] considers the randomness of angular statistics and proposes to use the von Mises–Fisher distribution and Purkayastha distribution to protect the privacy of directional data.

Ways to bound PLRV. In the literature, Concentrated DP (CDP) [18] and zero-Concentrated DP (zCDP) [6] are established based on PLRV. They obtain the tail bound of PLRV by using its moment generating function (MGF). In contrast, we derive the tail bound of the PLRV using its q-th moments. An important conclusion in measure concentration theory is that the moment bound (which we adopted) for tail probabilities is always better than the ones derived using MGFs (see detailed discussions in Section 4.3). Consequently, by leveraging the sharper moment bounds, it is plausible that CDP/zCDP analyses could be further improved in the Gaussian noise setting. However, this direction of research lies beyond the scope of the present work, which focuses on designing a new noise. Additionally, our PLRV is related to a BetaPrime distribution which does not have an MGF. Hence, it is not feasible to characterize our mechanism using CDP/zCDP, which requires the underlying PLRV to be sub-Gaussian and to admit MGF. Notably, a recent work [51] proposes to directly optimize the PLRV in differentially private stochastic gradient descent (DPSGD) by considering randomized scale parameters governed by a Gamma distribution. This method enables task-specific optimization of DP-based training by adapting noise and clip parameters to the learning setup in DPSGD.

3 Preliminaries

In this section, we review some preliminaries in DP, statistics, and special functions. More details are deferred to Appendix A.1 and A.2 in the full version of this work [34].

Definition 1. *A randomized mechanism $\mathcal{M}$ satisfies (ϵ, δ)-DP if for any two neighboring datasets, $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{N}^{|\mathcal{X}|}$ that differ by only one data record, $\epsilon > 0$ and $0 < \delta < 1$, it satisfies*

$$\Pr[\mathcal{M}(\boldsymbol{x}) \in \mathcal{S}] \leq e^{\epsilon} \Pr[\mathcal{M}(\boldsymbol{x}') \in \mathcal{S}] + \delta,$$

where $\mathcal{S}$ denotes the output set.

Definition 2 (PLRV [6]). *Let $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{N}^{|\mathcal{X}|}$ be two neighboring datasets, $\mathcal{M} : \mathbb{N}^{|\mathcal{X}|} \to \mathcal{S}$ be a randomized mechanism, and P and Q be the distributions of $\mathcal{M}(\boldsymbol{x})$ and $\mathcal{M}(\boldsymbol{x}')$, respectively. PLRV associated with $\mathcal{M}$ and a randomized output $\mathbf{s}$ is defined as $\mathrm{PLRV}_{\mathcal{M}}(s) = \ln\left(\frac{\mathrm{d}P(\mathcal{M}(\boldsymbol{x})=\mathbf{s})}{\mathrm{d}Q(\mathcal{M}(\boldsymbol{x}')=\mathbf{s})}\right)$, where $\frac{\mathrm{d}P(\mathcal{M}(\boldsymbol{x})=\mathbf{s})}{\mathrm{d}Q(\mathcal{M}(\boldsymbol{x}')=\mathbf{s})}$ denotes the Radon-Nikodym derivative of P with respect to Q evaluated at $s \in \mathcal{S}$.*

Remark 1. *Note that the traditional notation of $\frac{\mathrm{d}P(\mathcal{M}(\boldsymbol{x})=\mathbf{s})}{\mathrm{d}Q(\mathcal{M}(\boldsymbol{x}')=\mathbf{s})}$ is due to the analogies with differentiation [4]. The derivative is a measurable function which only exists when P is absolutely continuous with respect to Q, denoted as $P \ll Q$. When both P and Q are continuous and share the same reference measure, $\frac{\mathrm{d}P(\mathcal{M}(\boldsymbol{x})=\mathbf{s})}{\mathrm{d}Q(\mathcal{M}(\boldsymbol{x}')=\mathbf{s})}$ reduces to the point-wise ratio of their probability density functions (PDF).*

Definition 3 (Beta distribution and Beta Prime distribution [29]). *A random variable X has a Beta distribution, i.e., $X \sim \mathsf{Beta}(\alpha, \beta)$, if it has the following PDF $f_X(x) = \frac{1}{B(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1}$, where $\alpha > 0, \beta > 0, 0 \leq x \leq 1$, and $B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$ is the Beta function.*

A random variable X has a Beta Prime distribution, i.e., $X \sim \mathsf{BetaPrime}(\alpha, \beta)$, if it has the following PDF, $f_X(x) = \frac{1}{B(\alpha,\beta)} x^{\alpha-1}(1+x)^{-\alpha-\beta}$, where $\alpha > 0, \beta > 0, x > 0$.

Lemma 1. *Beta distributed random variable can be converted to Beta Prime distributed random variable [42].*

(1) *If $X \sim \mathsf{Beta}(\alpha, \beta)$, then $\frac{X}{1-X} \sim \mathsf{BetaPrime}(\alpha, \beta)$.*
(2) *If $X \sim \mathsf{BetaPrime}(\alpha, \beta)$, then $\frac{1}{X} \sim \mathsf{BetaPrime}(\beta, \alpha)$.*

Definition 4. *(Chi distribution [20, p. 73]) The PDF of the Chi distribution with ν degrees of freedom is $f(x; \nu) = \frac{x^{\nu-1} e^{-\frac{x^2}{2}}}{2^{\frac{\nu}{2}-1}\Gamma(\frac{\nu}{2})}$ when $x \geq 0$, and $f(x; \nu) = 0$, otherwise.*

Definition 5. *The confluent hypergeometric function of the first kind [12, p. 322] is defined as*

$${}_1F_1(a; b; z) = \sum_{k=0}^{\infty} \frac{(a)_k}{(b)_k} \frac{z^k}{k!},$$

where $(a)_k$ and $(b)_k$ is the rising factorial, and $(a)_0 = 1$, $(a)_k = a(a+1)(a+2)\cdots(a+k-1)$.

Definition 6. *The parabolic cylinder function [23, p. 1028] is*

$$D_p(z) = \frac{e^{-\frac{z^2}{4}}}{\Gamma(-p)} \int_0^{\infty} e^{-xz-\frac{x^2}{2}} x^{-p-1} dx = 2^{\frac{p}{2}} e^{-\frac{z^2}{4}} \left\{ \frac{\sqrt{\pi}}{\Gamma(\frac{1-p}{2})} {}_1F_1\left(-\frac{p}{2}; \frac{1}{2}; \frac{z^2}{2}\right) - \frac{\sqrt{2\pi}z}{\Gamma(-\frac{p}{2})} {}_1F_1\left(\frac{1-p}{2}; \frac{3}{2}; \frac{z^2}{2}\right) \right\},$$

where $p < 0$ is a real number, and ${}_1F_1(a; b; x)$ is the confluent hypergeometric function of the first kind defined in Definition 5.

4 Main Results, Guidance and Simulation

This section elaborates on the core technical components of this paper. To facilitate readability, we unfold the discussion as follows.

• In Section 4.1, we present the design principle of our new product noise based on the polar decomposition of spherical noise.

• In Section 4.2, we derive the PLRV associated with our product noise by leveraging a novel geometric insight on random triangles.

• In Section 4.3, We first construct a product measure based on the derived PLRV and then obtain a tight tail probability by analyzing the moment bound of such measure.

• In Section 4.4, we establish the privacy guarantee of our product noise by minimizing the obtained tail probability.

• In Section 4.5, we demonstrate the advantages of product noise in high-dimensional settings, provide empirical guidelines for low dimensions, and validate all theoretical claims through simulations.

4.1 A New Noise

The design principle of our new noise is inspired by the following observation. We notice that when $f(\boldsymbol{x}) \in \mathbb{R}^M$, the multivariate noise considered in (1) is spherically symmetric. This type of noise can be equivalently generated via the product between a positive random variable R (usually referred to as the radius random variable) and a directional random variable $\boldsymbol{h} \in \mathbb{S}^{M-1}$ (the unit sphere embedded in $\mathbb{R}^M$). This is known as the polar decomposition [39].

Lemma 2 (Decomposition of Spherically Symmetric Random Variable [39]). *For a spherically symmetric distributed random variable $\mathbf{n} \in \mathbb{R}^M$, one can express it as $\mathbf{n} \overset{d}{=} R\boldsymbol{h}$, where $\overset{d}{=}$ means equality in distribution, R is a continuous univariate random variable (the radius) $\Pr[R \geq 0] = 1$, and $\boldsymbol{h}$ (independent of R) is uniformly distributed on the unit sphere surface $\mathbb{S}^{M-1}$ embedded in $\mathbb{R}^M$.*

Lemma 2 offers new insights into understanding the PLRV, highlights a drawback of the classic Gaussian noise, and motivates the design of a novel product noise that can be used to perturb $f(\boldsymbol{x})$ under (ϵ, δ)-DP guarantee. To be more specific, the classic Gaussian mechanism uses multivariate noise $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_{M\times M})$ to perturb $f(\boldsymbol{x}) \in \mathbb{R}^M$. This noise can be decomposed as

$$\mathbf{n} \overset{d}{=} \sigma\mathcal{N}(\mathbf{0}, \mathbf{I}_{M\times M}) \overset{d}{=} \sigma R\boldsymbol{h},$$

where $\sigma \geq \frac{\Delta_2 f}{\epsilon}\sqrt{2\log(\frac{1.25}{\delta})}$, $R \sim \chi_M$ (Chi distribution with M degrees of freedom) and $\boldsymbol{h} \sim \mathbb{S}^{M-1}$ [p. 748, Example C.7] [39]. Clearly, the squared magnitude of the classic Gaussian noise $||\mathbf{n}||_2^2 = \sigma^2||R\boldsymbol{h}||_2^2 = \sigma^2R^2$, is a scaled Chi-squared random variable with M degrees of freedom (here $R^2 \sim \chi_M^2$ for multivariate Gaussian). This creates a potential bottleneck in controlling the utility loss of the query result, particularly in high-dimensional settings. We argue that R with a small degree of freedom is promising in improving the utility and privacy trade-off.

To show this, let a spherically symmetric noise takes the form of

$$\mathbf{z} = \sigma_M R\boldsymbol{h} \in \mathbb{R}^M,$$

where σ_M is some constant, $R \sim \chi_\nu$ (Chi random variable with ν degrees of freedom, $\nu \in [1, M]$) and $\boldsymbol{h} \sim \mathbb{S}^{M-1}$. Clearly, for a given degree of freedom ν, $\mathbb{E}[||\mathbf{z}||_2^2] = \sigma_M^2\nu \propto \nu$, thus setting $\nu = 1$ is a promising choice to reduce the noise magnitude/improve utility. Additionally, the kurtosis and skewness of $||\mathbf{z}||_2^2$ are $3 + \frac{12}{\nu}$ and $\sqrt{\frac{8}{\nu}}$, respectively [7], both of which decrease as ν increases. A smaller kurtosis indicates that outliers or extremely large values are more likely to be generated,

and a smaller skewness indicates that most samples are located in the right region of the PDF. Hence, to ensure that it is less likely to have large noise and let the mass of $||\mathbf{z}||_2^2$ concentrate in the left region of PDF, we require large values of kurtosis and skewness, which are also maximized when $\nu = 1$ [39]. Thus, in this paper, we propose to explore the following noise generation scheme

$$\boxed{\text{new product noise:} \quad \mathbf{n} = \sigma_M R\boldsymbol{h}, R \sim \chi_1 \text{ and } \boldsymbol{h} \sim \mathbb{S}^{M-1},} \tag{2}$$

where σ_M is a to-be-determined constant decided by the dimension M and privacy parameters ϵ and δ, χ_1 denotes the Chi distribution with 1 degree of freedom, and $\boldsymbol{h} \sim \mathbb{S}^{M-1}$ means uniform distribution on sphere $\mathbb{S}^{M-1}$ (embedded in $\mathbb{R}^M$).

4.2 A New PLRV

Clearly, the new noise in (2) involves the product of two different measures; one involves χ_1 (which is the random radius), and the other one involves the direction of the noise. To analyze the privacy loss caused by the additive product noise in (2), we resort to the Radon-Nikodym derivative of the product measure and recall an important Lemma below.

Lemma 3 (Product Measure [14]). *Let $(X, \mathcal{A})$ and $((Y, \mathcal{B}))$ be measurable spaces, let μ_1 and ν_1 be finite measures on $(X, \mathcal{A})$, and μ_2 and ν_2 be finite measures on $(Y, \mathcal{B})$. If $(\nu_1 \ll \mu_1)$ and $(\nu_2 \ll \mu_2)$, then $(\nu_1 \times \nu_2 \ll \mu_1 \times \mu_2)$, and $\frac{\mathrm{d}(\nu_1 \times \nu_2)}{\mathrm{d}(\mu_1 \times \mu_2)}(x, y) = \frac{\mathrm{d}\nu_1}{\mathrm{d}\mu_1}(x)\frac{\mathrm{d}\nu_2}{\mathrm{d}\mu_2}(y)$, for $\mu_1 \times \mu_2$-almost everywhere, where $\ll$ stands for absolutely continuous, $\frac{\mathrm{d}\nu_1}{\mathrm{d}\mu_1}(x)$ denotes the Radon-Nikodym derivative of ν_1 with respect to μ_1 evaluated at x, similarly for $\frac{\mathrm{d}\nu_2}{\mathrm{d}\mu_2}(y)$.*

As a result, the PLRV associated with the noise defined in (2) can be alternatively characterized as in Proposition 1. This characterization differs from the existing PLRVs discussed in Section 1 and is **not** a Gaussian-type PLRV anymore.

Proposition 1. *The PLRV of perturbation using the product noise proposed in (2) has the following upper bound*

$$\mathrm{PLRV}_{\mathcal{M}}(\mathbf{s}) \leq \frac{\Delta_2 f}{\sigma_M}\left(R + \frac{\Delta_2 f}{\sigma_M}\right)\frac{1}{\sin\theta}, \tag{3}$$

*where $R \sim \chi_1$ and θ is the **random angle** formed by a random noise $\mathbf{n}$ and $\boldsymbol{v}$. We use $\boldsymbol{v}$ to denote the difference between the computation function $f(\cdot)$ evaluated on a pair of neighboring dataset $\boldsymbol{x}$ and $\boldsymbol{x}'$, i.e., $\boldsymbol{v} \triangleq f(\boldsymbol{x}) - f(\boldsymbol{x}')$ shown in Fig. 1.*

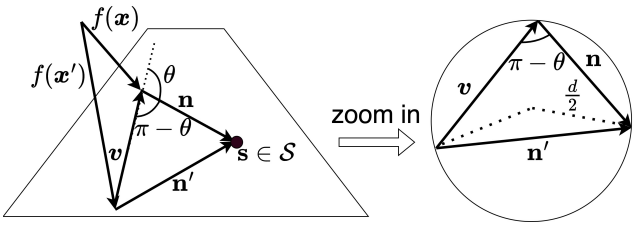


Fig. 1. Geometric interpretation of noise perturbation.

Fig. 1 geometrically visualizes that, to protect privacy, we are interested in the difference $\boldsymbol{v} \triangleq f(\boldsymbol{x}) - f(\boldsymbol{x}')$, since it is what additive noise must obscure [17, p. 264]. In other words, $\boldsymbol{v}$, $\mathbf{n}$ (used to perturb $f(\boldsymbol{x})$), and $\mathbf{n}'$ (used to perturb $f(\boldsymbol{x}')$) forms a **random** triangle, i.e., $\mathbf{n}' - \mathbf{n} = \boldsymbol{v}$. The PLRV in (3) differs from the existing works by explicitly considering this geometric interpretation. In

particular, the result of (3) represents a diameter (scaled by a constant $\frac{\Delta_2 f}{\sigma_M}$) of the circumcircle of a formed **random** triangle. This geometric idea is inspired by [27], which is the seminal work that leverages the randomness of a triangle to perform measure concentration analysis on PLRV. However, [27] imposes a strong assumption on the independence of $\mathbf{n}$ and $\mathbf{n}'$ and consider that the angle formed by $\mathbf{n}$ and $\mathbf{n}'$ are nearly orthogonal (i.e., within a range of $\pm\theta_0$ of $\frac{\pi}{2}$ for some small θ_0). The mechanism developed in [27] only works under extremely restricted privacy parameters, e.g., $\epsilon < \frac{1}{M}$. In this work, we (i) lift the assumption on the independence between $\mathbf{n}$ and $\mathbf{n}'$, (ii) consider the random angle θ formed by $\mathbf{n}$ and $f(\boldsymbol{x}) - f(\boldsymbol{x}')$, (iii) explicitly derive the closed-form PDF of $\frac{1}{\sin\theta}$. Our proposed product noise in (2) works for all $\epsilon > 0$.

Next, we provide the proof of Proposition 1.

Proof. Let $\mathbf{n} = \sigma_M R\boldsymbol{h}$ and $\mathbf{n}' = \sigma_M R'\boldsymbol{h}'$ be the noise used to perturb $f(\boldsymbol{x})$ and $f(\boldsymbol{x}')$, respectively. Then, we have $f(\boldsymbol{x}) + \mathbf{n} = f(\boldsymbol{x}') + \mathbf{n}' = \mathbf{s} \in \mathcal{S}$ and $\frac{||\mathbf{s}-f(\boldsymbol{x})||_2}{\sigma_M} \sim \chi_1$, $\frac{||\mathbf{s}-f(\boldsymbol{x}')||_2}{\sigma_M} \sim \chi_1$.

By defining the corresponding PDF notations, e.g., $f_{R\boldsymbol{h}}$, $f_{R'\boldsymbol{h}'}$, f_R, and $f_{\boldsymbol{h}}$, the PLRV associated with our product noise in (2) is

$$
\begin{aligned}
\text{PLRV}_{\mathcal{M}}(\mathbf{s}) &\overset{(a)}{=} \ln\left(\frac{f_{R\boldsymbol{h}}[f(\boldsymbol{x}) + \sigma_M R\boldsymbol{h} = \mathbf{s} \big| \mathbf{s} \in \mathcal{S}]}{f_{R\boldsymbol{h}}[f(\boldsymbol{x}') + \sigma_M R'\boldsymbol{h}' = \mathbf{s} \big| \mathbf{s} \in \mathcal{S}]}\right) \\
&= \ln\left(\frac{f_{R\boldsymbol{h}}[R\boldsymbol{h} = \frac{\mathbf{s}-f(x)}{\sigma_M} \big| \mathbf{s} \in \mathcal{S}]}{f_{R'\boldsymbol{h}'}[R'\boldsymbol{h}' = \frac{\mathbf{s}-f(\boldsymbol{x}')}{\sigma_M} \big| \mathbf{s} \in \mathcal{S}]}\right) \\
&\overset{(b)}{=} \ln\left(\frac{f_R[R = \frac{||\mathbf{s}-f(\boldsymbol{x})||_2}{\sigma_M}]}{f_{R'}[R' = \frac{||\mathbf{s}-f(\boldsymbol{x}')||_2}{\sigma_M}]} \cdot \frac{f_{\boldsymbol{h}}[\boldsymbol{h} \sim \mathbb{S}^{M-1}]}{f_{\boldsymbol{h}'}[\boldsymbol{h}' \sim \mathbb{S}^{M-1}]}\right) \\
&\overset{(c)}{=} \ln\left(\frac{f_R[R = \frac{||\mathbf{s}-f(\boldsymbol{x})||_2}{\sigma_M}]}{f_{R'}[R' = \frac{||\mathbf{s}-f(\boldsymbol{x}')||_2}{\sigma_M}]}\right) \\
&\overset{(d)}{=} \ln\left(\frac{\frac{1}{2^{-1/2}\Gamma(1/2)} e^{-\left(\frac{||\mathbf{s}-f(\boldsymbol{x})||_2}{\sigma_M}\right)^2/2}}{\frac{1}{2^{-1/2}\Gamma(1/2)} e^{-\left(\frac{||\mathbf{s}-f(\boldsymbol{x}')||_2}{\sigma_M}\right)^2/2}}\right) \\
&= \frac{1}{2\sigma_M^2}(||\mathbf{s} - f(\boldsymbol{x}')||_2 + ||\mathbf{s} - f(\boldsymbol{x})||_2)(||\mathbf{s} - f(\boldsymbol{x}')||_2 - ||\mathbf{s} - f(\boldsymbol{x})||_2) \\
&\leq \frac{1}{2\sigma_M^2}(||\mathbf{n}'||_2 + ||\mathbf{n}||_2)||\boldsymbol{v}||_2,
\end{aligned}
\tag{4}
$$

where (a) is due to Definition 2 and Remark 1, (b) follows from Lemma 3, and (c) holds because both $\boldsymbol{h}$ and $\boldsymbol{h}'$ are uniformly distributed on $\mathbb{S}^{M-1}$ and their probability densities cancel with each other. Finally, (d) is obtained by substituting in $R \sim \chi_1$ and $R' \sim \chi_1$.

Next, we proceed to bound (4) using the following geometric interpretation. For any specific output $\mathbf{s} \in \mathcal{S}$ and the noise vectors $\mathbf{n}$ and $\mathbf{n}'$ used to obscure $f(\boldsymbol{x})$ and $f(\boldsymbol{x}')$ in $\mathbb{R}^M$, we have

$$
\begin{aligned}
&\mathcal{M}(\boldsymbol{x}) = \mathcal{M}(\boldsymbol{x}') = \mathbf{s} \in \mathcal{S} \\
\Leftrightarrow &f(\boldsymbol{x}) + \mathbf{n} = f(\boldsymbol{x}') + \mathbf{n}' = \mathbf{s} \in \mathcal{S} \Leftrightarrow \mathbf{n}' - \mathbf{n} = f(\boldsymbol{x}) - f(\boldsymbol{x}') = \boldsymbol{v},
\end{aligned}
$$

which implies that $\boldsymbol{v} \triangleq f(\boldsymbol{x}) - f(\boldsymbol{x}')$, $\mathbf{n}$ and $\mathbf{n}'$ all belong to a specific hyperplane. Such a hyperplane is non-deterministic and its randomness is determined by $\mathcal{M}$. As shown in Fig. 1, under the definition of PLRV, $\boldsymbol{v}$, $\mathbf{n}$, and $\mathbf{n}'$ forms a **random** triangle, i.e., $\mathbf{n}' - \mathbf{n} = \boldsymbol{v}$. For any arbitrary $\boldsymbol{v}$, the direction of

$\mathbf{n}$ is independent of the direction of $\boldsymbol{v}$ due to the considered noise generation scheme (2). We use θ to represent the angle between $\boldsymbol{v}$ and $\mathbf{n}$, and use $\frac{d}{2}$ to represent the radius of the circumcircle (cf. Fig. 1 (right)). The upper bound of the length of each edge of a triangle is the diameter of the circumcircle of the triangle. According to the law of sines, we have $d = \frac{||\mathbf{n}'||_2}{\sin(\pi-\theta)} = \frac{||\mathbf{n}+\boldsymbol{v}||_2}{\sin\theta}$. Thus,

$$
\begin{aligned}
(4) &\leq \frac{1}{2\sigma_M^2} \cdot 2d \cdot ||\boldsymbol{v}||_2 = \frac{||\boldsymbol{v}||_2}{\sigma_M^2} \frac{||\mathbf{n}+\boldsymbol{v}||_2}{\sin(\theta)} \leq \frac{||\boldsymbol{v}||_2}{\sigma_M^2} \frac{||\mathbf{n}||_2 + ||\boldsymbol{v}||_2}{\sin\theta} \\
&\leq \frac{\Delta_2 f}{\sigma_M} \left(\frac{||\mathbf{n}||_2}{\sigma_M} + \frac{\Delta_2 f}{\sigma_M} \right) \frac{1}{\sin\theta} = \frac{\Delta_2 f}{\sigma_M} \left(R + \frac{\Delta_2 f}{\sigma_M} \right) \frac{1}{\sin\theta}, \quad R \sim \chi_1,
\end{aligned}
$$

which concludes the proof of Proposition 1. □

4.3 Moment Bound of the New PLRV

Although Proposition 1 considers an upper bound of PLRV, it turns out to have tight measure concentration and enables a tight bound on the privacy loss. Based on (3), we define two random variables

$$
W \triangleq R + \lambda, \quad U \triangleq \frac{1}{\sin\theta}, \tag{5}
$$

where $\lambda = \frac{\Delta_2 f}{\sigma_M}$ and θ is the random angle formed by the random noise $\mathbf{n}$ and the difference $\boldsymbol{v} \triangleq f(\boldsymbol{x}) - f(\boldsymbol{x}')$. Then, to achieve (ϵ, δ)-DP using the noise proposed in (2), our main technique focuses on investigating a tight measure concentration on the product of two random variables, i.e.,

$$
\Pr\left[WU \geq \frac{\sigma_M}{\Delta_2 f} \epsilon \right] \leq \delta, \tag{6}
$$

which is a sufficient condition for the product noise to satisfy $\Pr[\mathrm{PLRV}_{\mathcal{M}}(\mathbf{s}) \geq \epsilon] \leq \delta$. Usually, to study the tail bound of the product random variable, WU, one first needs to derive its PDF, f_{WU}, which involves the Mellin convolution [43] between two PDFs, i.e., f_W and f_U. To avoid this computationally heavy reasoning, we instead derive the moment bound[3] [40] on the product of W and U, which will only involve the individual PDFs due to the independence between W and U. To be more specific, we can set

$$
\Pr[WU \geq t] \leq \min_q \mathbb{E}[(WU)^q] t^{-q} = \min_q \mathbb{E}[W^q] \mathbb{E}[U^q] t^{-q} = \delta \ll 1, \tag{7}
$$

where q is positive and the equality holds due to the independence of W and U (as discussed in Lemma 2, the radius random variable is independent of the directional random variable).

To analyze (7), we first study $\mathbb{E}[W^q]$ and $\mathbb{E}[U^q]$, then derive the tail probability of $\Pr[WU \geq t]$.

4.3.1 ***Moment of*** W. According to (2), we have $\frac{||\mathbf{n}||_2}{\sigma_M} = R \sim \chi_1$. Then, the PDF of W can be obtained by a simple transformation of a random variable, i.e., $W = R + \lambda$, where $\lambda = \frac{\Delta_2 f}{\sigma_M}$. Apply Lemma 4 in Appendix A.1 in [34], and substituting $f(r;1) = \frac{\sqrt{2}}{\sqrt{\pi}} e^{-\frac{r^2}{2}}$ (Definition 4), we have

$$
f_W(w) = \sqrt{\frac{2}{\pi}} e^{-\frac{(w-\lambda)^2}{2}}, w \geq \lambda. \tag{8}
$$

Next, we can immediately bound the value of $E[W^q]$.

[3]Note that some other DP mechanisms, e.g., RDP [37] and Moments Accountant [1], essentially consider deriving a tail probability of PLRV using the Cramér-Chernoff bound, i.e., $\Pr[\Upsilon \geq t] \leq \inf_{\lambda>0} \mathbb{E}[e^{\lambda(\Upsilon - t)}]$, where Υ indicates a random variable. However, as proved in [40], moment bound for tail probabilities are always better than Cramér-Chernoff bound. More precisely, let $t > 0$, we have the best moment bound for the tail probability $\Pr[\Upsilon > t] = \min_q \mathbb{E}[\Upsilon^q] t^{-q} \leq \inf_{\lambda>0} \mathbb{E}[e^{\lambda(\Upsilon - t)}]$ always hold. Thus, our technique provides tighter bounds on PLRV.

Proposition 2. *An upper bound on the q-th moment of random variable W is*

$$\mathbb{E}[W^q] \leq 2^{-\frac{q}{2}} e^{-\frac{\lambda^2}{2}} \Gamma(q+1) \left(\frac{1}{\Gamma\left(\frac{q+2}{2}\right)} {}_1F_1\left(\frac{q+1}{2}; \frac{1}{2}; \frac{\lambda^2}{2}\right) + \frac{\sqrt{2}\lambda}{\Gamma\left(\frac{q+1}{2}\right)} {}_1F_1\left(\frac{q+2}{2}; \frac{3}{2}; \frac{\lambda^2}{2}\right) \right), \tag{9}$$

where ${}_1F_1(\cdot,\cdot,\cdot)$ is the confluent hypergeometric function of the first kind defined in Definition 5.

Proof. Based on the definition of q-th moment, we have

$$\begin{aligned}
\mathbb{E}(W^q) =& \int_\lambda^\infty w^q f_W(w) dw \overset{(a)}{=} \int_\lambda^\infty w^q \sqrt{\frac{2}{\pi}} e^{-\frac{(w-\lambda)^2}{2}} dw \\
=& \sqrt{\frac{2}{\pi}} \int_\lambda^\infty w^q e^{-\frac{1}{2}w^2} e^{\lambda w} e^{-\frac{\lambda^2}{2}} dw = \sqrt{\frac{2}{\pi}} e^{-\frac{\lambda^2}{2}} \int_\lambda^\infty w^q e^{-\frac{1}{2}w^2 + \lambda w} dw \\
\overset{(b)}{\leq}& \sqrt{\frac{2}{\pi}} e^{-\frac{\lambda^2}{2}} \int_0^\infty w^q e^{-\frac{1}{2}w^2 + \lambda w} dw \overset{(c)}{=} \sqrt{\frac{2}{\pi}} e^{-\frac{\lambda^2}{2}} \frac{D_{-q-1}(-\lambda)\Gamma(q+1)}{e^{-\frac{\lambda^2}{4}}},
\end{aligned}$$

where (a) is achieved by substituting in (8), (b) can be achieved by expanding the integration range for non-negative integrand, and (c) is obtained by applying Definition 6 with $q = -p-1$ (i.e., $p = -q-1$), $z = -\lambda$. Then, by plugging in the expression of the parabolic cylinder function $D_{-q-1}(-\lambda)$ (also see Definition 6), we can complete the proof. □

4.3.2 ***Moment of*** U. We first derive the PDF of $U \triangleq \frac{1}{\sin\theta}$, where θ is the random angle between $\boldsymbol{v}$ and $\mathbf{n}$ (see Fig. 1).

Proposition 3. *The PDF of $U \triangleq \frac{1}{\sin\theta}$ is*

$$f_U(u) = \frac{2}{\sqrt{\pi}} \frac{\Gamma\left(\frac{M}{2}\right)}{\Gamma\left(\frac{M-1}{2}\right)} (u^2-1)^{-\frac{1}{2}} u^{1-M}, \quad u \geq 1. \tag{10}$$

Proof Sketch. This proposition can be proved by applying a sequence of transformation of random variables and the connection between Beta and BetaPrime distributions in Lemma 1. In particular, we prove the following results. All detailed steps are deferred to Appendix A.3 in [34].

$$\begin{aligned}
&X \triangleq \cos\theta, f_X(x) = \frac{1}{\sqrt{\pi}} \frac{\Gamma\left(\frac{M}{2}\right)}{\Gamma\left(\frac{M-1}{2}\right)} (1-x^2)^{\frac{M-3}{2}}, x \in [-1,1], \\
&Y \triangleq 1 - X^2 = \sin^2\theta, f_Y(y) = \frac{1}{\sqrt{\pi}} \frac{\Gamma\left(\frac{M}{2}\right)}{\Gamma\left(\frac{M-1}{2}\right)} y^{\frac{M-3}{2}} \frac{1}{\sqrt{1-y}}, y \in [0,1], \\
&Z \triangleq \frac{1}{Y} = \frac{1}{\sin^2\theta}, f_Z(z) = \frac{1}{\sqrt{\pi}} \frac{\Gamma\left(\frac{M}{2}\right)}{\Gamma\left(\frac{M-1}{2}\right)} (z-1)^{-\frac{1}{2}} z^{-\frac{M}{2}}, z \in [1,\infty), \\
&U \triangleq \sqrt{Z} = \frac{1}{\sin\theta}, f_U(u) = \frac{2}{\sqrt{\pi}} \frac{\Gamma\left(\frac{M}{2}\right)}{\Gamma\left(\frac{M-1}{2}\right)} (u^2-1)^{-\frac{1}{2}} u^{1-M}, u \in [1,\infty).
\end{aligned}$$

□

Next, we arrive at the q-th moment of U.

Proposition 4. *The q-th moment $(1 < q < M-1)$ of the random variable $U \triangleq \frac{1}{\sin\theta}$ is*

$$\mathbb{E}[U^q] = \frac{\Gamma\left(\frac{M}{2}\right)}{\Gamma\left(\frac{M-1}{2}\right)} \frac{\Gamma\left(\frac{M-q}{2} - \frac{1}{2}\right)}{\Gamma\left(\frac{M-q}{2}\right)}. \tag{11}$$

Proof Sketch. Based on the definition of moments, we have

$$\mathbb{E}[U^q] = \int_1^\infty u^q f_U(u)du = \frac{2}{\sqrt{\pi}} \cdot \frac{\Gamma\left(\frac{M}{2}\right)}{\Gamma\left(\frac{M-1}{2}\right)} \int_1^\infty u^q (u^2-1)^{-\frac{1}{2}} u^{1-M} du$$
$$\overset{(*)}{=} \frac{\Gamma\left(\frac{M}{2}\right)}{\Gamma\left(\frac{M-1}{2}\right)} \cdot \frac{\Gamma\left(\frac{M-q}{2} - \frac{1}{2}\right)}{\Gamma\left(\frac{M-q}{2}\right)}, \quad q \in [1, M-1].$$

Step $(*)$ can be evaluated by invoking the kernel function of the BetaPrime $(\frac{1}{2}, \frac{M-q-1}{2})$ distribution, which requires $\frac{M-q-1}{2} > 0$. The detailed steps are deferred to Appendix A.4 in [34]. □

4.3.3 ***Obtaining the tail probability.*** Given $\mathbb{E}[W^q]$, $\mathbb{E}[U^q]$, and (7), we arrive at the following tail probability, $\Pr[WU \geq t]$.

Proposition 5. *Given any* $1 < q < M - \frac{3}{2}$, $k > 1$, *and* $t^2 > 2k^{\frac{2}{q}} \frac{\left(\frac{q+3}{2}\right)^{\left(1+\frac{2}{q}\right)}}{e^{\left(1+\frac{1}{q}\right)}}$, *we have*

$$\Pr[WU \geq t] \leq \min_q \underbrace{\frac{1}{k} \cdot \frac{e^{-\frac{\lambda^2}{2}}}{\sqrt{\pi}} \cdot \frac{{}_1F_1\left(\frac{q+1}{2}; \frac{1}{2}; \frac{\lambda^2}{2}\right) + \sqrt{2}\lambda\, {}_1F_1\left(\frac{q+2}{2}; \frac{3}{2}; \frac{\lambda^2}{2}\right)}{\sqrt{q+\frac{3}{4}}}}_{\delta_1} \cdot \underbrace{\frac{\sqrt{M-1}}{\sqrt{M-q-\frac{3}{2}}}}_{\delta_2}. \tag{12}$$

Proof Sketch. According to (7), the tail probability satisfies $\Pr[WU \geq t] \leq \min_q \mathbb{E}[W^q]\mathbb{E}[U^q]t^{-q}$. Hence, (12) can be proved by showing that $\mathbb{E}[W^q]t^{-q} \leq \delta_1$ as long as $t^2 > 2k^{\frac{2}{q}} \frac{\left(\frac{q+3}{2}\right)^{\left(1+\frac{2}{q}\right)}}{e^{\left(1+\frac{1}{q}\right)}}$ and $\mathbb{E}[U^q] \leq \delta_2$ when $q \in [1, M-\frac{3}{2}]$. Details are deferred to Appendix A.5 in [34]. □

4.4 Privacy Guarantee

Using the tail probability in Proposition 5, our main theoretical contribution is stated in Theorem 1.

Theorem 1. *Given* $f(\boldsymbol{x}) \in \mathbb{R}^M$ $(M > 2)$ *with* l_2 *sensitivity* $\Delta_2 f$ *and any constant* $k > 1$, *if we set*

$$\sigma_M^2 \geq \frac{(\Delta_2 f)^2}{\epsilon^2} t^2 \quad \textit{and} \quad t^2 = 2k^{\frac{4}{M}} \frac{\left(\frac{M}{4} + \frac{3}{2}\right)^{\left(1+\frac{4}{M}\right)}}{e^{\left(1+\frac{2}{M}\right)}},$$

then the product noise in (2) achieves (ϵ, δ)*-DP on the perturbed* $f(\boldsymbol{x})$, *where* $\epsilon > 0$. *By setting* $\lambda = \frac{\Delta_2 f}{\sigma_M}$, *the corresponding* δ *is given by*

$$\delta = \frac{e^{-\frac{\lambda^2}{2}}}{k\sqrt{\pi}} \left({}_1F_1\left(\frac{M}{4} + \frac{1}{2}; \frac{1}{2}; \frac{\lambda^2}{2}\right) + \sqrt{2}\lambda_1 F_1\left(\frac{M}{4} + 1; \frac{3}{2}; \frac{\lambda^2}{2}\right)\right) \frac{\sqrt{M-1}}{\sqrt{\frac{M}{2} - \frac{3}{2}}\sqrt{\frac{M}{2} + \frac{3}{4}}}, \tag{13}$$

where ${}_1F_1(\cdot,\cdot,\cdot)$ *is the confluent hypergeometric function of the first kind (see Definition 5).*

Proof Sketch. Combining the results of Proposition 5, (6) and (7), it is clear that (ϵ, δ)-DP can be attained if one sets $\frac{\sigma_M}{\Delta_2 f}\epsilon \geq t$, i.e., $\sigma_M^2 \geq \frac{(\Delta_2 f)^2}{\epsilon^2} t^2$. The value of δ can be determined by solving the minimization problem formulated in (12), i.e.,

$$\delta = \min_q \delta_1 \delta_2, \quad \text{where} \quad q \in \left(1, M - \frac{3}{2}\right).$$

The detailed steps to solve this optimization problem are deferred to Appendix A.6 in [34]. Our conclusion is that by selecting $q = \frac{M}{2}$, a negligible δ (e.g., $< 10^{-5}$) can be achieved. Hence, by letting $q = \frac{M}{2}$, we obtain δ as shown in (13), which completes the proof. □

In (13), the constant k serves as a tuning parameter; increasing k results in a smaller δ, as will be shown in Fig. 4(a) in Section 4.5. In high-dimensional scenarios (which is the focus of this paper), even with a large k, its impact on σ_M^2 remains minimal as the term $k^{\frac{4}{M}}$ approaches 1 when M is sufficiently large. We also provide the evaluation of the accurate δ (in (13)) and an approximate version of δ in Appendix A.6 in [34].

Theorem 1 gives an implicit recursive relationship between the noise parameter σ_M and the privacy parameters (ϵ, δ). To make it practical, we show how to obtain σ_M given the specific privacy parameters (ϵ^*, δ^*) in Algorithm 1.

Algorithm 1 Procedure to obtain σ_M in Theorem 1

Require: Dimension M; sensitivity $\Delta_2 f$; initial tuning parameter k; privacy parameters (ϵ^*, δ^*); step factor $\alpha > 1$

Ensure: Noise parameter σ_M that satisfies (ϵ^*, δ^*)-DP

1: Compute σ_M and δ using Theorem 1 with ϵ^* and k
2: **while** $\delta > \delta^*$ **do**
3: **Increase** k: $k \leftarrow \alpha k$, and **Repeat** Line 1
4: **end while**
5: **return** σ_M

4.5 Guidance and Simulation

In this section, we aim to provide an intuitive sense of how our proposed noise improves upon the classic Gaussian noise [16] and analytic Gaussian noise [2]. The comparison with the classic Gaussian noise is theoretical (considering both large and small dimension M), while that with the analytic Gaussian noise is numerical. A fully theoretically principled comparison between analytic Gaussian Mechanism and our product noise is computationally intractable. This is because our privacy parameter involves the confluent hypergeometric function, while that of the analytic Gaussian Mechanism relies on the standard error function (erf) of the Gaussian CDF; neither admits a closed-form inverse in terms of elementary functions. Consequently, for the comparison with the analytic Gaussian noise, we focused on numerical simulations and empirical evaluations.

4.5.1 Guidance on Mechanism Selection. To ensure fair comparison, we consider both our product noise and classic Gaussian noise achieve exact (ϵ, δ)-DP using the least noise scale, i.e., $\sigma_M = \frac{\Delta_2 f}{\epsilon} t$ for our product noise and $\sigma = \frac{\Delta_2 f}{\epsilon}\sqrt{2\log(\frac{1.25}{\delta})}$ for the classic Gaussian noise. We consider the ratio between the expected squared magnitude of the product noise and that of the classic Gaussian noise given a certain dimension M, i.e.,

$$f(M) = \frac{\mathbb{E}[||\mathbf{n}||_2^2]}{\mathbb{E}[||\mathbf{n}_{\text{classic}}||_2^2]} = \frac{\sigma_M^2 \mathbb{E}[\chi_1^2]}{\sigma^2 M} = \frac{\sigma_M^2}{\sigma^2 M},$$

and study the behavior of $f(M)$ when M is finite dimension and M approaches infinity. The conclusions are summarized in Corollary 1 and 2, respectively, which are all proved in Appendix A.7 in [34].

Corollary 1. *Let $\delta = 10^{-5}$, for all ϵ permitted, $f(M) < 1$ when $M \geq 14$.*

Remark 2. *Although this work targets high-dimensional settings (the minimum dimension M in our experiments presented in Section 6 is larger than 100), Corollary 1 provides practical guidance for determining when our product noise should be applied. Specifically, when dimension $M \geq 14$, it is preferable to adopt the proposed product noise over the classic Gaussian noise due to its lower noise magnitude. Furthermore, the classic Gaussian mechanism is only valid when $\epsilon \in (0, 1)$, whereas our product noise can be applied for any $\epsilon > 0$.*

Corollary 2. *For all $\delta \in (0, 1)$ and all ϵ permitted, $f(M)$ converges to $\frac{1}{4e \log\left(\frac{1.25}{\delta}\right)}$ as M approaches infinity.*

Remark 3. *In real-world applications, we usually set $\delta = \frac{1}{n}$, where n is the size of the dataset. Then, $\lim_{M\to\infty} f(M) = \mathbf{\Theta}\left(\frac{1}{\log n}\right)$; the expected magnitude of our product noise is reduced by a factor of $\mathbf{\Theta}(\log n)$ relative to the classic Gaussian noise on average. For example, when $\epsilon = 0.1$, $\delta = 10^{-5}$, and M is sufficiently large, the expected squared magnitude of our noise is approximately $\frac{1}{127}$ of that required by the classic Gaussian noise.*

4.5.2 Simulation. To corroborate Corollary 1, Fig. 2 presents the mesh plot comparing the expected squared magnitude of our product noise against the classic Gaussian noise and analytic Gaussian noise. We vary ϵ from 0 to 1, increase dimension M from 14 to 100, and fix $\delta = 10^{-5}$ and $k = 10^5$. Given the same ϵ, both classic Gaussian noise and analytic Gaussian noise (blue and green surfaces) grow linearly with dimension M. In contrast, our product noise (red surface) consistently maintains lower magnitudes, reflecting a sublinear increase in M.

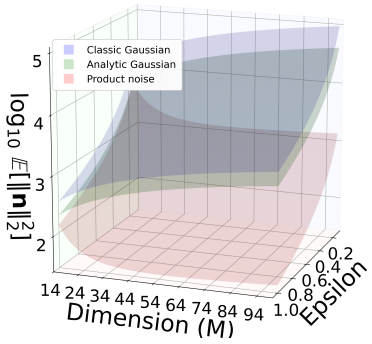


Fig. 2. $\log_{10} \mathbb{E}[||\mathbf{n}||_2^2]$ vs. M and ϵ.

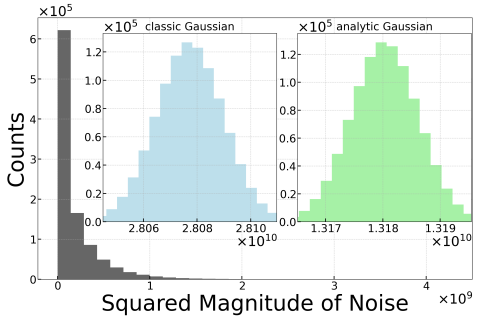
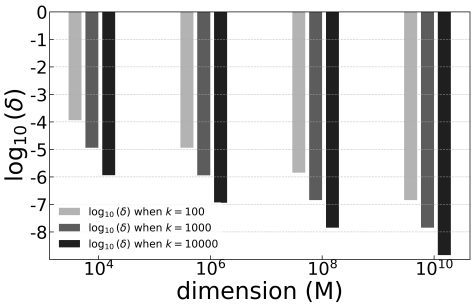


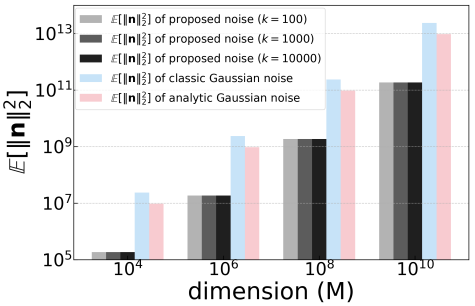
Fig. 3. Histogram of $\|\mathbf{n}\|_2^2$.

To corroborate Corollary 2, we also perform simulations in high-dimensional settings. In Fig. 3, we compare the histograms of squared noise magnitude ($||\mathbf{n}||_2^2$) achieved by our product noise and the Gaussian noises. Specifically, we set $\Delta_2 f = 1$, $\epsilon = 0.1$, $\delta = 10^{-6}$, dimension $M = 10^7$, and the constant $k = 10^5$. For each noise, we generate a million samples. As illustrated in Fig. 3, our product noise has lower squared magnitudes (e.g., on the order of 10^9). In contrast, the classic Gaussian noise and analytic Gaussian noise have high squared magnitudes (e.g., on the order of 10^{10}). Furthermore, the squared noise magnitude of our product noise is considerably more leptokurtic and right-skewed than that of the classic Gaussian noise and analytic Gaussian noise. This indicates that the kurtosis and skewness of the proposed mechanism are substantially increased compared to the classic Gaussian noise and analytic Gaussian noise, thereby enabling enhanced utility.

To verify the impact of the tuning parameter k, we also perform simulations by considering $\Delta_2 f = 1$, $\epsilon = 0.1$, and dimension $M \in \{10^4, 10^6, 10^8, 10^{10}\}$. For each M, we consider $k \in \{100, 1000, 10000\}$. First, we show the value of $\log_{10}(\delta)$ achieved by (13) in Fig. 4(a). Clearly, a small δ (i.e., $< 10^{-5}$) can be obtained by choosing a large value of the constant k. In Fig. 4(b), by fixing $\Delta_2 f = 1$ and $\epsilon = 0.1$, we evaluate the value of $\mathbb{E}[||\mathbf{n}||_2^2]$, under various M and k. Besides, we also compare with the classic Gaussian noise and analytic Gaussian noise. It is obvious that our noise magnitude (i) does not change significantly as k increases, and (ii) is far smaller than that of the classic Gaussian noise and analytic Gaussian noise.



(a) $\log_{10}(\delta)$ v.s. M



(b) $\mathbb{E}[||\mathbf{n}||_2^2]$ v.s. M

Fig. 4. Evaluations of δ and $\mathbb{E}[||\mathbf{n}||_2^2]$.

5 Applying to Privacy-Preserving Convex and Non-convex ERM

In this section, we discuss how our proposed product noise can be applied to solve empirical risk minimization (ERM) while ensuring differential privacy. Specifically, we consider both convex and non-convex ERM problems. For the convex case, we achieve DP guarantees using both output perturbation [10, 50] and objective perturbation [26]. For the non-convex case, we establish DP guarantees by leveraging gradient perturbation, specifically differentially private stochastic gradient descent (DPSGD) [1, 5].

5.1 Background on ERM

Many classic machine learning or AI tasks, e.g., regression, classification, and deep learning, can be formulated as ERM problems. In general, the formulated problem minimizes a finite sum of loss functions over a training dataset D. Given a dataset $D = \{d_1, d_2, \ldots, d_n\}$ where $d_i \in \mathbb{R}^M$. The goal of an ERM problem is to get an optimal model $\hat{\boldsymbol{\omega}} \in \mathbb{R}^M$ via solving the following optimization problem,

$$\hat{\boldsymbol{\omega}} = \arg\min_{\boldsymbol{\omega} \in \mathbb{R}^M} \mathcal{L}(\boldsymbol{\omega}; D).$$

$\mathcal{L}(\boldsymbol{\omega}; D) = \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{\omega}; d_i)$ is the empirical risk function, and $\ell(\boldsymbol{\omega}; d_i)$ is the loss function evaluated on each data sample d_i.

Convex Case. Under this case, the considered $\ell(\boldsymbol{\omega}; d_i)$ is convex and usually assumed to be L-Lipschitz (cf. Appendix A.8 in [34]). Some studies also assume $\ell(\boldsymbol{\omega}; d_i) - \ell(\boldsymbol{\omega}; d_i')$ is L-Lipschitz, such that D and D' only differ by the data of the i-th user, i.e., d_i and d_i'. In this paper, we consider using output perturbation and objective perturbation to solve convex ERM with DP guarantees.

- **Output perturbation.** This approach first obtains the optimal solution $\hat{\boldsymbol{\omega}}$ to the convex ERM and then perturbs $\hat{\boldsymbol{\omega}}$ using calibrated Gaussian [9, 10] or Laplace noise [50]. The key idea to

establish the privacy guarantee is to examine the gradient and the Hessian of functions $\mathcal{L}(\boldsymbol{\omega}; D)$ and $\ell(\boldsymbol{\omega}; d_i)$.

• **Objective perturbation.** This approach introduces a noise term (i.e., $\mathbf{n}^\top \boldsymbol{\omega}$) into the ERM objective function and minimizes the perturbed ERM using privacy-preserving solvers, e.g., the DP Frank-Wolfe [45] and stochastic gradient descent (SGD) [3, 50]. The privacy guarantee is established by leveraging the fact that both the original $\mathcal{L}(\boldsymbol{\omega}; D)$ and the noise term are differentiable everywhere; thus, there exists a unique $\boldsymbol{\omega}^*$ that optimizes both $\mathcal{L}(\boldsymbol{\omega}; D) + \mathbf{n}^\top \boldsymbol{\omega}$ and $\mathcal{L}(\boldsymbol{\omega}; D') + \mathbf{n}^\top \boldsymbol{\omega}$. Traditional objective perturbation attains DP only at the exact minimum when the gradient becomes $\mathbf{0}$, which is impractical for large-scale optimization due to practical computational limitations. Approximate Minima Perturbation (AMP) [26] improves the traditional objective perturbation by ensuring DP guarantee even with approximate minima, i.e., when the SGD algorithm does not converge. Thus, our work adopts the AMP approach in objective perturbation.

Non-convex Case. Under this case, the considered loss function $\ell(\boldsymbol{\omega}; d_i)$ is non-convex, e.g., image classification using deep neural networks. The most common approach to achieve DP in this case is to perturb the stochastic gradients using calibrated Gaussian noises [1, 5]. Since the gradients are perturbed in each iteration, it is important to accurately measure the cumulative privacy loss to make this approach practical.

5.2 Output Perturbation for Convex ERM

For convex ERM, we consider the following optimization problem,

$$\mathcal{L}(\boldsymbol{\omega}; D) = \frac{1}{n} \sum_{i=1}^{n} \ell(\boldsymbol{\omega}; d_i) + \frac{\Lambda}{2n} \|\boldsymbol{\omega}\|_2^2, \tag{14}$$

where Λ is the regularization parameter.

In the classic output perturbation, a calibrated multivariate Gaussian noise $\mathbf{n} \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$ is added to the optimal solution $\hat{\boldsymbol{\omega}}$ to guarantee (ϵ, δ)-DP. It is straightforward to replace the Gaussian noise with our product noise in the output perturbation to achieve a more favorable privacy and utility trade-off. The key steps are as follows.

First, compute the optimal solution, i.e., $\hat{\boldsymbol{\omega}} = \arg\min_{\boldsymbol{\omega} \in \mathbb{R}^M} \mathcal{L}(\boldsymbol{\omega}; D)$. Next, draw a product noise $\mathbf{n} = \sigma_M R \boldsymbol{h}$ ($R \sim \chi_1$ and $\boldsymbol{h} \sim \mathbb{S}^{M-1}$) to obfuscate $\hat{\boldsymbol{\omega}}$. The detailed steps are presented in Algorithm 2 (Product Noise-Based Output Perturbation) in Appendix B.1 in [34].

In the following, we present the privacy and utility guarantees of output perturbation using product noise. The proofs build upon the ideas in [10], with modifications on the analysis to account for the use of our product noise. We show the results in Corollary 3 and 4. The details in Appendix B.2 and B.3 in [34].

Corollary 3 (Privacy Guarantee of Product Noise-Based Output Perturbation (i.e., Algorithm 2 in Appendix B.1 in [34])). *The product noise-based Output Perturbation solves (14) in a (ϵ, δ)-DP manner if product noise $\mathbf{n} = \sigma_M R \boldsymbol{h}$ is added to the optimal solution $\hat{\boldsymbol{\omega}}$, where σ_M is $\frac{2L}{\Lambda} \cdot \frac{\sqrt{2k} \frac{2}{M} \left(\frac{M}{4} + \frac{3}{2}\right)^{\left(\frac{1}{2} + \frac{2}{M}\right)}}{\epsilon e^{\left(\frac{1}{2} + \frac{1}{M}\right)}}$. Here, δ is determined by a tuning parameter k and the dimension of the problem M (ref. (13)).*

Corollary 4 (Utility Guarantee of Product Noise-Based Output Perturbation (i.e., Algorithm 2 in Appendix B.1 in [34])). *In this algorithm, let $\hat{\boldsymbol{\omega}} = \arg\min_{\boldsymbol{\omega} \in \mathbb{R}^M} \frac{1}{n} \sum_{i=1}^{n} \ell(\boldsymbol{\omega}; d_i)$, and $\ell(\boldsymbol{\omega}; d_i)$ is L-Lipschitz, then we have the expected excess empirical risk*

$$\mathbb{E}\left[\mathcal{L}(\boldsymbol{\omega}_{\text{priv}}; D) - \mathcal{L}(\hat{\boldsymbol{\omega}}; D)\right] = L\mathbb{E}\left[\|\mathbf{n}\|_2\right].$$

Remark 4. *According to the results in Section 4, product noise exhibits much lower expected squared magnitude and a more stable noise distribution (characterized by kurtosis and skewness). It can significantly reduce the loss, $\mathcal{L}(\boldsymbol{\omega}_{\text{priv}}; D) - \mathcal{L}(\hat{\boldsymbol{\omega}}; D)$, under the same privacy parameter compared to classic Gaussian noise. We empirically evaluate this in Section 6.1 by considering differentially private logistic regression and support vector machine tasks on 6 benchmark datasets.*

5.3 Objective Perturbation for Convex ERM

In this section, we consider the product noise in objective perturbation to solve (14) in a differentially private manner. In particular, we consider the Approximate Minimum Perturbation (AMP) approach developed in [26], which is a variant of the classic objective perturbation [9, 10] and can establish DP guarantee even if (14) does not achieve the global minimum in limited SGD steps.

Similar to [26], we consider (14) satisfies the following standard assumptions, i.e., the loss function of each data point, $\ell(\boldsymbol{\omega}; d_i)$ is L-Lipschitz, convex in $\boldsymbol{\omega}$, has a continuous Hessian, and is β-smooth with respect to both $\boldsymbol{\omega}$ and d_i. Details of the common assumptions for ERM problems are provided in Appendix A.8 in [34].

According to [26], AMP introduces noise at two stages to ensure (ϵ, δ)-DP even when the optimizer does not achieve the exact minimum. To be more specific, a product noise $\mathbf{n}_1 = \sigma_{M1} R_1 \boldsymbol{h}_1$ ($R_1 \sim \chi_1$ and $\boldsymbol{h}_1 \sim \mathbb{S}^{M-1}$) is first added as a linear term to (14), i.e., $\mathcal{L}_{\text{priv}}(\boldsymbol{\omega}; D) = \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{\omega}; d_i) + \frac{\Lambda}{2n}\|\boldsymbol{\omega}\|_2^2 + \mathbf{n}_1^\top \boldsymbol{\omega}$. After obtaining an approximate minimizer of $\mathcal{L}_{\text{priv}}(\boldsymbol{\omega}; D)$ denoted as $\boldsymbol{\omega}_{\text{approx}}$, another product noise $\mathbf{n}_2 = \sigma_{M2} R_2 \boldsymbol{h}_2$ ($R_2 \sim \chi_1$ and $\boldsymbol{h}_2 \sim \mathbb{S}^{M-1}$) is added to obscure $\boldsymbol{\omega}_{\text{approx}}$. The value of σ_{M1} and σ_{M2} are calibrated using privacy parameters, Λ, and the number of training data n. The detailed steps are presented in Algorithm 3 (Product Noise-Based AMP) in Appendix C.1 in [34].

In what follows, we present the privacy and utility guarantee of Algorithm 3. Since it replaces the Gaussian noise with our proposed product noise, its privacy and utility guarantees can be established by adapting the proofs in [26]. We show the results in Corollary 5 and 6. The proof details are in Appendix C.2 and C.3 in [34].

Corollary 5 (Privacy Guarantee of Product Noise-Based AMP (i.e., Algorithm 3)). *The product noise-based AMP solves (14) in a (ϵ, δ)-DP manner if σ_{M1} and σ_{M2} are, respectively, set as $\sigma_{M1} = \frac{2L}{n} \cdot \frac{\sqrt{2}k^{\frac{2}{M}}\left(\frac{M}{4}+\frac{3}{2}\right)^{\left(\frac{1}{2}+\frac{2}{M}\right)}}{\epsilon_3 e^{\left(\frac{1}{2}+\frac{1}{M}\right)}}$ and $\sigma_{M2} = \frac{n\gamma}{\Lambda} \cdot \frac{\sqrt{2}k^{\frac{2}{M}}\left(\frac{M}{4}+\frac{3}{2}\right)^{\left(\frac{1}{2}+\frac{2}{M}\right)}}{\epsilon_2 e^{\left(\frac{1}{2}+\frac{1}{M}\right)}}$. δ is determined by a tuning parameter k and the dimension of the problem M (ref. (13)).*

Corollary 6 (Utility Guarantee of Product Noise-Based AMP (i.e., Algorithm 3)). *Let $\hat{\boldsymbol{\omega}} = \arg\min_{\boldsymbol{\omega}\in\mathbb{R}^M} \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{\omega}; d_i)$, and $r = \min\{M, 2 \cdot (\text{upper bound on rank of } \ell\text{'s Hessian})\}$. In Algorithm 3, if $\epsilon_i = \frac{\epsilon}{2}$ for $i \in \{1, 2\}$, $\epsilon_3 = \max\{\frac{\epsilon_1}{2}, \epsilon_1 - 0.99\}$, and the regularization parameter satisfies $\Lambda = \boldsymbol{\Theta}\left(\frac{1}{\|\hat{\boldsymbol{\omega}}\|_2}\left(\frac{L\gamma\sqrt{rM}}{\epsilon} + n\sqrt{\frac{L\gamma\sqrt{M}}{\epsilon}}\right)\right)$, then $\mathbb{E}\left[\mathcal{L}(\boldsymbol{\omega}_{\text{out}}; D) - \mathcal{L}(\hat{\boldsymbol{\omega}}; D)\right] = \boldsymbol{O}\left(\frac{\|\hat{\boldsymbol{\omega}}\|_2 L\sqrt{rM}}{n\epsilon} + \|\hat{\boldsymbol{\omega}}\|_2\sqrt{\frac{L\gamma\sqrt{M}}{\epsilon}}\right)$.*

Remark 5. *According to [26], $\delta = \frac{1}{n^2}$ (n is the number of training samples), then the utility loss achieved by Gaussian noise-based AMP is $\boldsymbol{O}\left(\frac{\|\hat{\boldsymbol{\omega}}\|_2 L\sqrt{rM\log n}}{n\epsilon} + \|\hat{\boldsymbol{\omega}}\|_2\sqrt{\frac{L\gamma\sqrt{M\log n}}{\epsilon}}\right)$ [26, Theorem 2]. Clearly, our product noise-based AMP reduces the utility loss by a factor of $\Theta(\sqrt{\log n})$. It implies that our method can obtain a better ERM model while maintaining equivalent privacy guarantees, and this advantage becomes more significant as the number of samples increases. In Section 6.2, we empirically validate this theoretical result through experiments on differentially private logistic regression and support vector machine tasks on 4 benchmark datasets.*

5.4 Gradient Perturbation for Non-Convex ERM

In this section, we use the product noise in gradient perturbation to solve ERM where the loss function for each data point, $\ell(\boldsymbol{\omega}; d_i)$, is non-convex. In non-convex ERM applications (e.g., deep neural networks), Differentially Private SGD (DPSGD) [1] has been widely accepted as the state-of-the-art technique. It extends the traditional SGD by enforcing gradient clipping followed by noise addition to guarantee privacy. To be more specific, given a mini-batch of training data of size I_t, each sample gradient $\boldsymbol{g}_t(d_i)$ is first clipped to ensure that its L_2-norm does not exceed a predefined threshold C, i.e., $\bar{\boldsymbol{g}}_t(d_i) = \boldsymbol{g}_t(d_i)/\max\{1, \frac{\|\boldsymbol{g}_t(d_i)\|_2}{C}\}$. Next, the clipped gradients are aggregated, and calibrated Gaussian noise $\mathbf{n}_t \sim \mathcal{N}(0, \sigma^2 C^2 \mathbf{I})$ is added, yielding the noisy gradient $\tilde{\boldsymbol{g}}_t = \frac{1}{I_t} \sum_{i \in I_t} \bar{\boldsymbol{g}}_t(d_i) + \mathbf{n}_t$.

According to our theoretical findings (Section 4), our product noise exhibits superior properties compared to classic Gaussian noise. Under the same privacy parameters, product noise significantly reduces noise magnitude. Consequently, we can replace $\mathbf{n}_t$ with the product noise in DPSGD to achieve a more favorable privacy and utility trade-off. We defer the pseudocode to Algorithm 4 (Product Noise-Based DPSGD) in Appendix D.1 in [34]. The privacy guarantee is shown in Corollary 7 and proved in Appendix D.2 in [34].

Corollary 7 (Privacy Guarantee of Product Noise-Based DPSGD (i.e., Algorithm 4)). *Given a gradient norm clipping parameter C, the product noise-based DPSGD achieves (ϵ, δ)-DP for each gradient descend step, if $\sigma_M = \frac{2\sqrt{2}C}{k}^{\frac{2}{M}} \left(\frac{M}{4} + \frac{3}{2}\right)^{\left(\frac{1}{2}+\frac{2}{M}\right)} \epsilon e^{\left(\frac{1}{2}+\frac{1}{M}\right)}$. Note δ is determined by k and the dimension of the problem M (ref. (13)).*

Remark 6. *Given a data sampling probability p and training steps T, the cumulative privacy loss of our product noise-based DPSGD is evaluated using privacy amplification via subsampling (see Theorem 2 in Appendix A.2 in [34]) followed by a distribution-independent composition [25] (see Theorem 3 in Appendix A.2 in [34]).*

The popular privacy composition techniques, e.g., the moments accountant (MA) [1] and central limit theorem (CLT)-based composition [13], cannot accommodate our product noise because they specifically require the perturbation noise to be a multivariate Gaussian. The numerical composition technique, PRV-based composition [22], is also not applicable because it requires the CDF of the PLRV. However, our PLRV involves the product of two random variables, each governed by a complex distribution, making the CDF analytically intractable.

As a result, the best composition technique that works for our product noise is the considered distribution-independent composition [25], which makes no assumptions on the noise distribution and remains applicable under arbitrary PLRV forms. In Section 6.3, we will show that our method can provide higher privacy guarantees when achieving comparable utility (training and testing accuracy).

6 Experiments

In this section, we conduct various case studies to evaluate the performance of the product noise in privacy-preserving convex and non-convex ERM. The dataset considered in each case study and the compared methods are summarized in Table 1.

Convex ERM Experiment Setups. For the convex case, we consider Logistic Regression (LR) and Huber Support Vector Machine (SVM) as applications. The corresponding loss functions are provided in Appendix E.1 of [34], and the dataset statistics are summarized in Table 5 of [34]. For output perturbation, we compare our product noise-based output perturbation with that using the classic Gaussian noise [10], analytic Gaussian noise [2], and multivariate Laplace noise (i.e., Permutation-based SGD) [50]. Regarding objective perturbation, we adopt the same strategy for hyperparameter-free (H-F) optimization in [26] and compare product noise-based AMP with H-F

Table 1. Datasets and Compared Methods Overview

DP-ERM	Compared Methods	Datasets
Convex ERM via Output Perturbation	classic Gaussian noise and analytic Gaussian noise [10] multivariate Laplace noise [50]	Adult, KDDCup99, MNIST, Synthetic-H, Real-sim, RCV1
Convex ERM via Objective Perturbation	classic Gaussian noise and analytic Gaussian noise based AMP [26]	MNIST, Synthetic-H, Real-sim, RCV1
Non-Convex ERM via Gradient Perturbation	classic DPSGD using MA [1] classic DPSGD using CLT [5] classic DPSGD using PRV [22]	Adult, IMDb, MovieLens, MNIST, CIFAR-10

AMP that uses both classic Gaussian noise and analytic Gaussian noise. For all datasets used in the convex case studies, we randomly shuffled the data and split it into 80% training and 20% testing sets. We evaluate the performance using the averaged test accuracy and standard deviation, which reflect models' stability across various experiment instances.

The goal of these convex experiment cases is to verify that, under the same privacy parameters ϵ and δ, convex ERM using our product noise achieves significantly higher utility (i.e., test accuracy) than using alternative noise (e.g., such as Gaussian and Laplace noises). We highlight the key observations of convex experiments as follows.

Observation 1 (Higher Utility under the Same or Stricter Privacy Guarantees). *The model using product noise consistently achieves higher test accuracy in all considered tasks for convex ERM under the same privacy parameter. Notably, on specific datasets, the performance even surpasses that of the non-private baseline. For some datasets, even under stricter privacy guarantees (i.e., smaller ϵ), the product noise-based method achieves higher accuracy than the other noise in convex ERM (ref. Section 6.1 and 6.2).*

Observation 2 (Higher Stability under the Same Privacy Guarantees). *In the convex ERM, perturbation using product noise significantly reduces the model utility fluctuation (measured in terms of the standard deviation of test accuracy) under the same privacy guarantee (ref. Section 6.1 and 6.2).*

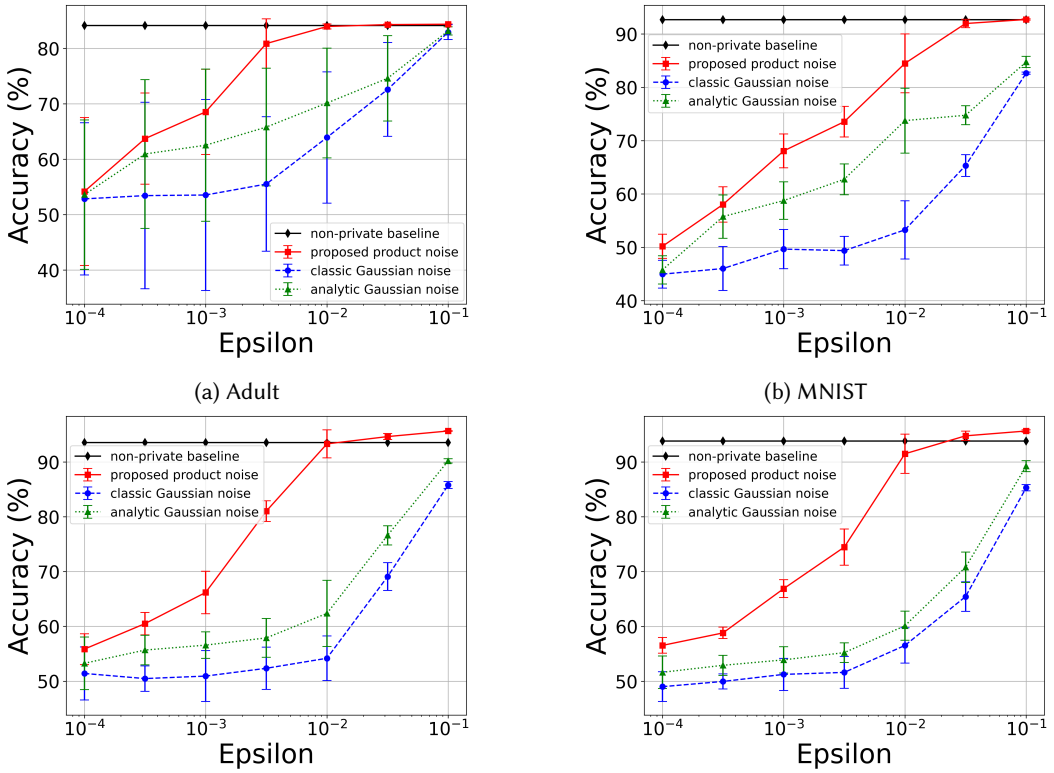
Non-convex ERM Experiment Setups. For the non-convex case, we consider various neural networks as classification models. In this case, we compare product noise-based DPSGD with the classic DPSGD using Gaussian noise. As discussed in Remark 6, given a specific data subsampling probability and training epoch, the cumulative privacy loss of our approach is evaluated using privacy amplification followed by a distribution-independent composition [25]. In contrast, the cumulative privacy loss caused by standard DPSGD is evaluated using the MA [1], PRV [22], and CLT [5] approaches.

The non-convex experiments aim to validate that, under comparable utility (i.e., test accuracy), our product noise-based DPSGD achieves stronger privacy guarantees (characterized by smaller privacy parameters ϵ and δ) than the classic DPSGD. The experiment results are consistent with our analysis in Section 4.5, which shows that when introducing noise of comparable magnitude, our mechanism requires smaller privacy parameters, thereby offering tighter privacy preservation. We highlight the key observation in non-convex ERM as follows.

Observation 3 (Stronger Privacy Guarantees under Comparable Utility). *To achieve comparable utility (i.e., testing and training accuracy across epochs) in non-convex ERM, models trained with product noise require smaller privacy parameters than those using classic Gaussian noise, thus yielding stronger privacy preservation (ref. Section 6.3).*

6.1 Case Study I: Output Perturbation

This case study evaluates the privacy and utility trade-off achieved by product noise in output perturbation for LR and Huber SVM tasks. As described in the experiment setups, we compare product noise-based output perturbation with the classic Gaussian noise [10], analytic Gaussian noise [2], and the multivariate Laplace noise [50]. The regularization parameter Λ is set to 10^{-2} for Adult and KDDCup99 and 10^{-4} for MNIST, Synthetic-H, Real-sim, and RCV1. The Lipschitz constant L is fixed at 1, and the tuning parameter k is set to 1000 for all datasets. To cover a wide range of privacy parameters, we vary ϵ from $\{10^{-4}, 10^{-\frac{7}{2}}, 10^{-3}, 10^{-\frac{5}{2}}, 10^{-2}, 10^{-\frac{3}{2}}, 10^{-1}\}$. For privacy parameter δ, we set it to $\frac{1}{n^2}$, where n is the size of the dataset. Each experiment was independently repeated 10 times.



(a) Adult (b) MNIST

(c) Synthetic-H (d) RCV1

Fig. 5. Test accuracy of output perturbation with logistic regression (LR). Product noise vs. classic Gaussian noise and analytic Gaussian noise. The experimental setup follows Chaudhuri et al. [10].

We first evaluate the performance of our product noise and Gaussian noise on LR tasks [10]. As shown in Fig. 5, product noise-based output perturbation consistently outperforms the classic Gaussian mechanism and analytic Gaussian mechanism for all considered privacy guarantees. At higher privacy regimes, our method even approaches or exceeds the performance of non-private baselines, particularly on high-dimensional datasets. On RCV1 with $\epsilon = 10^{-1}$, we achieve 95.64% test accuracy, surpassing the non-private baseline (93.79%). Even under a stringent privacy setting ($\epsilon = 10^{-\frac{5}{2}}$), our method attains 74.46% test accuracy on RCV1, while the one using classic Gaussian

noise and analytic Gaussian noise achieves only 65.43% and 70.80%, respectively, under a looser privacy guarantee ($\epsilon = 10^{-\frac{3}{2}}$). These results confirm our finding in Observation 1.

In terms of utility stability, product noise-based output perturbation yields lower standard deviations of test accuracy on multiple random experiment trials across all datasets compared to the classic and analytic Gaussian mechanisms, indicating a more stable utility. For example, on the RCV1 dataset with $\epsilon = 10^{-\frac{3}{2}}$, our method achieves a standard deviation of 0.00845, significantly lower than the classic Gaussian mechanism (0.02711) and analytic Gaussian mechanism (0.02772). This supports our finding in Observation 2.

More experiments on LR and Huber SVM. The experiment results considering other datasets and using multivariate Laplace noise are provided in Appendix E.2 of [34] (see Figures 10, 11, 12, and 13). We also present the l_2 error between the private and non-private models and FPR on LR and Huber SVM (see Tabel 6, 7, 8, and 9 in Appendix E.2 of [34])

6.2 Case Study II: Objective Perturbation

This case study evaluates the privacy and utility trade-off achieved by the product noise in objective perturbation. As discussed in the experiment setups, we adopt the same hyperparameter-free (H-F) optimization strategy used in [26], which sets the Gaussian noise scale σ according to [38, Lemma 4] and can accommodate all $\epsilon > 0$. Additionally, we also consider the analytic Gaussian noise [2] in H-F-based AMP. For fair comparison, we use the same optimization parameters (e.g., Lipschitz constant and stopping criterion) provided in [26]. Privacy parameter ϵ is selected from $\{10^{-2}, 10^{-\frac{3}{2}}, 10^{-1}, 10^{-\frac{1}{2}}, 10^{0}, 10^{\frac{1}{2}}, 10^{1}\}$ and $\delta = \frac{1}{n^2}$, where n is the number of training samples.

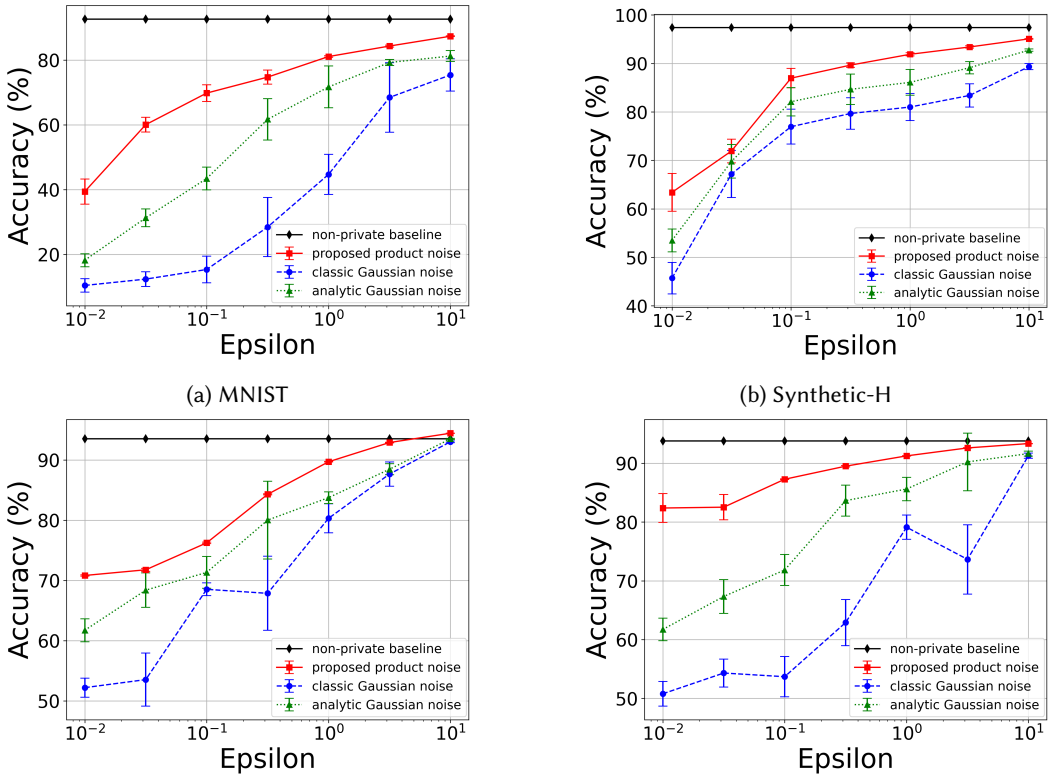
The experiment results on LR are shown in Fig. 6. It demonstrates that product noise-based AMP consistently outperforms those using Gaussian noises on all datasets under the same privacy parameters. can even surpass the non-private baselines. For example, on the Real-sim dataset with $\epsilon = 10$, our method achieves test accuracy of 94.45%, exceeding the classic Gaussian noise (93.03%), analytic Gaussian noise (93.51%) and the non-private baseline (93.52%).

Moreover, our method retains strong utility even under stringent privacy constraints, especially on high-dimensional datasets. With $\epsilon = 10^{-1}$, product noise-based AMP achieves 86.96% and 87.29% accuracy on Synthetic-H and RCV1, respectively, surpassing both classic and analytic Gaussian noise-based H-F AMP even when $\epsilon = 1$. These results further support our claim in Observation 1.

Regarding utility stability, product noise-based AMP consistently achieves lower standard deviation of test accuracy than Gaussian noise-based H-F AMP across most datasets, indicating greater stability for objective perturbation with product noise. For instance, on RCV1 at $\epsilon = 10^{\frac{1}{2}}$, the standard deviation is 0.0002 for product-noise based AMP, compared to 0.05897 for Gaussian noise-based H-F AMP and 0.04895 for analytic Gaussian noise-based H-F AMP. This further substantiates Observation 2.

In Table 2(a), we also present the l_2 error between the private and non-private models for this case study. The product noise-based Objective Perturbation consistently achieves significantly lower ℓ_2 error than both the classic and analytical Gaussian noises across all privacy parameters. Again, our method effectively mitigates learning model distortion, ensuring that the obtained ERM models stay close to the non-privacy ones.

In Table 2(b), we present the FPR obtained by various models given different ϵ. Again, our product noise achieves the lowest FPRs among all mechanisms. These results confirm that product noise maintains stable decision boundaries, reduces misclassifications, and at the same time, preserves strong DP guarantees. Combining Fig. 6 and Table 2(b), we observe that the increased accuracy of our method is accompanied by lower FPR, suggesting that the observed utility gains in certain experiments (e.g., Fig. 6(c)) can be attributed to improved generalization.



(a) MNIST (b) Synthetic-H

(c) Real-sim (d) RCV1

Fig. 6. Test accuracy of Objective Perturbation on LR.

Table 2. Objective perturbation with logistic regression (LR): ℓ_2-norm error and false positive rate (FPR).

(a) ℓ_2 error of Output Perturbation on LR.

ϵ	Mechanism	MNIST	Synthetic-H	Real-sim	RCV1
10^{-2}	classic	92.1	70.5	79.2	102.6
	analytic	22.4	32.8	36.6	53.4
	ours	**0.2**	**3.0**	**4.9**	**2.6**
10^{-1}	classic	44.6	63.6	63.5	141.9
	analytic	24.1	35.8	39.8	58.3
	ours	**6.0**	**1.6**	**0.7**	**0.1**
10^{0}	classic	44.2	70.4	77.9	102.8
	analytic	26.0	41.2	42.9	63.3
	ours	**1.3**	**5.6**	**1.3**	**0.9**
10^{1}	classic	59.5	49.3	119.1	85.7
	analytic	27.5	46.7	49.3	73.6
	ours	**4.5**	**1.2**	**1.2**	**2.3**

(b) FPR of Output Perturbation on LR.

ϵ	Mechanism	MNIST	Synthetic-H	Real-sim	RCV1
10^{-2}	classic	0.099±0.016	0.492±0.046	0.463±0.075	0.657±0.094
	analytic	0.099±0.014	0.228±0.035	0.440±0.068	0.322±0.057
	ours	**0.031±0.004**	**0.143±0.011**	**0.444±0.074**	**0.183±0.036**
10^{-1}	classic	0.084±0.015	0.420±0.015	0.199±0.022	0.477±0.082
	analytic	0.081±0.015	0.170±0.023	0.292±0.036	0.235±0.031
	ours	**0.036±0.006**	**0.077±0.009**	**0.091±0.030**	**0.027±0.004**
10^{0}	classic	0.041±0.008	0.207±0.008	0.150±0.024	0.137±0.022
	analytic	0.027±0.004	0.103±0.013	0.205±0.025	0.075±0.013
	ours	**0.022±0.002**	**0.049±0.005**	**0.020±0.009**	**0.048±0.010**
10^{1}	classic	0.022±0.004	0.120±0.004	0.119±0.010	0.065±0.008
	analytic	0.016±0.002	0.073±0.007	0.105±0.002	0.064±0.012
	ours	**0.015±0.003**	**0.029±0.002**	**0.007±0.000**	**0.057±0.006**

Remark 7. *Notably, although differential privacy is traditionally perceived to incur performance degradation, our experiments reveal a contrasting observation, i.e., in specific datasets (e.g., Real-sim in Fig. 6(c), the product noise-based private models avoid accuracy loss and even outperform the*

non-private baselines in test accuracy. This occurs because the non-private baseline may overfit the training data, particularly when the number of samples is of the same order as the data dimension (e.g., the Synthetic-H, Real-sim, and RCV1 datasets). The addition of DP noise through output or objective perturbation acts as a form of regularization, which suppresses overfitting and thereby improves generalization performance on unseen test data. In general, when the number of features (dimensions) is comparable to or exceeds the number of training samples, the model has sufficient flexibility to "memorize" the training data, leading to overfitting. In such cases, DP noise can act as an effective regularizer to mitigate overfitting. This phenomenon has also been observed in prior works, e.g., [3, 15, 26].

The experiments on Huber SVM using objective perturbation are similar to the LR tasks, i.e., product noise-based AMP can achieve higher test accuracy and utility stability. The detailed plots and tables are shown in Appendix E.3 of [34] (see Figure 14, Table 10 and 11).

6.3 Case Study III: Gradient Perturbation

In this case study, we first corroborate that, with less privacy leakage (measured in terms of cumulative ϵ), our proposed product-noise-based-DPSGD can achieve utility that is comparable with classic DPSGD [1]. Then, we show that we can provide the same level of robustness against membership inference attacks [48].

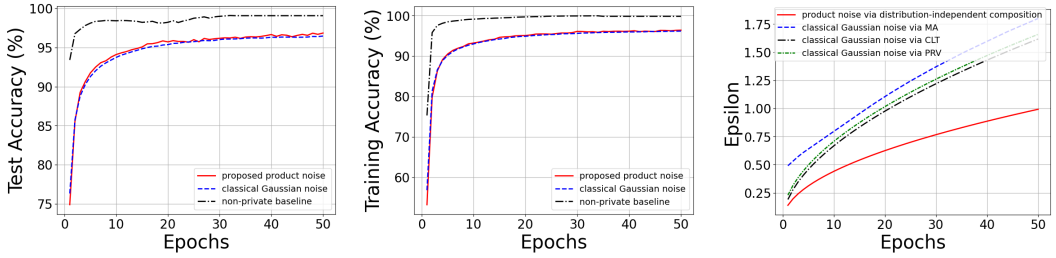
6.3.1 Comparable Utility, Yet Higher Privacy. First, we compare our product noise method with the classic Gaussian noise under identical training settings. More specifically, we aim to compare the privacy guarantees offered by each method when achieving comparable utility. We compute the cumulative privacy loss using the distribution-independent composition (Theorem 3 of [34]), and compare the results against those obtained using the MA [1], PRV [22], and CLT [5] composition approaches.

The experiments are conducted on 5 datasets: Adult, MovieLens, IMDb, MNIST, and CIFAR-10. Detailed setups for various datasets, including batch size, learning rate, gradient clipping threshold, noise multiplier (only used by classic DPSGD), initial ϵ, and tuning parameter k (initial ϵ and k only used by our product noise-based DPSGD) are summarized in Table 12 (Appendix E.4 of [34]). Each experiment is independently repeated 5 times.

MNIST. This dataset [32] contains $60,000$ training images and $10,000$ test images. We construct a convolutional neural network with the same architecture as in [5]. To ensure that different methods achieve comparable utility, we select the key training parameters as follows: the learning rate η_t is set to 0.15. For the Gaussian noise-based DPSGD, we set the noise multiplier $\sigma = 1.3$, and the gradient clipping norm is fixed at $C = 1.0$. For our product noise-based DPSGD, we set the initial privacy parameter $\epsilon = 0.3$ and the tuning parameter $k = 40,000$.

As shown in Fig. 7(a) and (b), our product noise-based DPSGD and the Gaussian noise-based DPSGD achieve comparable utility. However, our product noise-based DPSGD offers significantly stronger privacy preservation. As shown in Fig. 7(c), for 50 epochs, our method achieves $(0.99, 9.87 \times 10^{-6})$-DP (evaluated using privacy amplification followed by distribution-independent composition). This is notably smaller than the privacy guarantees obtained by the Gaussian noise-based DPSGD under MA ($(1.80, 10^{-5})$-DP), PRV ($(1.66, 10^{-5})$-DP), and CLT ($(1.62, 10^{-5})$-DP) composition approaches. Furthermore, the smaller δ value indicates a lower failure probability for the DP guarantee. These results support our argument in Observation 3 that our method can achieve tighter privacy guarantees under comparable utility.

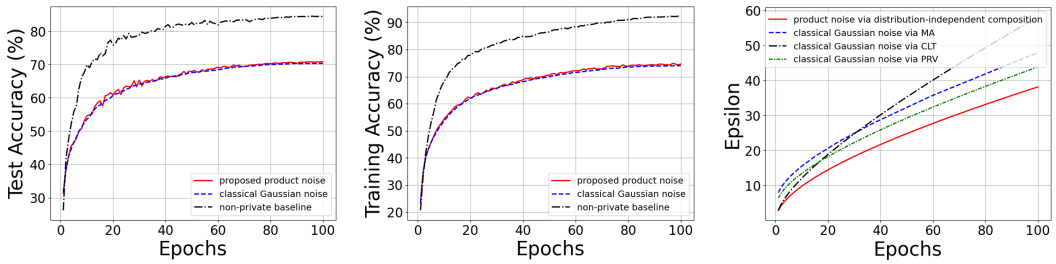
CIFAR-10. This dataset [31] consists of images belonging to 10 classes, with $50,000$ training and $10,000$ test examples. We also construct a convolutional neural network with the same architecture as in [5]. To ensure that different methods achieve comparable utility, we select the key training



(a) Test Accuracy vs Epochs (b) Training Accuracy vs Epochs (c) Epsilon vs Epochs

Fig. 7. DPSGD results on MNIST: Test accuracy, training accuracy, and privacy parameter (ϵ) over epochs.

parameters as follows: the learning rate η_t is set to 0.25. For the Gaussian noise-based DPSGD, we set the noise multiplier $\sigma = 0.50$, and the gradient clipping norm is fixed at $C = 1.5$. For our product noise-based DPSGD, we set the initial privacy parameter $\epsilon = 1.8$ and the tuning parameter $k = 300,000$.



(a) Test Accuracy vs Epochs (b) Training Accuracy vs Epochs (c) Epsilon vs Epochs

Fig. 8. DPSGD results on CIFAR-10: Test accuracy, training accuracy, and privacy parameter (ϵ) over epochs.

As shown in Fig. 8(a) and (b), our product noise-based DPSGD exhibits utility performance comparable to that of the Gaussian noise-based DPSGD throughout the training process, i.e., both methods also show a similar trend regarding training and testing accuracy. However, under a comparable utility, our product noise method demonstrates a notable advantage regarding privacy preservation. As illustrated in Fig. 8(c), by the 100th epoch, our method achieves a privacy guarantee of $(38.11, 9.94 \times 10^{-6})$-DP, which is substantially better than the guarantees obtained by the classical DPSGD under MA ($(47.83, 10^{-5})$-DP), PRV ($(43.80, 10^{-5})$-DP), and CLT ($(58.09, 10^{-5})$-DP) composition approaches. This result further supports the claim made in Observation 3.

Remark 8. *Our theoretical analysis (e.g., Theorem 1 and Corollary 1 and 2) focuses on the single-release setting, where the noisy result is only released once (as in Case Studies I and II). In terms of iterative settings (as in Case Study III) where the noisy results are repeatedly released, the final privacy cost depends on the considered privacy amplification and composition methods.*

In our current DPSGD experiments, the privacy loss of the baseline methods are accounted using MA/PRV/CLT approaches, which are unfortunately infeasible for our product noise. Since we have not yet developed a composition approach tailored to our noise, we use a distribution-independent, albeit loose, composition approach, which yields a pessimistic cumulative ϵ in iterative settings. This accounts for the modest privacy improvement observed in some experiments.

The experiment results on other datasets are shown in Appendix E.5 of [34], i.e., cf. Figure 15 (for adult), Figure 16 (for IMDB) and Figure 17 (for MovieLens). Similarly, DPSGD using our product noise also yields smaller privacy parameters under comparable utility.

6.3.2 Robustness against Membership Inference Attacks (MIAs). Now, we evaluate the robustness against MIAs in a black-box setting by comparing models trained with our product noise against those trained with the classic Gaussian noise. Following the experiment setups in a prior work [48], we use the tailored area under the ROC curve (AUC) as the robustness metric. In particular, [48] defines $\widetilde{\text{AUC}} \triangleq \max\{\text{AUC}, 0.5\}$, and the closer it is to 0.5, the more robust the model is against MIAs. Still, for the 5 datasets considered in Table 1, we train each corresponding model for 50 epochs, and repeat 5 times. All other settings follow Subsection 6.3.1.

Table 3. Tailored AUC ($\widetilde{\text{AUC}}$) in black-box MIAs [48].

Dataset	Classic Gaussian noise	Product noise
Adult	0.51±0.004	**0.50**±0.004
IMDb	0.51±0.005	**0.50±0.000**
MovieLens	0.50±0.000	0.50±0.000
MNIST	0.50±0.000	0.50±0.000
CIFAR-10	0.51±0.005	0.51±**0.004**

The results of tailored AUC ($\widetilde{\text{AUC}}$) are summarized in Table 3, which shows that the product noise achieves a level of resistance to MIAs comparable to that of the classic Gaussian noise; the obtained $\widetilde{\text{AUC}}$ are all around 0.5, which indicates that the black-box MIAs perform no better than random guessing the presence of a single record in the training dataset. The experiment results verify that when considering real-world privacy threats, even with a smaller noise scale, our product noise does not weaken privacy compared to the Gaussian baseline in DPSGD.

7 Conclusion

In this work, we propose a novel spherically symmetric noise to reduce utility loss and improve the accuracy of differentially private query results. The new noise leads to a new PLRV that can be presented as the product between two random variables, providing tighter measure concentration analysis and leading to small tail bound probability. In contrast, existing classic Gaussian mechanism and its variants usually characterize their PLRVs using Gaussian distributions. Simulation results show that when perturbing $f(\boldsymbol{x}) \in \mathbb{R}^M$ in high dimension, our proposed noise has expected squared magnitude far smaller than that required by the Gaussian mechanism and its variant. To validate the effectiveness of the developed noise, we apply it to privacy-preserving convex and non-convex ERM. Experiments on multiple datasets demonstrate substantial utility gains in diverse convex ERM models and notable privacy improvements in non-convex ERM models.

Acknowledgments

The work of Shuainan Liu, Tianxi Ji, and Zhongshuo Fang was supported in part by the U.S. Department of Agriculture under Grant AP25VSSP0000C026.

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep Learning with Differential Privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (Vienna, Austria) *(CCS '16)*. Association for Computing Machinery, New York, NY, USA, 308–318. doi:10.1145/2976749.2978318

[2] Borja Balle and Yu-Xiang Wang. 2018. Improving the Gaussian Mechanism for Differential Privacy: Analytical Calibration and Optimal Denoising. In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 80)*, Jennifer Dy and Andreas Krause (Eds.). PMLR, 394–403. https://proceedings.mlr.press/v80/balle18a.html

[3] Raef Bassily, Adam Smith, and Abhradeep Thakurta. 2014. Private Empirical Risk Minimization: Efficient Algorithms and Tight Error Bounds. In *Proceedings of the 2014 IEEE 55th Annual Symposium on Foundations of Computer Science (FOCS '14)*. IEEE Computer Society, USA, 464–473. doi:10.1109/FOCS.2014.56

[4] Patrick Billingsley. 2017. *Probability and measure*. John Wiley & Sons.

[5] Zhiqi Bu, Jinshuo Dong, Qi Long, and Weijie J Su. 2020. Deep learning with gaussian differential privacy. *Harvard data science review* 2020, 23 (2020), 10–1162.

[6] Mark Bun and Thomas Steinke. 2016. Concentrated Differential Privacy: Simplifications, Extensions, and Lower Bounds. In *Proceedings, Part I, of the 14th International Conference on Theory of Cryptography - Volume 9985*. Springer-Verlag, Berlin, Heidelberg, 635–658. doi:10.1007/978-3-662-53641-4_24

[7] George Casella and Roger L Berger. 2002. *Statistical inference*. Vol. 2. Duxbury Pacific Grove, CA.

[8] Thee Chanyaswad, Alex Dytso, H. Vincent Poor, and Prateek Mittal. 2018. MVG Mechanism: Differential Privacy under Matrix-Valued Query. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security* (Toronto, Canada) *(CCS '18)*. Association for Computing Machinery, New York, NY, USA, 230–246. doi:10.1145/3243734.3243750

[9] Kamalika Chaudhuri and Claire Monteleoni. 2008. Privacy-preserving logistic regression. In *Proceedings of the 22nd International Conference on Neural Information Processing Systems* (Vancouver, British Columbia, Canada) *(NIPS'08)*. Curran Associates Inc., Red Hook, NY, USA, 289–296.

[10] Kamalika Chaudhuri, Claire Monteleoni, and Anand D. Sarwate. 2011. Differentially Private Empirical Risk Minimization. *J. Mach. Learn. Res.* 12, null (July 2011), 1069–1109.

[11] Kamalika Chaudhuri, Anand D. Sarwate, and Kaushik Sinha. 2013. A near-optimal algorithm for differentially-private principal components. *J. Mach. Learn. Res.* 14, 1 (Jan. 2013), 2905–2943.

[12] Adri B Olde Daalhuis. 2010. Confluent hypergeometric functions. *NIST handbook of mathematical functions* 321 (2010), 349.

[13] Jinshuo Dong, Aaron Roth, and Weijie J. Su. 2022. Gaussian Differential Privacy. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 84, 1 (02 2022), 3–37. doi:10.1111/rssb.12454

[14] Rick Durrett. 2019. *Probability: theory and examples*. Vol. 49. Cambridge university press.

[15] Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. 2015. Generalization in adaptive data analysis and holdout reuse. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 2* (Montreal, Canada) *(NIPS'15)*. MIT Press, Cambridge, MA, USA, 2350–2358.

[16] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. 2006. Our data, ourselves: privacy via distributed noise generation. In *Proceedings of the 24th Annual International Conference on The Theory and Applications of Cryptographic Techniques* (St. Petersburg, Russia) *(EUROCRYPT'06)*. Springer-Verlag, Berlin, Heidelberg, 486–503. doi:10.1007/11761679_29

[17] Cynthia Dwork and Aaron Roth. 2014. The Algorithmic Foundations of Differential Privacy. *Found. Trends Theor. Comput. Sci.* 9, 3–4 (Aug. 2014), 211–407. doi:10.1561/0400000042

[18] Cynthia Dwork and Guy N. Rothblum. 2016. Concentrated Differential Privacy. arXiv:1603.01887 [cs.DS] https://arxiv.org/abs/1603.01887

[19] Kaitai. Fang and Yaoting. Zhang. 1990 - 1990. *Generalized multivariate analysis / Fang Kai-Tai, Zhang Yao- Ting*. Science Press, Beijing.

[20] Catherine Forbes, Merran Evans, Nicholas Hastings, and Brian Peacock. 2011. *Statistical distributions*. John Wiley & Sons.

[21] Arik Friedman and Assaf Schuster. 2010. Data mining with differential privacy. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Washington, DC, USA) *(KDD '10)*. Association for Computing Machinery, New York, NY, USA, 493–502. doi:10.1145/1835804.1835868

[22] Sivakanth Gopi, Yin Tat Lee, and Lukas Wutschitz. 2021. Numerical composition of differential privacy. In *Proceedings of the 35th International Conference on Neural Information Processing Systems (NIPS '21)*. Curran Associates Inc., Red Hook, NY, USA, Article 889, 12 pages.

[23] I. S. Gradshteyn and I. M. Ryzhik. 2007. *Table of integrals, series, and products* (seventh ed.). Elsevier/Academic Press, Amsterdam.

[24] Moritz Hardt and Kunal Talwar. 2010. On the geometry of differential privacy. In *Proceedings of the Forty-Second ACM Symposium on Theory of Computing* (Cambridge, Massachusetts, USA) *(STOC '10)*. Association for Computing Machinery, New York, NY, USA, 705–714. doi:10.1145/1806689.1806786

[25] Fengxiang He, Bohan Wang, and Dacheng Tao. 2021. Tighter Generalization Bounds for Iterative Differentially Private Learning Algorithms. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence (Proceedings of Machine Learning Research, Vol. 161)*, Cassio de Campos and Marloes H. Maathuis (Eds.). PMLR, 802–812. https://proceedings.mlr.press/v161/he21a.html

[26] Roger Iyengar, Joseph P. Near, Dawn Song, Om Thakkar, Abhradeep Thakurta, and Lun Wang. 2019. Towards Practical Differentially Private Convex Optimization. In *2019 IEEE Symposium on Security and Privacy (SP)*. 299–316. doi:10.1109/SP.2019.00001

[27] Tianxi Ji and Pan Li. 2024. Less is more: revisiting the Gaussian mechanism for differential privacy. In *Proceedings of the 33rd USENIX Conference on Security Symposium* (Philadelphia, PA, USA). USENIX Association, USA, Article 53, 18 pages.

[28] Tianxi Ji, Pan Li, Emre Yilmaz, Erman Ayday, Yanfang Ye, and Jinyuan Sun. 2021. Differentially private binary-and matrix-valued data query: An XOR mechanism. *Proceedings of the VLDB Endowment* 14, 5 (2021), 849–862.

[29] Norman L Johnson, Samuel Kotz, and Narayanaswamy Balakrishnan. 1995. *Continuous univariate distributions, volume 2*. Vol. 289. John wiley & sons.

[30] Shiva Prasad Kasiviswanathan, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. 2013. Analyzing Graphs with Node Differential Privacy. In *Theory of Cryptography*, Amit Sahai (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 457–476.

[31] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. 2009. CIFAR-10 (Canadian Institute for Advanced Research). Online. https://www.cs.toronto.edu/~kriz/cifar.html

[32] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324. doi:10.1109/5.726791

[33] Ninghui Li, Wahbeh Qardaji, Dong Su, and Jianneng Cao. 2012. PrivBasis: frequent itemset mining with differential privacy. *Proc. VLDB Endow.* 5, 11 (July 2012), 1340–1351. doi:10.14778/2350229.2350251

[34] Shuainan Liu, Tianxi Ji, Zhongshuo Fang, Lu Wei, and Pan Li. 2025. Privacy Loss of Noise Perturbation via Concentration Analysis of A Product Measure. arXiv:2512.06253 [cs.CR] https://arxiv.org/abs/2512.06253

[35] Xiyang Liu, Weihao Kong, Prateek Jain, and Sewoong Oh. 2022. DP-PCA: Statistically Optimal and Differentially Private PCA. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 29929–29943. https://proceedings.neurips.cc/paper_files/paper/2022/file/c150ebe1b9d1ca0eb61502bf979fa87d-Paper-Conference.pdf

[36] Frank McSherry and Ilya Mironov. 2009. Differentially private recommender systems: Building privacy into the Netflix Prize contenders. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Paris, France) *(KDD '09)*. Association for Computing Machinery, New York, NY, USA, 627–636. doi:10.1145/1557019.1557090

[37] Ilya Mironov. 2017. Rényi Differential Privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*. 263–275. doi:10.1109/CSF.2017.11

[38] Aleksandar Nikolov, Kunal Talwar, and Li Zhang. 2013. The geometry of differential privacy: the sparse and approximate cases. In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing* (Palo Alto, California, USA) *(STOC '13)*. Association for Computing Machinery, New York, NY, USA, 351–360. doi:10.1145/2488608.2488652

[39] Marc S Paolella. 2018. *Linear models and time-series analysis: regression, ANOVA, ARMA and GARCH*. John Wiley & Sons.

[40] Thomas K. Philips and Randolph Nelson and. 1995. The Moment Bound is Tighter than Chernoff's Bound for Positive Tail Probabilities. *The American Statistician* 49, 2 (1995), 175–178. doi:10.1080/00031305.1995.10476137

[41] Matthew Reimherr and Jordan Awan. 2019. Elliptical perturbations for differential privacy. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Curran Associates Inc., Red Hook, NY, USA, Article 914, 12 pages.

[42] Kyle Siegrist. 2017. Probability, mathematical statistics, stochastic processes.

[43] M.D. Springer. 1979. *The Algebra of Random Variables*. Wiley. https://books.google.com/books?id=qUDvAAAAMAAJ

[44] Thomas Steinke. 2022. Composition of Differential Privacy & Privacy Amplification by Subsampling. arXiv:2210.00597 [cs.CR] https://arxiv.org/abs/2210.00597

[45] Kunal Talwar, Abhradeep Thakurta, and Li Zhang. 2015. Nearly-optimal private LASSO. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 2* (Montreal, Canada) *(NIPS'15)*. MIT Press, Cambridge, MA, USA, 3025–3033.

[46] Soumya Wadhwa, Saurabh Agrawal, Harsh Chaudhari, Deepthi Sharma, and Kannan Achan. 2020. Data Poisoning Attacks against Differentially Private Recommender Systems. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 1617–1620. doi:10.1145/3397271.3401301

[47] Benjamin Weggenmann and Florian Kerschbaum. 2021. Differential Privacy for Directional Data. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security* (Virtual Event, Republic of Korea) *(CCS '21)*. Association for Computing Machinery, New York, NY, USA, 1205–1222. doi:10.1145/3460120.3484734

[48] Chengkun Wei, Minghu Zhao, Zhikun Zhang, Min Chen, Wenlong Meng, Bo Liu, Yuan Fan, and Wenzhi Chen. 2023. DPMLBench: Holistic Evaluation of Differentially Private Machine Learning. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security* (Copenhagen, Denmark) *(CCS '23)*. Association for Computing Machinery, New York, NY, USA, 2621–2635. doi:10.1145/3576915.3616593

[49] Peter H Westfall. 2014. Kurtosis as peakedness, 1905–2014. RIP. *The American Statistician* 68, 3 (2014), 191–195.

[50] Xi Wu, Fengan Li, Arun Kumar, Kamalika Chaudhuri, Somesh Jha, and Jeffrey Naughton. 2017. Bolt-on Differential Privacy for Scalable Stochastic Gradient Descent-based Analytics. In *Proceedings of the 2017 ACM International Conference on Management of Data* (Chicago, Illinois, USA) *(SIGMOD '17)*. Association for Computing Machinery, New York, NY, USA, 1307–1322. doi:10.1145/3035918.3064047

[51] Qin Yang, Nicholas Stout, Meisam Mohammady, Han Wang, Ayesha Samreen, Christopher J. Quinn, Yan Yan, Ashish Kundu, and Yuan Hong. 2025. PLRV-O: Advancing Differentially Private Deep Learning via Privacy Loss Random Variable Optimization. In *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security* (Taipei, Taiwan) *(CCS '25)*. Association for Computing Machinery, New York, NY, USA, 306–320. doi:10.1145/3719027.3765151

Received July 2025; revised October 2025; accepted November 2025