EAI Endorsed Transactions
on Scalable Information Systems
Research Article EAI.EU

Communication-Efficient Differentially Private Parameter Estimation for Distributed Networks and Systems

Yanling Wang[1,*]

[1]Zhengzhou University of Industrial Technology, Zhengzhou, 450000, Henan, China

Abstract

In emerging distributed networks and systems, large-scale data are frequently stored across cloud, edge, and multi-center nodes, making centralized statistical analysis infeasible and creating privacy leakage risks during the transmission of gradients, parameters, and intermediate statistics. To address these issues, this paper proposes a communication-efficient differentially private parameter estimation framework for distributed statistical inference. Two privacy-preserving schemes are designed by injecting multivariate Gaussian noise into transmitted first-order gradients and into both initial parameters and gradients. The analysis considers (ε, δ)-differential privacy under calibrated noise and composition accounting. Under the stated regularity, initialization, partition-growth and noise-scale conditions, the estimators remain statistically consistent and retain the convergence order of the non-private one-step estimator. Monte Carlo simulations under $\varepsilon=0.5$ and $\delta=10^{-4}$ show that the private estimators converge to the true parameter values as the number of machines or local sample size increases, while the privacy budget controls the expected privacy-accuracy tradeoff. A real-world case study on the U.S. Airline Database further demonstrates that the relative deviations between privacy-preserving estimators and non-private baselines remain around 2%, and the per-node runtimes of the two private schemes are only 0.0118 s and 0.0123 s, compared with 19.61 s for centralized computation on the sampled dataset. These results indicate that the proposed method can provide rigorous privacy protection, high communication efficiency, and practically acceptable estimation accuracy for privacy-sensitive distributed data analytics.

Keywords: data privacy, differential privacy, distributed networks and systems, communication-efficient inference, parameter estimation, federated statistical analysis, Gaussian noise mechanism

Received on 09 June 2026, accepted on 13 October 2026, published on 18 November 2026

Copyright © 2026 Yanling Wang, licensed to EAI. This is an open access article distributed under the terms of the CC BY-NC-SA 4.0, which permits copying, redistributing, remixing, transformation, and building upon the material in any medium so long as the original work is properly cited.

doi: 10.4108/eetsis.13396

1. Introduction

With advances in data collection technologies and computer computing capacity, the collection, storage and analysis of big data have become more efficient and rapid, giving rise to sophisticated big data resources. As a fundamental strategic national asset, big data has evolved into a core factor of production on par with physical assets and human capital. Driven by the booming application of big data and expanding data scale, an increasing volume of data is being shared, opened up and interactively utilised. Society expects to leverage big data to fuel economic growth, refine social governance, and upgrade governmental service provision and regulatory capabilities.

*Corresponding author. Email: wangyanling_112233@163.com

However, mismatched with such trends is the lack of clear definition for rights and liabilities concerning data ownership, collection, storage and utilisation. These institutional gaps directly impact big data applications and data security at the macro level. While generating enormous value, data also poses substantial challenges to data governance. As an integral component of the internet governance framework, data security and privacy protection have emerged as one of the primary bottlenecks constraining the development of big data applications.

China currently boasts over 800 million internet users. Amid mounting data security risks, the need to safeguard data security and personal privacy stands out prominently and merits high priority. Addressing data security and privacy protection has gradually become a key agenda for governments at all levels, industries, enterprises and academia, and constitutes a top priority for China's national big data strategy implementation and the accelerated development of Digital China.

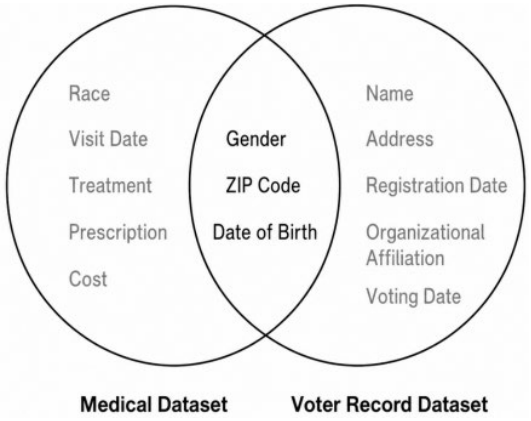


Figure 1. Example of linkage attack (based on the linkage example in [1])

Leakage or corruption of personal private data may inflict irreversible heavy losses on nations and enterprises in terms of economy, reputation and core competitiveness. A growing number of incidents such as anomalous social media breaches and credit card fraud stemming from private information disclosure have sparked widespread public concern and debate over data security and privacy protection. A representative case took place in Massachusetts, USA [1]. As illustrated in Figure 1, the left-hand dataset contains medical records released by the Massachusetts Group Insurance Commission (GIC), which insured approximately 135,000 state employees and their family members. The right-hand dataset is the Cambridge voter registration list, which Sweeney purchased for 20 US dollars [1].

Explicit identifiers, including names and addresses, were removed from the medical records, while gender, date of birth and zip code remained available as quasi-identifiers. Sweeney linked these fields to the Cambridge voter registration list and identified the then-governor's medical record [1]. This case became a canonical demonstration that de-identified records can be re-identified through linkage with external public data [1]. Likewise, de Montjoye et al. analyzed three months of credit-card transactions from 1.1 million individuals and found that four spatiotemporal transaction points were sufficient to uniquely identify 90% of individuals [2]. Related system-level safeguards have been studied in edge federated learning [3], secure aggregation [4], privacy-aware traffic-data processing [5], and cloud access control [6]. Studies of IoT-based intelligent transportation and zero-trust RFID sensing provide further application context [7,8]. Lightweight and mobility-aware federated learning in cellular and vehicular networks also indicates that communication-efficient model updating is essential for scalable distributed intelligence [9,10].

To safeguard data security, scholars have developed a suite of privacy-preserving technologies including data anonymization and encryption. Among these, the framework and criteria of Differential Privacy have garnered extensive attention, as it enables quantitative evaluation of an algorithm's privacy protection effectiveness and mitigates security risks stemming from prior background knowledge. Among the foundational works, Reference [11] developed a sensitivity-calibrated randomization framework for privacy-preserving statistical queries and characterized privacy through the indistinguishability of outputs generated from neighboring datasets. Under this framework, an algorithm's data protection capability can be rigorously quantified via mathematical and statistical means, and theoretical analysis can verify whether a computational mechanism fulfills the required privacy guarantee. A standard implementation adds sensitivity-calibrated noise to released queries or intermediate statistics. The privacy guarantee bounds the change in output distributions caused by one record, including when an adversary has auxiliary information.

Reference [12] studies distributed protocols for generating shares of random noise, including Gaussian and exponentially distributed noise, so that privacy-preserving statistical queries can be implemented without relying on a trusted database administrator. Rather than serving as a general classification of univariate noise distributions, its main contribution is secure distributed noise generation. Reference [13] subsequently systematized the foundations and principal mechanisms of Differential Privacy, including sensitivity analysis and standard noise-addition techniques. Nevertheless, inherent tradeoffs exist between big-data development goals and personal privacy protection demands: precise query outcomes can hardly be retained with minimal injected noise while enforcing privacy safeguards [14-16]. Accordingly, in-depth research is required to quantify how Differential Privacy affects general data analytics performance.

In prevalent real-world big-data application scenarios, data are widely dispersed across thousands of distinct servers. Restricted by privacy constraints and physical hardware limitations, raw data generally cannot be

aggregated for centralized computation on a single machine. Compared with conventional statistical approaches, modern data analytics mostly adopt parallel computing, distributed architectures or divide-and-conquer algorithms. Information exchange across distributed nodes inevitably requires communication transmission, which includes two-way data flows: message delivery from the central computing node to local computing terminals, and transmission of descriptive statistics from local nodes back to the central node for subsequent computation. Any outbound data transfer beyond local storage introduces inherent risks of privacy leakage, which necessitates encryption or privacy-preserving processing for transmitted information. It remains a critical open challenge to preserve statistical accuracy of distributed algorithms while fully safeguarding data confidentiality. Reference [17] establishes tight additive-error lower bounds for differentially private multi-party summation, highlighting the cost of privacy in distributed aggregation. Reference [18] develops an accurate, scalable, and verifiable protocol for federated differentially private averaging and evaluates the associated privacy-utility tradeoff. More broadly, privacy, communication efficiency, and statistical accuracy interact strongly in distributed estimation. Rather than implying an unavoidable tradeoff in every setting, Reference [19] shows that carefully designed joint encoding and decoding mechanisms can, for several canonical estimation problems, attain order-optimal accuracy under simultaneous privacy and communication constraints.

In the context of new distributed networks and systems, such privacy risks are further amplified by cloud-edge collaboration, federated statistical inference, distributed artificial intelligence, and multi-center data analysis. Although these paradigms avoid direct centralization of raw records, they still require repeated exchange of gradients, model parameters, or summary statistics across servers and edge terminals. Communication-efficient statistical inference has therefore become a key foundation for reducing cross-node transmission overhead [20-22]. At the same time, recent studies have investigated differentially private federated learning and communication-efficient privacy mechanisms [23-27], while gradient-inversion research has shown that shared model updates may reveal sensitive information even when raw data never leave local devices [28]. Therefore, privacy protection for intermediate transmitted quantities is as important as raw-data protection, especially when communication efficiency and statistical accuracy must be jointly maintained in large-scale distributed systems.

On the other hand, as a newly emerging paradigm for distributed computing, the efficient communication algorithm proposed in Reference [20] boasts prominent practical advantages over alternative methods by cutting communication overhead in distributed deployment, and algorithms built upon this philosophy have evolved into one of the mainstream research directions in distributed computing; follow-up relevant works can be found in literatures [21,22]. Nevertheless, it remains an open research problem whether tailored noise models can be constructed for such novel algorithms to satisfy differential privacy constraints while preserving the accuracy of estimation outcomes amid privacy protection.

This paper is positioned at the intersection of differential privacy, communication-efficient distributed inference, and privacy-preserving distributed systems. Specifically, it focuses on protecting intermediate information transmission in communication-efficient parameter estimation algorithms. The main contributions are threefold. First, we construct two multivariate Gaussian-noise mechanisms for transmitted gradients and parameter-gradient pairs, thereby adapting differential privacy to the communication-efficient distributed estimation framework. Second, the analysis addresses (ε,δ)-differential privacy through noise calibration and composition accounting, and establishes consistency and convergence-rate preservation under the stated regularity, initialization, partition-growth and noise-scale conditions. Third, both simulation and real-data experiments demonstrate that the proposed schemes maintain estimation accuracy while reducing communication pressure, which makes them suitable for privacy-sensitive distributed networks and multi-center analytical systems.

1.1. Related work and positioning

Differential privacy mechanisms. Differential privacy has become a rigorous mathematical foundation for privacy-preserving data analysis because it limits the influence of any single record on algorithmic outputs. Existing studies have developed Laplace, Gaussian, and distributed-noise mechanisms for query release, model training, and federated learning [11-13,23]. In recent differentially private federated learning, the protected objects have expanded from raw samples to gradients, model updates, and intermediate statistics, which is consistent with the motivation of this paper. However, many existing works emphasize model training accuracy, whereas the present study focuses on statistically interpretable parameter estimation under distributed computing.

Distributed and federated statistical inference. Communication-efficient distributed inference has been widely studied to reduce the cost of transmitting high-dimensional derivatives or local datasets [20-22]. In federated and cloud-edge systems, similar communication bottlenecks arise because local nodes are expected to exchange compact updates rather than raw data. Recent lightweight and mobility-aware federated learning studies in cellular and vehicular networks further confirm the need to reduce communication overhead while maintaining model performance in dynamic distributed environments [9,10]. Recent studies have also investigated communication-efficient private federated learning through dithering, gradient pruning, differentially private sketches, and Byzantine-robust differential privacy mechanisms [24-27]. Compared with these learning-oriented approaches, this paper studies a classical

parameter-estimation problem and derives explicit statistical consistency guarantees for privacy-aware estimators.

Privacy leakage from intermediate updates. Recent research on privacy attacks in federated learning has shown that gradients and shared model parameters may leak sensitive information through gradient inversion, model inversion, or membership inference attacks [23,28]. These findings indicate that simply keeping raw data locally is insufficient for privacy protection in distributed systems. Therefore, this paper treats transmitted gradients and initial parameters as privacy-sensitive objects and designs two noise-injection schemes to protect them while preserving the communication-efficient structure of distributed estimation.

Positioning relative to existing studies. Prior studies have characterized the interactions and tradeoffs among privacy, statistical accuracy, and communication in distributed computation [17-19], while communication-efficient inference methods have reduced transmission overhead in large-scale statistics [20-22]. The present study focuses on the local one-step estimation framework of [20], with separate protection for gradient transmission and for parameter-gradient transmission. It evaluates statistical consistency under explicit regularity, initialization, partition-growth and noise-scale conditions, together with finite-sample simulations and an airline-data case study.

2. Differential privacy under communication-efficient distributed algorithms

2.1. Differential privacy

Differential privacy bounds how the output distribution of a randomized algorithm changes between neighboring datasets. This distributional guarantee limits the influence of a single record without assuming that the adversary lacks auxiliary information [13]. The formal definitions are specified as follows:

Definition 1 Two datasets X and X' are defined as adjacent datasets if and only if they differ by exactly one single record [13].

Definition 2 An algorithm $f: D^n \to \mathbb{R}^k$ is said to satisfy (ε, δ)-differential privacy if for any pair of adjacent datasets $X, X' \in D^n$ and any measurable subset $S \subseteq \mathbb{R}^k$, the following inequality holds [13]:

$$\mathbb{P}(f(X) \in S) \leq exp(\varepsilon)\mathbb{P}(f(X') \in S) + \delta \quad (1)$$

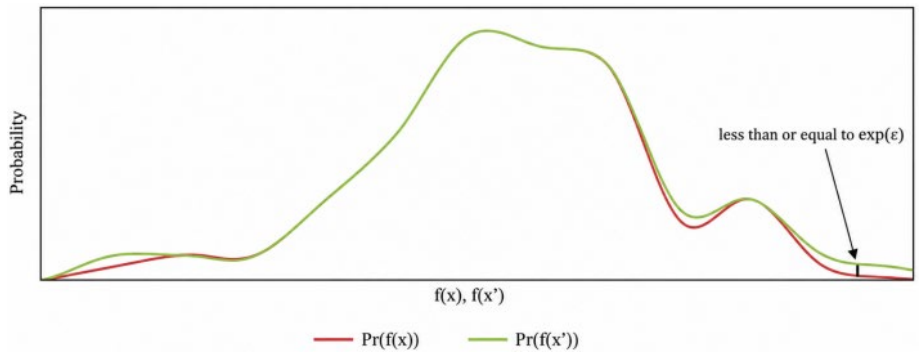


Figure 2. Schematic diagram of the differential privacy model

The differential privacy model is illustrated in Figure 2, and the specific implications of its parameters are detailed below:

The parameter ε is the privacy budget and controls the multiplicative bound on output probabilities for adjacent datasets. For fixed δ, a smaller ε gives a stronger privacy guarantee. The parameter δ is the additive slack in this probability bound; δ=0 gives pure ε-differential privacy [13]. Reference [29] gives illustrative ε values of 0.01, 0.1, 0.5 and 0.8 and describes small δ values, typically δ≤0.0001. These examples do not establish a universal recommended interval for ε.

Parameter selection should reflect the application, adjacency relation, sensitivity bounds, utility requirements and composition of all released quantities [13].

2.2. Privacy preservation based on multivariate gaussian noise mechanism

In this subsection, we develop a differential privacy preservation scheme for general data vectors. For a dataset X with sample size n, a mapping g() transforms X into a p-dimensional vector $\boldsymbol{V} = (V_1, V_2, \cdots, V_p)^T$, i.e., V=g(X). Direct publication of V risks privacy leakage since it embeds abundant raw data information; therefore, privacy-preserving processing is mandatory. We construct the following multivariate Gaussian privacy mechanism:

$$\boldsymbol{S} = \boldsymbol{V} + \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\eta}_i \quad (2)$$

where ηi denotes independent and identically distributed random noise following a multivariate Gaussian distribution with zero mean and diagonal covariance matrix $\boldsymbol{\Sigma} = diag(\sigma_1^2, \sigma_2^2, \ldots, \sigma_p^2)$, and each ηi is statistically independent of V. In accordance with Definition 2, the proposed system is designed to satisfy (ε, δ)-differential privacy.

Sensitivity calibration is made explicit as follows. For the j-th coordinate of the transmitted vector, define $\Delta_j = \sup_{X \sim X'} | g_j(X) - g_j(X')$. The release domain is assumed to be bounded by [aj,bj], so that $\Delta_j \le b_j - a_j$; under the Gaussian approximation used in Theorem 1, the stated 3σ interval provides the corresponding bounded domain for this calculation. The total privacy budget is divided across the p coordinates as ε/p, and σ_j^2 is selected as the smallest variance satisfying Equation (3) for the chosen Δj and δ. Therefore, the perturbation scale is calibrated from an explicit sensitivity bound and the prespecified privacy budget rather than from the realized transmitted value.

Theorem 1

Assume identical privacy protection strength is assigned across all dimensions, namely the privacy budget parameter for each dimension equals ε/p. Let V have mean vector $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_p)^T$ and covariance matrix $\boldsymbol{\Sigma}' = (\sigma'_{i,j}), i, j = 1,2, \ldots, p$, where $\sigma'_{i,j}$ denotes the entry at the i-th row and j-th column of $\boldsymbol{\Sigma}'$. The matrix satisfies symmetry: $\sigma'_{i,j} = \sigma'_{j,i}$ for $i \ne j$, and the diagonal entry is denoted as $\sigma'_{j,j} = \sigma_j'^2$. If the per-dimension variance σ_j^2 of the aforesaid random noise satisfies Equation (3):

$$\int_{\mu_j - \frac{\varepsilon\sigma_j^2}{6np\sigma'_j}}^{\mu_j + \frac{\varepsilon\sigma_j^2}{6np\sigma'_j}} \frac{1}{\sqrt{2\pi\left(\frac{1}{n}\sigma_j^2 + \sigma_j'^2\right)}} exp\left\{ -\frac{(x-\mu_j)^2}{2\left(\frac{1}{n}\sigma_j^2 + \sigma_j'^2\right)} \right\} dx = \delta \tag{3}$$

The proposed mechanism rigorously satisfies (ε,δ)-differential privacy.

Proof: Per Definition 2, for multivariate vector V=g(X) encoding information of dataset X, vector V'=g(X') mapped from any adjacent dataset X' of X, and arbitrary output vector S within the codomain, the following inequality holds:

$$\frac{f(\boldsymbol{S}-\boldsymbol{V})}{f(\boldsymbol{S}-\boldsymbol{V}')} \le exp(\varepsilon) \tag{4}$$

where f() denotes the probability density function of a multivariate Gaussian distribution with zero mean and covariance matrix $n^{-1}\boldsymbol{\Sigma}$. We proceed with the following algebraic simplification:

$$\begin{aligned} \frac{f(\boldsymbol{S}-\boldsymbol{V})}{f(\boldsymbol{S}-\boldsymbol{V}')} &= exp\left\{ \left| \frac{-n}{2} \left[\boldsymbol{V}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{V} - \boldsymbol{V}'^T\boldsymbol{\Sigma}^{-1}\boldsymbol{V}' - 2\boldsymbol{S}\boldsymbol{\Sigma}^{-1}(\boldsymbol{V}-\boldsymbol{V}') \right] \right| \right\} \\ &= exp\left\{ \frac{n}{2} \left| \sum_{j=1}^{p} \left[\frac{V_j^2}{\sigma_j^2} - \frac{V_j'^2}{\sigma_j^2} - \frac{2S_j(V_j - V_j')}{\sigma_j^2} \right] \right| \right\} \\ &\le exp\left\{ \frac{n}{2} \sum_{j=1}^{p} \left| \frac{(V_j - V_j')(V_j + V_j' - 2S_j)}{\sigma_j^2} \right| \right\} \le exp\{\varepsilon\}, \end{aligned} \tag{5}$$

in which Sj stands for the j-th entry of vector S. Under the assumption of identical privacy guarantee across all dimensions, Equation (5) holds when

$$\sigma_j^2 = \max_{V_j, V_j'} \frac{np}{2\varepsilon} \left| (V_j - V_j')(V_j + V_j' - 2S_j) \right| \tag{6}$$

By symmetry of Vj and Vj', we assume without loss of generality that $V_j \ge V_j'$ and $V_j, V_j' \in [a, b]$. From the definition, for arbitrary Vj and Vj', the constraint is satisfied as:

$$P\left(\frac{V_j + V_j'}{2} + \frac{\varepsilon\sigma_j^2}{np(V_j - V_j')} \ge S_j \ge \frac{V_j + V_j'}{2} - \frac{\varepsilon\sigma_j^2}{np(V_j - V_j')} \right) \le \delta \tag{7}$$

The multivariate Gaussian privacy-preserving mechanism satisfies (ε, δ)-differential privacy, which requires σ_j^2 to fulfill the following constraint:

$$P\left(\frac{a+b}{2} + \frac{\varepsilon\sigma_j^2}{np(b-a)} \ge S_j \ge \frac{a+b}{2} - \frac{\varepsilon\sigma_j^2}{np(b-a)} \right) \le \delta \tag{8}$$

In the above setup, the data vector asymptotically follows a multivariate Gaussian distribution, so Equation (8) can be evaluated under the Gaussian assumption where Vj and Vj' obey normal distributions. Let $\boldsymbol{\Sigma}' = (\sigma'_{i,j})$ denote the covariance matrix of transmitted vector V, complying with the conditions specified in the theorem. Accordingly, the variance of each dimension for random variable S generated by the multivariate Gaussian privacy mechanism equals $\frac{1}{n}\sigma_j^2 + \sigma_j'^2, j = 1,2, \ldots, p$. Further, we adopt the 3σ rule for normal distributions: for a normally distributed random variable with mean μ and standard deviation σ, approximately 99.7% of its realizations fall within the interval $[\mu - 3\sigma, \mu + 3\sigma]$. Restricting the domain of (Vj,Vj') to this 3σ interval yields $\max_{V_j, V_j'} |V_j - V_j'| = 6\sigma_j'$ and $\max_{V_j, V_j'} |V_j + V_j'| = 2\mu_j + 6\sigma_j'$, with μj denoting the mean of Vj. Combining with Inequality (4), the proposed mechanism rigorously satisfies (ε, δ)-differential privacy as long as the noise variance is no smaller than the σ_j^2 solved from Equation (3), proof completed.

2.3. Communication-efficient distributed algorithm

In this subsection, we briefly introduce communication-efficient distributed algorithms to specify the concrete formulation of the data vector requiring privacy protection, namely vector V defined in Section 2.2, within practical algorithm implementations.

For model optimization, the parameter θ is generally estimated via the loss function $\ell(\cdot): \Theta \times \mathcal{Z} \to \mathbb{R}$, with the optimization objective formulated as:

$$\boldsymbol{\theta}^* \in argmin_{\boldsymbol{\theta}\in\Theta} \mathbb{E}_{\boldsymbol{\theta}} \ell(\boldsymbol{\theta}; Z) \tag{9}$$

where Θ denotes the feasible domain of parameter θ, and $\mathcal{Z}$ stands for the sample space of all observations Z.

Since raw data are scattered across separate local computing nodes, direct empirical parameter estimation corresponding to Equation (9) is infeasible. Reference [20] proposed an alternative communication-efficient algorithm that enables parameter estimation by leveraging data stored on disparate machines. Specifically, suppose a total of N samples are split over K computing nodes with n local samples per node (N=Kn). Let zik denote the i-th sample stored on the k-th machine ($i = 1,2,\dots,n; k = 1,2,\dots,K$). We define the local loss function and global empirical loss function as below:

$$\begin{cases} \ell_k(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{\theta}; z_{ik}), \quad k = 1,2,\dots,K; \\ \ell_N(\boldsymbol{\theta}) = \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n} \ell(\boldsymbol{\theta}; z_{ik}) = \frac{1}{K}\sum_{k=1}^{K} \ell_k(\boldsymbol{\theta}). \end{cases} \tag{10}$$

We perform Taylor expansion on the global loss function as follows:

$$\begin{aligned} \ell_N(\boldsymbol{\theta}) = & \ell_N(\bar{\boldsymbol{\theta}}) + \langle \nabla \ell_N(\bar{\boldsymbol{\theta}}), \boldsymbol{\theta} - \bar{\boldsymbol{\theta}} \rangle + \sum_{j=2}^{\infty} \frac{1}{j!} \nabla^j \ell_N(\bar{\boldsymbol{\theta}})(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}})^{\otimes j} \\ = & \ell_N(\bar{\boldsymbol{\theta}}) + \langle \nabla \ell_N(\bar{\boldsymbol{\theta}}), \boldsymbol{\theta} - \bar{\boldsymbol{\theta}} \rangle + \frac{1}{2}\langle \boldsymbol{\theta} - \bar{\boldsymbol{\theta}}, \nabla^2 \ell_N(\bar{\boldsymbol{\theta}})(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}) \rangle \\ & + \sum_{j=3}^{\infty} \frac{1}{j!} \nabla^j \ell_N(\bar{\boldsymbol{\theta}})(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}})^{\otimes j}. \end{aligned} \tag{11}$$

Here ∇^j denotes the j-th order derivative of the function, and $\bar{\theta}$ stands for an initial estimator of θ satisfying the initialization conditions stated below. Since higher-order derivatives (order higher than one) are transmitted between computing nodes in matrix form, such communication induces substantial transmission overhead. Following the idea in Reference [20], we replace the higher-order derivatives of the global loss function with those from the local loss function on the first node to construct a revised objective function $\tilde{\ell}_N(\boldsymbol{\theta})$:

$$\begin{aligned} \tilde{\ell}_N(\boldsymbol{\theta}) = & \ell_N(\bar{\boldsymbol{\theta}}) + \langle \nabla \ell_N(\bar{\boldsymbol{\theta}}), \boldsymbol{\theta} - \bar{\boldsymbol{\theta}} \rangle + \frac{1}{2}\langle \boldsymbol{\theta} - \bar{\boldsymbol{\theta}}, \nabla^2 \ell_1(\bar{\boldsymbol{\theta}})(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}) \rangle \\ & + \frac{1}{2}\langle \boldsymbol{\theta} - \bar{\boldsymbol{\theta}}, \nabla^2 (\ell_N - \ell_1)(\bar{\boldsymbol{\theta}})(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}) \rangle + \sum_{j=3}^{\infty} \frac{1}{j!} \nabla^j \ell_N(\bar{\boldsymbol{\theta}})(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}})^{\otimes j} \\ = & \ell_N(\bar{\boldsymbol{\theta}}) + \langle \nabla \ell_N(\bar{\boldsymbol{\theta}}), \boldsymbol{\theta} - \bar{\boldsymbol{\theta}} \rangle + \frac{1}{2}\langle \boldsymbol{\theta} - \bar{\boldsymbol{\theta}}, \nabla^2 \ell_1(\bar{\boldsymbol{\theta}})(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}) \rangle + O\left(\frac{1}{\sqrt{n}} \|\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}\|_2^2\right) \\ \approx & \ell_N(\bar{\boldsymbol{\theta}}) + \langle \nabla \ell_N(\bar{\boldsymbol{\theta}}), \boldsymbol{\theta} - \bar{\boldsymbol{\theta}} \rangle + \frac{1}{2}\langle \boldsymbol{\theta} - \bar{\boldsymbol{\theta}}, \nabla^2 \ell_1(\bar{\boldsymbol{\theta}})(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}) \rangle. \end{aligned} \tag{12}$$

Based on Equation (12), the one-step estimation formula for model parameters is derived as:

$$\tilde{\boldsymbol{\theta}} = argmin_{\boldsymbol{\theta}\in\Theta} \tilde{\ell}_N(\boldsymbol{\theta}) = \bar{\boldsymbol{\theta}} - \left[\nabla^2 \ell_1(\bar{\boldsymbol{\theta}})\right]^{-1} \nabla \ell_N(\bar{\boldsymbol{\theta}}) \tag{13}$$

Equation (13) gives a local one-step approximation to the global empirical estimator [20]. Its communication consists of two phases across K computing nodes, corresponding to 2(K-1) point-to-point vector messages. The workflow proceeds as follows: first, the initial estimate vector $\bar{\theta}$ computed on the first node is broadcast to the remaining K-1 nodes. Each local machine then evaluates $\nabla \ell_k(\bar{\boldsymbol{\theta}})$ and sends the resulting gradient vector back to the first node, where the one-step parameter estimate is finalized.

Nevertheless, raw vector transmission carries inherent risks of privacy leakage, mandating privacy safeguards for all communicated quantities. Specifically, both $\bar{\theta}$ and $\nabla \ell_k(\bar{\boldsymbol{\theta}})$ must be protected during transmission.

2.4. Properties of privacy-preserving parameter estimation

In this subsection, we systematically analyze the asymptotic properties of parameter estimation after embedding two distinct privacy-preserving schemes into the intermediate information transmission of the aforementioned communication-efficient algorithm.

Scheme 1: Multivariate Gaussian noise defined in Section 2.2 is injected into gradient vector $\nabla \ell_k(\bar{\boldsymbol{\theta}})$ prior to transmission to the central node; the central computer executes parameter estimation using the noisy received gradients.

Scheme 2: Noise perturbation is applied simultaneously to both $\bar{\theta}$ and $\nabla \ell_k(\bar{\boldsymbol{\theta}})$. The perturbed vectors are then transmitted across all computing nodes for subsequent computation.

Let Z denote the collection of independent and identically distributed observations. To facilitate theoretical analysis on estimation properties, we impose

the following regularity assumptions, where C1, C2, C3 stand for positive constants:

Assumption A1: The parameter space Θ is a compact convex subset of $\mathbb{R}^d$, with d being the parameter dimension. The true parameter is an interior point of the parameter space, with $\boldsymbol{\theta}^* \in \Theta$ and $\sup_{\boldsymbol{\theta}\in\Theta}\|\boldsymbol{\theta}-\boldsymbol{\theta}^*\|_2 > 0$.

Assumption A2: The Fisher information matrix $I(\boldsymbol{\theta}) = \nabla^2\ell^*(\boldsymbol{\theta})$ is invertible at θ*. Namely, there exist positive constants μ-, μ+ such that $\mu_- I_d \preceq I(\boldsymbol{\theta}^*) \preceq \mu_+ I_d$, where Id is the d×d identity matrix and $\ell^*(\boldsymbol{\theta}) = \mathbb{E}_Z[\ell(\boldsymbol{\theta})]$.

Assumption A3: For any $\gamma > 0$, there exists some ε>0 satisfying:

$$\liminf_{n\to\infty} P\left\{\inf_{\|\boldsymbol{\theta}-\boldsymbol{\theta}^*\|_2 \geq \gamma}\left(\ell_j(\boldsymbol{\theta}) - \ell_j\left(\boldsymbol{\theta}^*\right)\right) \geq \varepsilon\right\} = 1 \tag{14}$$

Assumption A4: Define neighborhood $U(\rho) = \{\boldsymbol{\theta} \in \mathbb{R}^d : \|\boldsymbol{\theta}-\boldsymbol{\theta}^*\|_2 \leq \rho\}$. There exist constants G, L and a random variable M(Z) such that

$$\begin{cases} \mathbb{E}[\|\nabla\ell(\boldsymbol{\theta},Z)\|_2^{16}] \leq G^{16}, \quad \mathbb{E}[\|\nabla^2\ell(\boldsymbol{\theta},Z) - I(\boldsymbol{\theta})\|_2^{16}] \leq L^{16}, & \boldsymbol{\theta} \in U(\rho), \\ \left\|\nabla^2\ell(\boldsymbol{\theta},Z) - \nabla^2\ell(\boldsymbol{\theta}',Z)\right\|_2 \leq M(Z)\left\|\boldsymbol{\theta}-\boldsymbol{\theta}'\right\|_2, & \boldsymbol{\theta},\boldsymbol{\theta}' \in U(\rho). \end{cases} \tag{15}$$

Assumptions (A1)–(A4) follow the low-dimensional M-estimation setting in [20]. The convergence-rate comparisons below additionally require conditions on initialization, partition growth and perturbation scale.

Additional rate conditions. For the root-N comparisons in Theorems 2 and 3, the parameter dimension is fixed, the initial estimator is root-n consistent, and n, K and N=Kn grow jointly with $n \geq c\sqrt{N}$ for a fixed c>0. Gradient perturbations are mutually independent and have covariance matrices with uniformly bounded largest eigenvalues. For Scheme 2, the averaged perturbation added to the initial parameter vector is independent of the gradient perturbations and has a covariance matrix with largest eigenvalue of order 1/N. These conditions supplement (A1)–(A4).

Assumption A1 specifies the inclusion relation between the true parameter θ* and parameter space Θ;

Assumption A2 implies local convexity of the loss function at θ*, guaranteeing the existence of a local minimizer; Assumption A3 ensures global identifiability of θ*, a canonical prerequisite for proving consistency of parameter estimators; Assumption A4 is a smoothness condition that bounds the moments of derivatives of the loss function to sufficiently restrain estimation error.

In this subsection, we investigate the theoretical properties for the following two privacy-preserving schemes.

2.4.1 Parameter estimation under the first privacy-preserving scheme

The first scheme imposes privacy protection only on the transmitted gradient $\nabla\ell_j(\overline{\boldsymbol{\theta}})$. Suppose the j-th computing node satisfies (εj, δj)-differential privacy for j=1,2,…,K; then the privacy guarantee for the gradient releases is $\left(\max_j \varepsilon_j, \max_j \delta_j\right)$. By homogeneity across computing nodes, we assume without loss of generality that $\varepsilon_1 = \varepsilon_2 = \cdots = \varepsilon_K = \varepsilon$ and $\delta_1 = \delta_2 = \cdots = \delta_K = \delta$, such that the gradient-transmission channel satisfies (ε, δ)-differential privacy.

Privacy-budget composition for Scheme 1 is interpreted at the gradient-release level. The p coordinates of one gradient vector jointly use the node-level budget (εj,δj), with εj/p allocated to each coordinate in Theorem 1. Because the K local datasets are disjoint, parallel composition applies across nodes [13,29]; hence the collection of gradient releases satisfies (maxj εj, maxj δj)-differential privacy. Under homogeneous allocation, εj=ε and δj=δ, this reduces to (ε,δ)-differential privacy for the gradient-transmission channel. Scheme 1 therefore treats the initial estimator as public or non-sensitive; when the initial estimator must also be protected, Scheme 2 should be used.

Following Theorem 1, we add noise perturbation to the first-order derivative of the loss function on the j-th node during communication:

$$\nabla\ell_{j,PV}(\overline{\boldsymbol{\theta}}) = \nabla\ell_j(\overline{\boldsymbol{\theta}}) + \frac{1}{n}\sum_{i=1}^n \boldsymbol{\eta}_{ij} \tag{16}$$

Where $\boldsymbol{\eta}_{ij} \sim \mathcal{N}\left(\mathbf{0}, diag\left(\sigma_{j1}^2, \ldots, \sigma_{jd}^2\right)\right)$, and each $\sigma_{ji}^2 (i = 1,2,\ldots,d)$ satisfies Equation (3), which ensures (ε, δ)-differential privacy. For notational simplicity, define $\overline{\boldsymbol{\eta}}_j = \frac{1}{n}\sum_{i=1}^n \boldsymbol{\eta}_{ij}$. The modified objective function is written as:

$$\begin{aligned} \ell_{N,PV_1}(\boldsymbol{\theta}) = &\left\langle \frac{1}{K}\sum_{j=1}^K \left(\nabla\ell_j(\overline{\boldsymbol{\theta}}) + \overline{\boldsymbol{\eta}}_j\right), \boldsymbol{\theta} - \overline{\boldsymbol{\theta}}\right\rangle \\ &+ \frac{1}{2}\left\langle \nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\boldsymbol{\theta} - \overline{\boldsymbol{\theta}}), \boldsymbol{\theta} - \overline{\boldsymbol{\theta}}\right\rangle \end{aligned} \tag{17}$$

The resulting privacy-aware parameter estimator is:

$$\widetilde{\boldsymbol{\theta}}_{PV1} = argmin_{\boldsymbol{\theta}\in\Theta}\, \ell_{N,PV_1}(\boldsymbol{\theta}) \tag{18}$$

Theorem 2 Under Assumptions (A1)–(A4) and the additional rate conditions, $\widetilde{\boldsymbol{\theta}}_{PV1} \xrightarrow{P} \boldsymbol{\theta}'$.

Proof: The first- and second-order derivatives of the objective functions corresponding to the original distributed algorithm and the privacy-preserving modified algorithm satisfy:

$$\begin{aligned}\nabla^2\ell_{N,PV_1}(\boldsymbol{\theta}) &= \nabla^2\tilde{\ell}_N(\boldsymbol{\theta}) = \nabla^2\ell_1(\overline{\boldsymbol{\theta}}),\\ \nabla\ell_{N,PV_1}(\boldsymbol{\theta}) &= \frac{1}{K}\sum_{j=1}^{K}\left(\nabla\ell_j(\overline{\boldsymbol{\theta}})+\overline{\boldsymbol{\eta}}_j\right)+\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}),\\ \nabla\tilde{\ell}_N(\boldsymbol{\theta}) &= \frac{1}{K}\sum_{j=1}^{K}\nabla\ell_j(\overline{\boldsymbol{\theta}})+\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}).\end{aligned} \tag{19}$$

From the optimality condition $\nabla\tilde{\ell}_N(\widetilde{\boldsymbol{\theta}})=\mathbf{0}$, we obtain:

$$\nabla\ell_N(\widetilde{\boldsymbol{\theta}}) = \frac{1}{K}\sum_{j=1}^{K}\nabla\ell_j(\overline{\boldsymbol{\theta}})+\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\widetilde{\boldsymbol{\theta}}-\overline{\boldsymbol{\theta}})=\mathbf{0} \tag{20}$$

Combined with Equation (20) and the stationary condition $\nabla\ell_{N,PV_1}(\widetilde{\boldsymbol{\theta}}_{PV1})=\mathbf{0}$, we have:

$$\begin{aligned}\mathbf{0}=\nabla\ell_{N,PV_1}(\widetilde{\boldsymbol{\theta}}_{PV1}) &= \frac{1}{K}\sum_{j=1}^{K}\left(\nabla\ell_j(\overline{\boldsymbol{\theta}})+\overline{\boldsymbol{\eta}}_j\right)+\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\widetilde{\boldsymbol{\theta}}_{PV1}-\overline{\boldsymbol{\theta}})\\ &\quad+\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\widetilde{\boldsymbol{\theta}}-\overline{\boldsymbol{\theta}})\\ &= \frac{1}{K}\sum_{j=1}^{K}\overline{\boldsymbol{\eta}}_j+\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\widetilde{\boldsymbol{\theta}}_{PV1}-\widetilde{\boldsymbol{\theta}}).\end{aligned} \tag{21}$$

Rearranging yields

$$\widetilde{\boldsymbol{\theta}}_{PV1}-\widetilde{\boldsymbol{\theta}} = -\left[\nabla^2\ell_1(\overline{\boldsymbol{\theta}})\right]^{-1}\left(\frac{1}{K}\sum_{j=1}^{K}\overline{\boldsymbol{\eta}}_j\right) \tag{22}$$

Since $\bar{\theta}$ satisfies the stated initialization condition and $\frac{1}{K}\sum_{j=1}^{K}\overline{\boldsymbol{\eta}}_j=\frac{1}{N}\sum_{i=1}^{n}\sum_{j=1}^{K}\boldsymbol{\eta}_{ij}$, where each ηij is an independent zero-mean Gaussian random vector, the stated covariance bounds imply:

$$\left\|\widetilde{\boldsymbol{\theta}}_{PV1}-\widetilde{\boldsymbol{\theta}}\right\|_2 = O_p\left(\frac{1}{\sqrt{N}}\right) \tag{23}$$

Under these rate conditions, the one-step bound in [20] gives $\left\|\widetilde{\boldsymbol{\theta}}-\boldsymbol{\theta}^*\right\|_2=O_p\left(\frac{1}{\sqrt{N}}\right)$. By the triangle inequality:

$$\left\|\widetilde{\boldsymbol{\theta}}_{PV1}-\boldsymbol{\theta}^*\right\|_2 \le \left\|\widetilde{\boldsymbol{\theta}}_{PV1}-\widetilde{\boldsymbol{\theta}}\right\|_2+\left\|\widetilde{\boldsymbol{\theta}}-\boldsymbol{\theta}^*\right\|_2 = O_p\left(\frac{1}{\sqrt{N}}\right) \tag{24}$$

Therefore, $\widetilde{\boldsymbol{\theta}}_{PV1}\xrightarrow{P}\boldsymbol{\theta}^*$.

Under the additional rate conditions, Theorem 2 implies that after injecting noise into first-order gradients, the estimator obtained via distributed computation using perturbed gradient information remains consistent, with the same convergence rate as the conventional non-private estimator.

2.4.2 Parameter estimation under the second privacy-preserving scheme

The second scheme applies privacy perturbation to both $\nabla\ell_j(\overline{\boldsymbol{\theta}})$ and $\bar{\theta}$ to satisfy (εj,δj)-differential privacy. Following Theorem 1, we perturb $\bar{\theta}$ during transmission as:

For Scheme 2, the initial-parameter release and the gradient release may depend on the same local data, so sequential composition is applied within a node. We allocate (εj/2,δj/2) to the parameter perturbation and (εj/2,δj/2) to the gradient perturbation; the sequential composition property [13,29] then gives the node-level guarantee (εj,δj). Parallel composition over the disjoint node datasets yields (maxj εj, maxj δj), and therefore the homogeneous setting εj=ε, δj=δ gives an overall (ε,δ)-differential privacy guarantee for the two protected transmission layers.

$$\overline{\boldsymbol{\theta}}_{PV}(Z_1) = \overline{\boldsymbol{\theta}}(Z_1)+\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{\eta}_i \tag{25}$$

where $\boldsymbol{\eta}_i\sim\mathcal{N}(\mathbf{0},diag(\sigma_1^2,\ldots,\sigma_d^2))$, and each $\sigma_i^2(i=1,2,\ldots,d)$ satisfies Equation (3), which guarantees (ε, δ)-differential privacy. For notational brevity, define $\overline{\boldsymbol{\eta}}=\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{\eta}_i$. The modified objective function reads:

$$\begin{aligned}\ell_{N,PV_2}(\boldsymbol{\theta}) &= \left\langle\frac{1}{K}\sum_{j=1}^{K}\left(\nabla\ell_j(\overline{\boldsymbol{\theta}}+\overline{\boldsymbol{\eta}})+\overline{\boldsymbol{\eta}}_j\right),\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}\right\rangle\\ &\quad+\frac{1}{2}\left\langle\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}),\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}\right\rangle\end{aligned} \tag{26}$$

Define the privacy-aware estimator:

$$\widetilde{\boldsymbol{\theta}}_{PV2} = argmin_{\boldsymbol{\theta}\in\Theta}\,\ell_{N,PV_2}(\boldsymbol{\theta}) \tag{27}$$

Theorem 3: Under Assumptions (A1)–(A4) and the additional rate conditions, $\widetilde{\boldsymbol{\theta}}_{PV2}\xrightarrow{P}\boldsymbol{\theta}^*$.

Proof:

The first- and second-order derivatives of the objective function for the privacy-preserving algorithm are given by:

$$\begin{aligned}\nabla^2\ell_{N,PV_2}(\boldsymbol{\theta}) &= \nabla^2\ell_1(\overline{\boldsymbol{\theta}})\\ \nabla\ell_{N,PV_2}(\boldsymbol{\theta}) &= \frac{1}{K}\sum_{j=1}^{K}\left(\nabla\ell_j(\overline{\boldsymbol{\theta}}+\overline{\boldsymbol{\eta}})+\overline{\boldsymbol{\eta}}_j\right)+\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}).\end{aligned} \tag{28}$$

From the stationarity condition $\nabla\ell_{N,PV_2}(\widetilde{\boldsymbol{\theta}}_{PV2})=\mathbf{0}$ together with Equation (20):

$$\begin{aligned}\mathbf{0} = & \quad \nabla \ell_{N,PV_2}(\widetilde{\boldsymbol{\theta}}_{PV2}) \\ = & \quad \tfrac{1}{K}\textstyle\sum_{j=1}^{K}\left(\nabla \ell_j(\overline{\boldsymbol{\theta}}+\overline{\boldsymbol{\eta}})+\overline{\boldsymbol{\eta}}_j\right)+\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\widetilde{\boldsymbol{\theta}}_{PV2}-\widetilde{\boldsymbol{\theta}})+\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\widetilde{\boldsymbol{\theta}}-\overline{\boldsymbol{\theta}}) \\ = & \quad \tfrac{1}{K}\textstyle\sum_{j=1}^{K}\left(\nabla \ell_j(\overline{\boldsymbol{\theta}})+\overline{\boldsymbol{\eta}}\int_0^1\nabla^2\ell_j((1-\rho)\overline{\boldsymbol{\theta}}+\rho(\overline{\boldsymbol{\theta}}+\overline{\boldsymbol{\eta}}))d\rho\right)+\tfrac{1}{K}\textstyle\sum_{j=1}^{K}\overline{\boldsymbol{\eta}}_j \\ & \quad +\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\widetilde{\boldsymbol{\theta}}_{PV2}-\widetilde{\boldsymbol{\theta}})+\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\widetilde{\boldsymbol{\theta}}-\overline{\boldsymbol{\theta}}) \\ = & \quad \tfrac{1}{K}\textstyle\sum_{j=1}^{K}\left(\overline{\boldsymbol{\eta}}\int_0^1\nabla^2\ell_j((1-\rho)\overline{\boldsymbol{\theta}}+\rho(\overline{\boldsymbol{\theta}}+\overline{\boldsymbol{\eta}}))d\rho\right)+\tfrac{1}{K}\textstyle\sum_{j=1}^{K}\overline{\boldsymbol{\eta}}_j+\nabla^2\ell_1(\overline{\boldsymbol{\theta}})(\widetilde{\boldsymbol{\theta}}_{PV2}-\widetilde{\boldsymbol{\theta}})\end{aligned} \tag{29}$$

Therefore,

$$\widetilde{\boldsymbol{\theta}}_{PV_2}-\widetilde{\boldsymbol{\theta}}=-\left[\nabla^2\ell_1(\overline{\boldsymbol{\theta}})\right]^{-1}\left[\tfrac{1}{K}\textstyle\sum_{j=1}^{K}\overline{\boldsymbol{\eta}}_j+\tfrac{1}{K}\textstyle\sum_{j=1}^{K}\overline{\boldsymbol{\eta}}\int_0^1\nabla^2\ell_j((1-\rho)\overline{\boldsymbol{\theta}}+\rho(\overline{\boldsymbol{\theta}}+\overline{\boldsymbol{\eta}}))d\rho\right]. \tag{30}$$

Since $\bar{\theta}$ satisfies the stated initialization condition, and $\frac{1}{K}\sum_{j=1}^{K}\overline{\boldsymbol{\eta}}_j=\frac{1}{N}\sum_{i=1}^{K}\sum_{j=1}^{n}\boldsymbol{\eta}_{ij}$, $\overline{\boldsymbol{\eta}}=\frac{1}{n}\sum_{j=1}^{n}\boldsymbol{\eta}_j$, where ηij and ηj are mutually independent random vectors following zero-mean normal distributions, the stated covariance bounds imply:

$$\left\|\widetilde{\boldsymbol{\theta}}_{PV_2}-\widetilde{\boldsymbol{\theta}}\right\|_2=O_p\left(\tfrac{1}{\sqrt{N}}\right) \tag{31}$$

Under these rate conditions, the one-step bound in [20] gives $\left\|\widetilde{\boldsymbol{\theta}}-\boldsymbol{\theta}^*\right\|_2=O_p\left(\frac{1}{\sqrt{N}}\right)$. Hence,

$$\left\|\widetilde{\boldsymbol{\theta}}_{PV_2}-\boldsymbol{\theta}^*\right\|_2\le\left\|\widetilde{\boldsymbol{\theta}}_{PV_2}-\widetilde{\boldsymbol{\theta}}\right\|_2+\left\|\widetilde{\boldsymbol{\theta}}-\boldsymbol{\theta}^*\right\|_2=O_p\left(\tfrac{1}{\sqrt{N}}\right) \tag{32}$$

Thus, $\widetilde{\boldsymbol{\theta}}_{PV_2}\xrightarrow{P}\boldsymbol{\theta}^*$. This completes the proof.

Under (A1)–(A4) and the additional rate conditions, Theorems 2 and 3 establish consistency and retain the root-N convergence order of the non-private one-step estimator. Scheme 2 protects the initial-parameter release in addition to the gradient release; this broader coverage does not imply a smaller privacy budget or universally better accuracy. The statistical rate comparisons are conditional on the stated covariance bounds and partition-growth regime.

On one hand, the proposed framework can be generalized to other algorithms; on the other hand, alternative noise mechanisms such as Laplace noise and exponential-distributed noise can also be incorporated into the privacy-preserving design.

3. Numerical Examples

In this section, we adopt Monte Carlo simulation to evaluate the finite-sample parameter estimation performance of the aforementioned privacy-preserving approaches. Specifically, we compare the estimation accuracy between conventional algorithms and their privacy-augmented counterparts to quantify how differential privacy affects the precision and stability of parameter estimators. In the final part of this chapter, we further investigate how different choices of privacy budget ε impact the estimation outcomes of privacy-protected algorithms.

We consider a linear regression model, where parameter estimation is implemented via the one-step estimator introduced earlier with privacy protection enforced by the multivariate Gaussian noise mechanism. The independently and identically distributed observations $\{(Y_{ik},X_{ik}),i=1,2,\ldots,n;k=1,2,\ldots,K\}$ are generated from the following data-generating process:

$$Y_{ij}=X_{ij}\boldsymbol{\theta}+e_{ij}, \tag{33}$$

where $e_{ij}\sim\mathcal{N}(0,\sigma_e^2)$, and the baseline estimator is defined as $\overline{\boldsymbol{\theta}}=(X_1^{\top}X_1)^{-1}X_1^{\top}Y_1$. Here, (Yk, Xk) denotes the dataset collected from the k-th machine with (k=1,2,…,K).

Concretely, four estimation approaches are taken into consideration:

(i) Full-sample ordinary least squares (OLS, benchmark): we assume a centralized supercomputer aggregates all available datasets and computes the standard least-squares estimator directly, denoted as $\widehat{\boldsymbol{\theta}}_1=(X^{\top}X)^{-1}X^{\top}Y$.

(ii) Distributed one-step estimation: we implement the aforementioned one-step distributed estimator derived in Equation (13), denoted as $\hat{\theta}_2$; the detailed implementation is summarized in Algorithm 1.

Algorithm 1. One-Step Estimation (Without Privacy Protection)

1. Initialize the starting point $\bar{\theta}$;
2. Broadcast $\bar{\theta}$ to the remaining (K-1) computing nodes;
3. For k = 2,3,…,K do
4. Compute $\nabla \ell_k(\overline{\boldsymbol{\theta}})$
5. Send $\nabla \ell_k(\overline{\boldsymbol{\theta}})$ back to the first computing node
6. End For
7. Compute $\nabla \ell_N(\overline{\boldsymbol{\theta}}) = \frac{1}{K}\sum_{k=1}^{K} \nabla \ell_k(\overline{\boldsymbol{\theta}})$;
8. Compute $\widehat{\boldsymbol{\theta}}_2 = \overline{\boldsymbol{\theta}} - \left[\nabla^2 \ell_1(\overline{\boldsymbol{\theta}})\right]^{-1} \nabla \ell_N(\overline{\boldsymbol{\theta}})$

(iii) Privacy-preserving distributed algorithm: we still adopt the one-step estimation framework, where the multivariate Gaussian privacy-preserving mechanism is imposed on the first-order gradient $\nabla \ell_k(\overline{\boldsymbol{\theta}})$ during data transmission; see Equation (18). The resulting estimator is denoted as $\widehat{\boldsymbol{\theta}}_3$, with detailed implementation outlined in Algorithm 2.

Algorithm 2. One-Step Estimation with Privacy Protection on First-Order Gradients

1. Initialize the starting point $\bar{\theta}$;
2. Broadcast $\bar{\theta}$ to the remaining (K-1) computing nodes;
3. For k = 2,3,…,K do
4. Compute $\nabla \ell_k(\overline{\boldsymbol{\theta}})$
5. Send $\nabla \ell_k(\overline{\boldsymbol{\theta}}) + \frac{1}{n}\sum_{j=1}^{n} \boldsymbol{\eta}_{jk}$ back to the first computing node
6. End For
7. Compute $\nabla \ell_N(\overline{\boldsymbol{\theta}}) = \frac{1}{K}\nabla \ell_1(\overline{\boldsymbol{\theta}}) + \frac{1}{K}\sum_{k=2}^{K}\left[\nabla \ell_k(\overline{\boldsymbol{\theta}}) + \frac{1}{n}\sum_{j=1}^{n} \boldsymbol{\eta}_{jk}\right]$;
8. Compute $\widehat{\boldsymbol{\theta}}_3 = \overline{\boldsymbol{\theta}} - \left[\nabla^2 \ell_1(\overline{\boldsymbol{\theta}})\right]^{-1} \nabla \ell_N(\overline{\boldsymbol{\theta}})$

(iv) We consider the distributed algorithm where the multivariate Gaussian privacy-preserving mechanism is imposed on both the first-order gradient $\nabla \ell_k(\overline{\boldsymbol{\theta}})$ and the initial parameter $\bar{\theta}$ during data transmission; see Equation (27). The corresponding estimator is denoted as $\widehat{\boldsymbol{\theta}}_4$, with detailed implementation summarized in Algorithm 3.

Algorithm 3. One-Step Estimation with Double-Layer Privacy Protection

1. Initialize the starting point $\bar{\theta}$;
2. Broadcast $\overline{\boldsymbol{\theta}} + \frac{1}{n}\sum_{j=i}^{n} \boldsymbol{\eta}_j$ to the remaining (K-1) computing nodes;
3. For k = 2,3,…,K do
4. Compute $\nabla \ell_k(\overline{\boldsymbol{\theta}})$
5. Send $\nabla \ell_k\left(\overline{\boldsymbol{\theta}} + \frac{1}{n}\sum_{j=i}^{n} \boldsymbol{\eta}_j\right)$ back to the first computing node
6. End For
7. Compute $\nabla \ell_N(\overline{\boldsymbol{\theta}}) = \frac{1}{K}\nabla \ell_1(\overline{\boldsymbol{\theta}}) + \frac{1}{K}\sum_{k=2}^{K}\left[\nabla \ell_k\left(\overline{\boldsymbol{\theta}} + \frac{1}{n}\sum_{j=i}^{n} \boldsymbol{\eta}_j\right) + \frac{1}{n}\sum_{j=1}^{n} \boldsymbol{\eta}_{jk}\right]$;
8. Compute $\widehat{\boldsymbol{\theta}}_4 = \overline{\boldsymbol{\theta}} - \left[\nabla^2 \ell_1(\overline{\boldsymbol{\theta}})\right]^{-1} \nabla \ell_N(\overline{\boldsymbol{\theta}})$

We next investigate the impact of privacy-preserving algorithms on parameter estimation. We fix the per-machine sample size n=64 and vary the number of computing nodes K from 100 to 5000; we then fix K=100 and vary n ranging from 100 to 10000. The main experiments use ε=0.5 and δ=10-4 as prespecified settings consistent with the illustrative values in [29]. These are experimental choices, rather than a universal parameter recommendation.

All numerical results are averaged over 10000 independent repeated trials. Figures 3 and 4 visualize the estimation performance with varying K and n, respectively. The black solid lines correspond to the benchmark estimator obtained by a centralized supercomputer using the full dataset (i.e., the optimal estimate from classical centralized statistical methods); the red lines represent the communication-efficient distributed algorithm without privacy protection; the blue and green lines denote the estimators from the two differential privacy-preserving schemes, respectively. Subplots (a) and (b) in each figure plot the empirical mean of the estimated parameters for the first and second coordinate of θ, respectively.

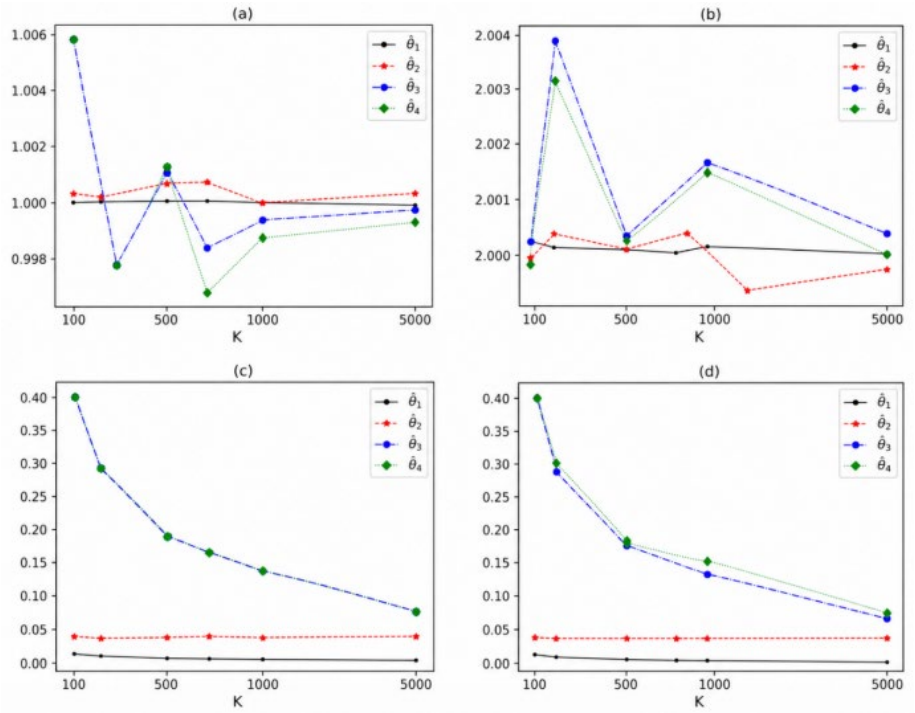


Figure 3. Parameter estimation results of the four methods with n=64
Subplot (a) shows the empirical mean of estimates for the first parameter dimension; (b) corresponds to the empirical mean for the second dimension; (c) reports the empirical standard deviation of estimates for the first dimension; and (d) gives the empirical standard deviation for the second dimension

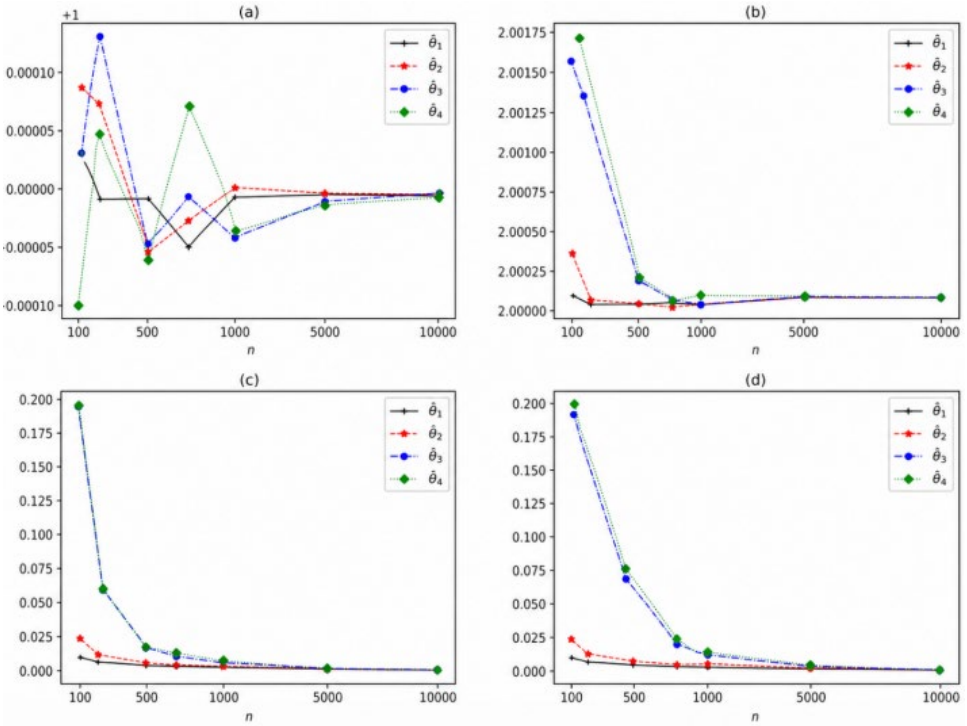


Figure 4. Parameter estimation results of the four methods with K=100
Subplot (a) shows the empirical mean of estimates for the first parameter dimension; (b) corresponds to the empirical mean for the second dimension; (c) reports the empirical standard deviation of estimates for the first dimension; and (d) gives the empirical standard deviation for the second dimension

3.1. Ablation analysis of scheme 1 and scheme 2

As shown in Figures 3 and 4, all four estimation methods converge toward the true parameter values 1 and 2 as K and n grow, which provides finite-sample evidence of decreasing estimation bias under the tested privacy settings; it does not by itself prove asymptotic unbiasedness. Subplots (c) and (d) report the empirical standard deviation of the two-dimensional parameter estimates. From Figures 3(c) and 3(d), we observe that for small per-node sample sizes, the standard deviations of the privacy-regularized estimators fail to converge to those of the non-private baseline even with an extremely large number of partitioned computing nodes. By contrast, Figures 4(c) and 4(d) demonstrate that the standard deviations of estimators from Method 3 (Scheme 1, Algorithm 2) and Method 4 (Scheme 2, Algorithm 3) gradually converge to the non-private results of Method 1 and Method 2 as the overall sample size increases.

By examining the blue and green curves in subplots (c) and (d), we observe that the two privacy-preserving schemes impose nearly identical impacts on the robustness of parameter estimation. Specifically, within the communication-efficient distributed framework, the performance degradation induced by privatizing the parameter vector is markedly milder than that caused by privatizing the gradient vector, leading to similar empirical performance between the two privacy-protected algorithms. Under the additional covariance conditions, the parameter perturbation in Equation (30) has an effect of order $O\left(\frac{1}{\sqrt{N}}\right)$ on the final parameter estimator. In contrast, per Equation (22), directly adding privacy noise to the first-order gradient introduces an estimation error of order $O\left(\frac{1}{\sqrt{N}}\right)$. These bounds give the same asymptotic order and do not imply that the parameter perturbation is negligible. The close performance of $\widehat{\boldsymbol{\theta}}_3$and $\widehat{\boldsymbol{\theta}}_4$ is an observation for the tested configurations.

The comparison in Figures 3 and 4 also serves as a controlled ablation of the two privacy layers. With n=64 and K=100, the standard deviations of Scheme 1 and Scheme 2 are both approximately 0.40 for the two parameter coordinates; when K increases to 5000, they decrease to about 0.07–0.08 and remain nearly overlapped. With K=100 and n increasing from 100 to 10000, the corresponding standard deviations decrease from approximately 0.19–0.20 to values close to the non-private curves. Across both node-number and local-sample-size settings, Scheme 2 introduces only a small additional fluctuation relative to Scheme 1, showing that the extra parameter-layer perturbation strengthens privacy coverage without materially changing the finite-sample convergence trend.

3.2. Hyperparameter sensitivity analysis

In addition, we further investigate how the privacy budget ε affects parameter estimation under privacy constraints. We fix the number of computing nodes K=100 and total sample size N=5000, retain the same model setup as before, and set the privacy parameter δ=10-4. All numerical results are averaged over 10000 independent repeated trials and plotted in Figure 5. Subplots (a) and (b) depict the absolute value of estimation bias for the first and second dimension of parameters under the first privacy-preserving scheme, while subplots (c) and (d) illustrate the absolute estimation bias of the two parameter dimensions for the second privacy scheme. Consistent with theoretical expectations, as ε increases (i.e., the enforced privacy protection strength weakens), the privatized parameter estimators become increasingly accurate with shrinking estimation bias. This confirms an inherent tradeoff between privacy protection level and estimation accuracy; practitioners need to strike a proper balance between these two competing objectives according to practical application requirements.

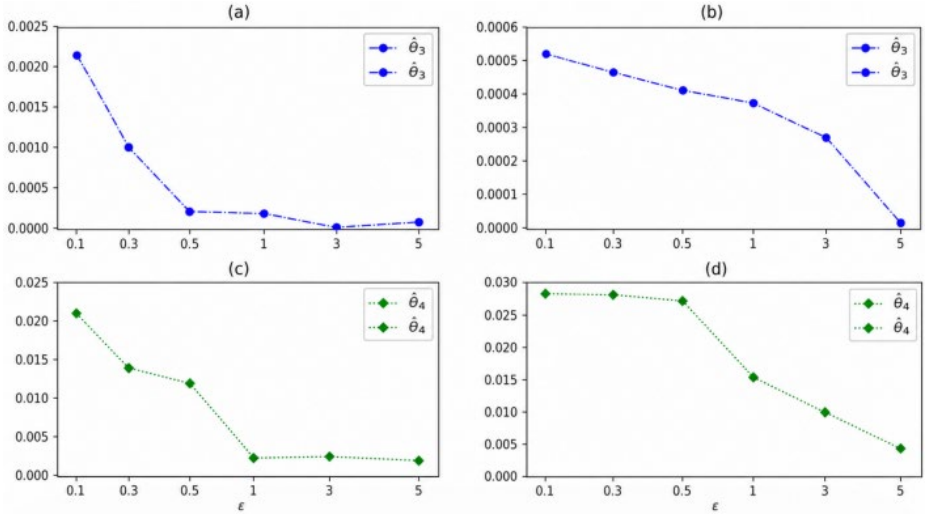


Figure 5. Privacy-preserving parameter estimation results with K=100 and N=5000

Subplot (a) shows the absolute estimation bias of the first parameter dimension under the first privacy scheme; (b) denotes the absolute estimation bias of the second dimension for the first privacy scheme; (c) presents the absolute estimation bias of the first dimension under the second privacy scheme; and (d) gives the absolute estimation bias of the second dimension for the second privacy scheme

Figure 5 quantitatively confirms the privacy–accuracy tradeoff. For Scheme 1, the absolute bias of the first parameter dimension decreases from approximately 2.1×10−3 at ε=0.1 to about 2.0×10−4 at ε=0.5 and becomes nearly zero when ε reaches 3–5; the second-dimension bias declines from about 5.2×10−4 at ε=0.1 to almost zero at ε=5. For Scheme 2, the first-dimension bias falls from approximately 2.1×10−2 to 1.5×10−3 and the second-dimension bias from about 2.8×10−2 to 4.5×10−3 over the same ε range. The bias is therefore most sensitive in the strong-privacy region (small ε), whereas the marginal accuracy gain becomes smaller as ε increases.

The covariance dimension p also controls the perturbation strength because the total budget is equally divided among coordinates. For example, at ε=0.5, p=2, 4, 8, and 16 correspond to per-coordinate budgets ε/p of 0.25, 0.125, 0.0625, and 0.03125, respectively. Thus, doubling p halves the budget available to each coordinate; according to Equation (3), a smaller ε/p requires a larger admissible noise variance σ_j^2. Consequently, the total perturbation energy $\mathbb{E}\|\boldsymbol{\eta}\|^2 = \sum_j \sigma_j^2$ increases with dimension under equal allocation, which explains why very high-dimensional transmitted vectors can suffer a less favorable privacy–accuracy tradeoff.

4. Empirical Analysis

In this chapter, we apply our proposed method to a practical big-data scenario using data from the U.S. Airline Database; detailed dataset descriptions are available in Reference [30], with raw data hosted at the URL: http://stat-computing.org/dataexpo/2009. This dataset contains complete flight records for all U.S. airlines spanning 1987 to 2008. Each record corresponds to an individual flight, documenting flight delay status, actual and scheduled takeoff/landing times, flight distance and duration, operation date, as well as origin and destination airports.

Reference [30] uses the U.S. Airline dataset as a large-scale distributed regression benchmark and reports a logistic-regression analysis of flight-delay status with a high-dimensional set of continuous and dummy-coded categorical predictors. The purpose of citing [30] here is therefore to establish the dataset source and distributed-computing context, rather than to reproduce its specific model specification. Because the present method is developed for low-dimensional interpretable parameter estimation, we construct a simplified working model using four continuous covariates available in the same dataset—actual departure time, scheduled departure time, actual flight duration, and flight distance. Categorical variables such as carrier, origin, destination, month, and day-of-week, which are represented by dummy variables in [30], are intentionally excluded from our working model. This simplification is a methodological choice of the present study and should not be interpreted as a conclusion drawn from [30] that those variables are redundant or insignificant.

Our empirical objective is to evaluate the proposed privacy-preserving estimator on a transparent low-dimensional regression task built from the Airline dataset. Accordingly, we specify the following linear working model using the four continuous predictors defined above:

$$Y_j = X_{1j}\theta_1 + X_{2j}\theta_2 + X_{3j}\theta_3 + X_{4j}\theta_4 + \eta + e_j \quad (34)$$

Unlike the logistic-regression setup in Reference [30], we define the continuous response Y as raw flight delay time: positive values denote arrival delays, and negative values correspond to early arrivals. We retain four core predictors consistent with prior research: X1: actual departure time; X2: scheduled departure time; X3: actual flight duration; X4: flight distance. No centering preprocessing is applied to raw data, so an intercept term η is explicitly included in the model formulation. We emphasize that this linear specification serves merely as a working model rather than a full representation of real-world airline operational mechanisms. Our two primary research goals for this empirical setup are as follows:

First, estimation outcomes derived from this working model should exhibit stable and interpretable relationships among the selected continuous variables, allowing us to assess the internal plausibility of the simplified specification.

Second, after injecting noise for differential privacy protection, the resulting parameter estimates should remain closely aligned with non-private benchmark results, verifying that our privacy-augmented algorithm preserves satisfactory estimation accuracy.

Owing to the massive raw dataset size, we randomly sample 1.2×108 records for practical empirical analysis. To match the design of our distributed algorithm, these sampled records are randomly split into K=6000 disjoint groups, where subscript j indexes the group affiliation of each observation, and each group contains n=20000 samples. Across all groups, the random error term follows a homogeneous normal distribution: $e_j \sim \mathcal{N}(0, \sigma_e^2)$.

The sampling and partitioning procedure is as follows. Records with missing or invalid values in the selected response/predictor fields are removed first, after which only the variables used in Equation (34) are retained. The 1.2×10^8 observations are then sampled uniformly without replacement from the usable records, randomly permuted, and partitioned into 6000 mutually disjoint groups of 20000 observations each. The identical sampled dataset

and partition are used by Methods 1–4 so that the reported differences are attributable to distributed computation and privacy perturbation rather than to different resampling realizations.

In numerical implementation, we adopt the four estimation approaches introduced in the simulation study of section 3 for model parameter inference, defined as follows: Method 1: Centralized direct estimation; Method 2: Privacy-free distributed estimation; Method 3: Distributed estimation with gradient-domain differential privacy; Method 4: Distributed estimation under dual-layer privacy protection.

For the two privacy-preserving schemes (Method 3 and Method 4), we retain identical privacy hyperparameters as set in Chapter 3's numerical experiments, i.e., εn=0.5, δ=0.0001. We thereby obtain estimators $\hat{\theta}_1, \hat{\theta}_2, \hat{\theta}_3, \hat{\theta}_4, \hat{\eta}$ across all four methods, with numerical outcomes summarized in Table 1.

Table 1. Estimation results of regression modeling on U.S. airline dataset using non-private Methods (1), (2) and privacy-preserving Methods (3), (4)

Estimation Method	Actual Departure Time $\hat{\theta}_1$	Scheduled Departure Time $\hat{\theta}_2$	Actual Flight Duration $\hat{\theta}_3$	Flight Distance $\hat{\theta}_4$	Intercept $\hat{\eta}$
Method 1	0.03486	−0.02629	0.6108	−0.07321	−26.8023
Method 2	0.03451	−0.02598	0.6107	−0.07317	−26.7511
Method 3	0.03453	−0.02599	0.6103	−0.07313	−26.7347
Method 4	0.03422	−0.02520	0.6107	−0.07329	−26.7387

Because our working model differs fundamentally from the logistic-regression specification in Reference [30], the coefficient signs in Table 1 are interpreted only within the present continuous-response linear model and are not claimed to reproduce the coefficients or substantive conclusions of [30]. Within this working model, $\hat{\theta}1$ and $\hat{\theta}2$ have comparable magnitudes but opposite signs. Since X1−X2 represents the difference between actual and scheduled departure times, the combined effect of X1 and X2 should be interpreted jointly rather than as direct evidence of effects reported in prior airline studies. The estimates of $\hat{\theta}3$ and $\hat{\theta}4$ also have opposite signs, indicating that actual flight duration and flight distance have different conditional associations with the continuous delay response after controlling for the other selected variables. Because flight duration and distance are strongly related operational quantities, these marginal coefficients should be interpreted cautiously and do not imply causal effects. The main purpose of Table 1 is therefore to compare the stability of the four estimation methods under an identical working model, rather than to draw substantive conclusions about airline-delay mechanisms.

From the privacy protection perspective, all four approaches yield fairly consistent estimation results. Although Methods 3 and 4 inject additive noise into transmitted information to satisfy differential privacy constraints, their resultant estimators remain reasonably accurate compared with the non-private centralized and distributed benchmarks, with relative deviations hovering around 2%. Method 4, equipped with extra noise perturbation for dual privacy protection, deviates slightly more from the fully non-private baseline than Method 3 (which only adds noise on gradient values). Nevertheless, the performance gap between Method 4 and Method 2 stays at a comparable magnitude to that between Method 1 and Method 2. This demonstrates that the accuracy degradation incurred by the proposed privacy mechanisms is on par with the precision loss originating from distributed partitioning for massive datasets. Therefore, the presented privacy-preserving scheme is practically acceptable for distributed real-data applications, fulfilling the core requirement of safeguarding privacy without severely impairing estimation efficiency.

In addition, we further evaluate the four methods in terms of computational efficiency and memory footprint over large-scale data by comparing their computational overhead on identical datasets. For Method 1 (centralized computation on a single machine without privacy), the total runtime is 19.61 seconds. When the workload is split across K=6000 local worker nodes under Method 2, the per-machine runtime drops to roughly 0.0021 seconds. For the privacy-enhanced Methods 3 and 4, their per-node computation costs are close, at 0.0118 s and 0.0123 s respectively. This outcome is reasonable: despite Method 4 adopting double-layer noise encryption, its first round of privatization is executed on the master server, leading to nearly identical runtime overhead on worker machines as Method 3.

By comparing privacy-free and privacy-augmented algorithms, we observe that the computational efficiency of privacy-preserving schemes does not degrade to an intolerable level relative to Method 2, the privacy-free distributed approach with optimal runtime performance. It is worth noting that the extra runtime overhead of privacy-enabled algorithms mainly stems from on-the-fly generation of random noise required for privacy

guarantees. Unlike simulation environments, real-world noise generation and encryption can be offloaded to dedicated hardware on local servers instead of being fully processed by worker nodes. This avoids consuming computing resources of worker machines and only incurs modest overhead for data transmission and validation. Moreover, as the number of local worker nodes (i.e., split subtasks) rises, the per-node cost for random noise generation further declines; our empirical setup deploys K=6000 worker nodes for collaborative computation, making the incremental computational overhead entirely acceptable.

We next discuss memory consumption. The sample used for empirical evaluation is randomly extracted from the full raw dataset; the cleaned subset occupies approximately 2 GB of disk storage, while the local raw-data representation used in our implementation occupies approximately 12 GB. Around 18 GB of runtime memory is required to process the sampled subset. These figures are specific to our preprocessing and storage format and are not directly comparable with the 52 GB airline-data representation reported in Reference [30]. Nevertheless, both settings illustrate that full-scale airline analysis can impose substantial memory demands on a single machine. This observation further motivates distributed computation for large-scale real-world datasets.

5. Conclusion

This paper develops Gaussian-noise mechanisms for protecting gradient transmission and initial-parameter transmission in communication-efficient distributed parameter estimation. The framework retains the interpretability of the underlying regression model. Under the stated regularity, initialization, partition-growth and noise-scale conditions, the resulting estimators are consistent and retain the root-N convergence order of the non-private one-step estimator. Simulations and the airline-data case study demonstrate a privacy-accuracy tradeoff and limited additional estimation error in the configurations evaluated. These empirical findings support the practical use of the framework within its stated modeling and privacy-accounting assumptions.

The present study also has several limitations. First, the equal allocation ε/p becomes conservative as the parameter or gradient dimension grows; the resulting increase in required noise covariance can degrade finite-sample accuracy for extremely high-dimensional vectors. Second, the current theory uses multivariate Gaussian perturbation with a diagonal covariance structure and a bounded sensitivity domain, while the empirical study focuses on a low-dimensional linear working model. Future research will investigate adaptive and non-uniform privacy-budget allocation, structured or data-independent covariance design, compression-aware privacy accounting, and high-dimensional sparse or nonlinear distributed models, as well as alternative mechanisms such as Laplace-based perturbation.

References

[1] Sweeney L. K-anonymity: a model for protecting privacy. Int J Uncertain Fuzziness Knowl Based Syst. 2002;10(5):557-70. doi:10.1142/S0218488502001648.

[2] de Montjoye YA, Radaelli L, Singh VK, Pentland AS. Unique in the shopping mall: on the reidentifiability of credit card metadata. Science. 2015;347(6221):536-9. doi:10.1126/science.1256297.

[3] Xue J, Yan W. Edge computing communication privacy protection method based on federated learning algorithm. EAI Endorsed Trans Scalable Inf Syst. 2026;12(9). doi:10.4108/eetsis.12254.

[4] Guo Z, Sun Y, Zhang X, Wu L. An efficient privacy-preserving secure aggregation scheme for federated learning with input verification and dropout resistance. EAI Endorsed Trans Scalable Inf Syst. 2026;12(8). doi:10.4108/eetsis.11991.

[5] Sun Y, Cao S. Optimization of urban traffic signal control system on Hadoop platform based on privacy computing. EAI Endorsed Trans Scalable Inf Syst [Internet]. 2026 May 11 [cited 2026 Nov 18];12(10). Available from: https://eudl.eu/doi/10.4108/eetsis.11814.

[6] Li H. A cloud environment security access control scheme based on federated learning and fuzzy logic integration. EAI Endorsed Trans Scalable Inf Syst. 2026;12(9). doi:10.4108/eetsis.11731.

[7] Wang J, Su J. A review of object detection techniques in IoT-based intelligent transportation systems. Comput Mater Contin. 2025;84(1):125-52. doi:10.32604/cmc.2025.064309.

[8] Su J, Dong H, Xia D, Liu AX, Wang B. A zero trust method for tag array authentication of UHF RFID sensing. IEEE Trans Netw. 2026;34:427-41. doi:10.1109/TON.2025.3599016.

[9] Jin Z, Zhang X, Su J, Zhang L, Shen J. Subgraph-driven lightweight federated learning for spatiotemporal cellular traffic prediction. IEEE Trans Netw Serv Manag. 2026;23:1435-48. doi:10.1109/TNSM.2025.3645253.

[10] Jin Z, Yang C, Ye Y, Zhang L, Shen J, Su J. Mobility-aware semi-asynchronous federated learning for vehicular networks. IEEE Trans Veh Technol. 2026;75(2):2001-12. doi:10.1109/TVT.2025.3603978.

[11] Dwork C, McSherry F, Nissim K, Smith A. Calibrating noise to sensitivity in private data analysis. In: Halevi S, Rabin T, editors. Theory of cryptography. Proceedings of the Third Theory of Cryptography Conference (TCC 2006); 2006 Mar 4-7; New York, NY, USA. Berlin: Springer; 2006. p. 265-84. (Lecture Notes in Computer Science; vol. 3876). doi:10.1007/11681878_14.

[12] Dwork C, Kenthapadi K, McSherry F, Mironov I, Naor M. Our data, ourselves: privacy via distributed noise generation. In: Vaudenay S, editor. Advances in cryptology - EUROCRYPT 2006. Proceedings of the 25th International Conference on the Theory and Applications of Cryptographic Techniques; 2006 May 28-Jun 1; St. Petersburg, Russia. Berlin: Springer; 2006. p. 486-503. (Lecture Notes in Computer Science; vol. 4004). doi:10.1007/11761679_29.

[13] Dwork C, Roth A. The algorithmic foundations of differential privacy. Found Trends Theor Comput Sci. 2014;9(3-4):211-407. doi:10.1561/0400000042.

[14] Dinur I, Nissim K. Revealing information while preserving privacy. In: Proceedings of the Twenty-Second ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS '03); 2003 Jun 9-12; San Diego, CA, USA. New York (NY): Association for Computing Machinery; 2003. p. 202-10. doi:10.1145/773153.773173.

[15] Dwork C, McSherry F, Talwar K. The price of privacy and the limits of LP decoding. In: Proceedings of the 39th Annual ACM Symposium on Theory of Computing (STOC '07); 2007 Jun 11-13; San Diego, CA, USA. New York (NY): Association for Computing Machinery; 2007. p. 85-94. doi:10.1145/1250790.1250804.

[16] Dwork C, Yekhanin S. New efficient attacks on statistical disclosure control mechanisms. In: Wagner D, editor. Advances in cryptology - CRYPTO 2008. Proceedings of the 28th Annual International Cryptology Conference; 2008 Aug 17-21; Santa Barbara, CA, USA. Berlin: Springer; 2008. p. 469-80. (Lecture Notes in Computer Science; vol. 5157). doi:10.1007/978-3-540-85174-5_26.

[17] Chan THH, Shi E, Song D. Optimal lower bound for differentially private multi-party aggregation. In: Epstein L, Ferragina P, editors. Algorithms - ESA 2012. Proceedings of the 20th Annual European Symposium on Algorithms; 2012 Sep 10-12; Ljubljana, Slovenia. Berlin: Springer; 2012. p. 277-88. (Lecture Notes in Computer Science; vol. 7501). doi:10.1007/978-3-642-33090-2_25.

[18] Sabater C, Bellet A, Ramon J. An accurate, scalable and verifiable protocol for federated differentially private averaging. Mach Learn. 2022;111(11):4249-93. doi:10.1007/s10994-022-06267-9.

[19] Chen WN, Kairouz P, Özgür A. Breaking the communication-privacy-accuracy trilemma. IEEE Trans Inf Theory. 2023;69(2):1261-81. doi:10.1109/TIT.2022.3218772.

[20] Jordan MI, Lee JD, Yang Y. Communication-efficient distributed statistical inference. J Am Stat Assoc. 2019;114(526):668-81. doi:10.1080/01621459.2018.1429274.

[21] Fan J, Guo Y, Wang K. Communication-efficient accurate statistical estimation. J Am Stat Assoc. 2023;118(542):1000-10. doi:10.1080/01621459.2021.1969238.

[22] Zhou X, Chang L, Xu P, Lv S. Communication-efficient and Byzantine-robust distributed learning with statistical guarantee. Pattern Recognit. 2023;137:109312. doi:10.1016/j.patcog.2023.109312.

[23] Fu J, Hong Y, Ling X, Wang L, Ran X, Sun Z, et al. Differentially private federated learning: a systematic review. ACM Comput Surv. 2026;58(11):271. doi:10.1145/3801079.

[24] Hasırcıoğlu B, Gündüz D. Communication efficient private federated learning using dithering. In: ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 2024 Apr 14-19; Seoul, Republic of Korea. Piscataway (NJ): IEEE; 2024. p. 7575-9. doi:10.1109/ICASSP48485.2024.10446222.

[25] Zhang M, Xie Z, Yin L. Private and communication-efficient federated learning based on differentially private sketches. arXiv [Preprint]. 2024 Oct 8 [cited 2026 Nov 18]. arXiv:2410.05733. Available from: https://arxiv.org/abs/2410.05733.

[26] Li Y, Du W, Han L, Zhang Z, Liu T. A communication-efficient, privacy-preserving federated learning algorithm based on two-stage gradient pruning and differentiated differential privacy. Sensors. 2023;23(23):9305. doi:10.3390/s23239305.

[27] Li M, Xiao D, Liang J, Huang H. Communication-efficient and Byzantine-robust differentially private federated learning. IEEE Commun Lett. 2022;26(8):1725-9. doi:10.1109/LCOMM.2022.3180113.

[28] Carletti V, Foggia P, Mazzocca C, Parrella G, Vento M. SoK: gradient inversion attacks in federated learning. In: Bauer L, Pellegrino G, editors. Proceedings of the 34th USENIX Security Symposium (USENIX Security 25) [Internet]; 2025 Aug 13-15; Seattle, WA, USA. Berkeley (CA): USENIX Association; 2025 [cited 2026 Nov 18]. p. 6439-59. Available from: https://www.usenix.org/conference/usenixsecurity25/presentation/carletti

[29] Wang J, Liu S, Li Y. A review of differential privacy in individual data release. Int J Distrib Sens Netw. 2015;11(10):259682. doi:10.1155/2015/259682.

[30] Zhu X, Li F, Wang H. Least-square approximation for a distributed system. J Comput Graph Stat. 2021;30(4):1004-18. doi:10.1080/10618600.2021.1923517.