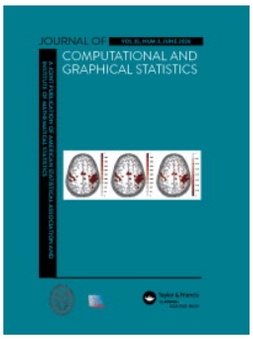

Taylor & Francis
Taylor & Francis Group

Journal of Computational and Graphical Statistics

ISSN: 1061-8600 (Print) 1537-2715 (Online) Journal homepage: www.tandfonline.com/journals/ucgs20

Communication-Efficient Estimation for Non-randomly Distributed and Missing Data

Xirui Liu, Mixia Wu, Ke Yang & Bangshu Liu

To cite this article: Xirui Liu, Mixia Wu, Ke Yang & Bangshu Liu (03 Jun 2026): Communication-Efficient Estimation for Non-randomly Distributed and Missing Data, Journal of Computational and Graphical Statistics, DOI: 10.1080/10618600.2026.2648595

To link to this article: https://doi.org/10.1080/10618600.2026.2648595

JOURNAL OF COMPUTATIONAL AND GRAPHICAL STATISTICS
https://doi.org/10.1080/10618600.2026.2648595

Communication-Efficient Estimation for Non-randomly Distributed and Missing Data

Xirui Liu[a], Mixia Wu[b], Ke Yang[b], and Bangshu Liu[c]

[a]School of Mathematical Sciences, Guizhou Normal University, Guiyang, China; [b]School of Mathematics, Statistics and Mechanics, Beijing University of Technology, Beijing, China; [c]College of Economics and Management, Beijing University of Technology, Beijing, China

ABSTRACT

Traditional distributed methods rely on two fundamental assumptions: completeness, which assumes that datasets across local machines be fully observed, and randomness, which stipulates that data be randomly distributed across these machines. Violating these assumptions can significantly impair the statistical efficiency of distributed estimators or even render them inconsistent. In this paper, we focus on distributed estimation for data that is non-randomly and non-uniformly distributed, and contains missing values. Depending on whether the local Hessian matrix can be transformed, we propose two Communication-efficient Pilot One-step Update (CPOU) methods. These methods integrate pilot sampling to guarantee estimator consistency, inverse probability weighting (IPW) to reduce bias arising from missing data, and one-step updating to ensure that the efficiency of the proposed estimators is comparable to that of the global estimator. Theoretical analysis and empirical studies demonstrate the good performance of the proposed methods. Supplementary materials for this article are available online.

ARTICLE HISTORY
Received November 2024
Accepted March 2026

KEYWORDS
Distributed algorithm; Inverse probability weighting; Non-randomly distributed; One-step update; Pilot sample

1. Introduction

Over the past two decades, distributed statistical inference has become a prominent research area due to its benefits in efficient computation and system scalability (Damiani et al. 2015; Guo et al. 2020; Cadena et al. 2021; Fan, Guo, and Wang 2023). Unlike the traditional centralized methods, distributed statistical learning conducts estimation tasks across multiple machines. In a typical distributed learning setup, the data is divided and stored across K local machines, each of which is connected to a central machine. The central machine coordinates the aggregation of local computations and facilitates communication between the local machines to achieve a global estimation. Recently, numerous distributed statistical methods are developed, which can be divided into two classes, namely one-shot (OS) and multi-round methods. OS methods require only a single round of communication. Specifically, the local machines operate in parallel and send their results to the central machine. Then the central machine aggregates the results into a final outcome (Lin and Xi 2011; Zhang, Wainwright, and Duchi 2012; Liu and Ihler 2014; Huang and Huo 2019; Zhu, Li, and Wang 2021). While the OS strategies are communication-efficient, they require a sufficiently large sample size on each local machine. In contrast, multi-round approaches, which alternate between local computations and global aggregations, remove the restriction on local sample size but need higher communication costs than OS methods (Shamir, Srebro, and Zhang 2014; Jordan, Lee, and Yang 2019; Fan, Guo, and Wang 2023).

Most existing distributed methods rely on the assumption that data are randomly distributed across local machines, which is often unrealistic in practice. In many applications, data are partitioned by time or location, leading to distributional heterogeneity. To address this issue, Wang et al. (2021) proposed the one-step upgraded pilot (OSUP) method, which first obtains a pilot estimate from a random subsample and then refines it via a one-step update to achieve statistical efficiency. This framework has been further extended to quantile regression (Pan et al. 2022), modal regression (Li, Wang, and Xu 2023), and penalized regression (Wang and Li 2023).

The aforementioned works all assume that data are fully observed, but missing data is common across various fields, such as surveys, clinical trials, and longitudinal studies. Complete case (CC) analysis, which excludes all observations with missing data, can lead to biased and inefficient estimators (Little and Rubin 1987; Robins, Rotnitzky, and Zhao 1994). Inverse probability weighting (IPW) is widely used to deal with missing data. For instance, Shi et al. (2021) developed a IPW-based distributed M-estimation method for randomly distributed dataset with missing values. Similarly, Pan et al. (2025) proposed a distributed approach for linear regression with covariates missing at random. In scenarios involving distributed missing response data with high-dimensional covariates, Zhang, Zhang, and Wang (2023) introduced a distributed sparse M-estimator based on IPW strategy.

To the best of our knowledge, there has been little research on non-randomly distributed data with missing values in the current literature. Xiong et al. (2023) tackle causal inference in federated settings with inherent population heterogeneity. This paper primarily focuses on data that was partitioned for non-statistical reasons (e.g., storage constraints), which may not reflect underlying population heterogeneity. Our goal is to achieve a statistically efficient estimate from such a biased storage layout, not necessarily to model site-specific causal mech-

CONTACT Mixia Wu *wumixia@bjut.edu.cn* School of Mathematics, Statistics and Mechanics, Beijing University of Technology, Beijing, China.
Supplementary materials for this article are available online. Please go to www.tandfonline.com/r/JCGS.

anisms. We proposes two Communication-Efficient Pilot One-Step Update (CPOU) methods to accommodate varying levels of communication constraints. The proposed methods involve three key steps: Firstly, we use IPW to weight the loss function, correcting the bias introduced by missing data. Secondly, pilot estimator is obtained using the weighted loss function and a pilot sample randomly selected from each local machine. The pilot estimator is expected to be statistically consistent because of the random sampling nature of the pilot sample. Finally, we refine the pilot estimator through two distinct one-step updates. Compared with the SA algorithm proposed by Shi et al. (2021), our methods accommodate non-randomly distributed data and necessitates only three rounds of communication, thereby eliminating the need for multiple iterations. Furthermore, theoretical results demonstrate that the proposed estimators are statistically as efficient as the global estimator, which is derived using the pooled data from all local machines. Their effectiveness is demonstrated through numerical simulations and a real-world data application.

The rest of this paper is organized as follows. Section 2 discusses the distributed statistical models under non-randomness and missingness. Section 3 introduces our new methods. Theoretical properties of the proposed CPOU estimators are shown in Section 4. In Section 5, we provide numerical simulation analysis to validate the finite sample performance of the proposed methods. A real-world dataset example is given in Section 6. Section 7 concludes the paper.

2. Problem Setup

In this section, we introduce the distributed estimation problem with non-random distributed and missing data.

Let $\mathbf{Z} \in \mathcal{Z}$ be a random variable vector from an unknown probability distribution $\mathbb{P}$. Assume that our parameter of interest $\boldsymbol{\theta}^* \in \Theta \subset \mathbb{R}^p$ is the unique minimizer of the population loss function $F(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{Z}\mathbb{P}}[\ell(\mathbf{Z};\boldsymbol{\theta})]$, where ℓ is a loss function: $\mathcal{Z} \times \Theta \to \mathbb{R}$. In many estimation problems, ℓ is chosen as the negative log-likelihood function. Given i.i.d. observations $\{\mathbf{Z}_i\}_{i=1}^N$ from $\mathbb{P}$, a natural estimator of $\boldsymbol{\theta}^*$ is the minimizer of the empirical loss function

$$f(\boldsymbol{\theta}) = \frac{1}{N}\sum_{i=1}^{N} \ell(\mathbf{Z}_i;\boldsymbol{\theta}). \tag{1}$$

In practice, missing data is common: only part of $\mathbf{Z}$ is consistently observed. Write $\mathbf{Z} = (\mathbf{Z}^o, \mathbf{Z}^m)$, where $\mathbf{Z}^o$ is always observed and $\mathbf{Z}^m$ may be missing. In addition, auxiliary variables $\mathbf{T} \in \mathcal{T}$ related to the missing mechanism may be available. For convenience, define $\mathbf{W} = (\mathbf{Z}^o, \mathbf{T})$ as the fully observed covariates.

We assume the missing mechanism is missing at random (MAR). Let δ be the indicator of observation ($\delta = 1$ if $\mathbf{Z}^m$ is observed, otherwise $\delta = 0$). Under MAR assumption,

$$\mathbb{P}(\delta = 1|\mathbf{Z}, \mathbf{T}) = \mathbb{P}(\delta = 1|\mathbf{Z}^o, \mathbf{T}) \doteq \pi(\mathbf{W};\boldsymbol{\gamma}),$$

where $\pi : \mathcal{W} \times \Gamma \to [0,1]$ is known as the propensity score function, parameterized by $\boldsymbol{\gamma} \in \Gamma \subset \mathbb{R}^d$. Since the true propensity score function is unknown, it is commonly estimated by logistic regression (Robins and Rotnitzky 1995),

$$\pi(\mathbf{W};\boldsymbol{\gamma}) = \frac{\exp\{\gamma^\top \mathbf{W}\}}{1+\exp\{\gamma^\top \mathbf{W}\}}.$$

The nuisance parameter $\boldsymbol{\gamma}^*$ is defined as the minimizer of the expected loss

$$\boldsymbol{\gamma}^* = \arg\min_{\boldsymbol{\gamma}\in\Gamma} \mathbb{E}[g(\mathbf{W};\boldsymbol{\gamma})],$$

with g chosen as the negative log-likelihood: $g(\mathbf{W}_i;\boldsymbol{\gamma}) = -\left[\delta_i \log \pi(\mathbf{W}_i;\boldsymbol{\gamma}) + (1-\delta_i)\log(1-\pi(\mathbf{W}_i;\boldsymbol{\gamma}))\right]$. Its empirical minimizer is $\widehat{\boldsymbol{\gamma}} = \arg\min_{\boldsymbol{\gamma}\in\Gamma} G(\boldsymbol{\gamma})$, where $G(\boldsymbol{\gamma}) = \frac{1}{N}\sum_{i=1}^{N} g(\mathbf{W}_i;\boldsymbol{\gamma})$.

Noting that subjects with fully observed data constitute a biased sample from the population, we adopt the IPW strategy to correct the selection bias by weighting the complete cases. The global weighted loss function is defined as

$$f^{IPW}(\boldsymbol{\theta},\boldsymbol{\gamma}) = \frac{1}{N}\sum_{i=1}^{N} \frac{\delta_i}{\pi(\mathbf{W}_i;\boldsymbol{\gamma})}\ell(\mathbf{Z}_i;\boldsymbol{\theta}),$$

where the weight $\delta_i/\pi(\mathbf{W}_i;\boldsymbol{\gamma})$ both selects complete observations ($\delta_i = 1$) and upweights those with higher missing probability (Horvitz and Thompson 1952). Under the MAR assumption, this adjustment mitigates bias from missing data. The IPW estimator is then defined as

$$\widehat{\boldsymbol{\theta}}_{IPW} = \arg\min_{\boldsymbol{\theta}\in\Theta} f^{IPW}(\boldsymbol{\theta},\boldsymbol{\gamma}). \tag{2}$$

Consider the distributed setting where the N data are stored disjointly across K local machines, each holding a subsample of n_k observations. Denote $\mathcal{S}_k$ as the sample index set of on the k-th local machine. Let $f_k^{IPW}(\boldsymbol{\theta},\boldsymbol{\gamma}) = \frac{1}{n_k}\sum_{i\in\mathcal{S}_k} \frac{\delta_i}{\pi(\mathbf{W}_i;\boldsymbol{\gamma})}\ell(\mathbf{Z}_i;\boldsymbol{\theta})$ and $\widehat{\boldsymbol{\theta}}_k = \arg\min_{\boldsymbol{\theta}\in\Theta} f_k^{IPW}(\boldsymbol{\theta},\boldsymbol{\gamma})$ be the local loss function and local estimator, respectively. Consequently, the global loss function in (2) can be rewritten as

$$f^{IPW}(\boldsymbol{\theta},\boldsymbol{\gamma}) = \sum_{k=1}^{K} \frac{n_k}{N} f_k^{IPW}(\boldsymbol{\theta},\boldsymbol{\gamma}). \tag{3}$$

Since each machine has access only to its local data, it can compute only its own local loss function. For randomly distributed datasets, Shi et al. (2021) presented a distributed method for estimating $\boldsymbol{\theta}$ by minimizing (3). The performance of their algorithm heavily relies on selecting an appropriate representative local machine for calculating the Hessian matrix. However, no local Hessian matrix can provide a good approximation of the global one in non-randomly distributed setting.

3. Proposed Methods

In this section, we introduce the proposed algorithms for distributed estimation with non-randomly distributed and missing data.

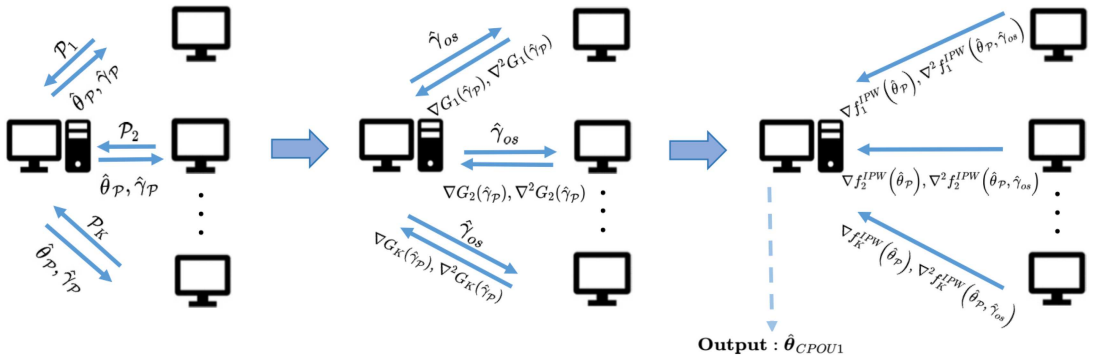

Figure 1. Illustration of the CPOU1 algorithm.

3.1. The CPOU1 Algorithm

We introduce the first CPOU (CPOU1) algorithm. Clearly, directly minimizing f^{IPW} is computationally intensive for distributed data. Based on the local quadratic approximation, Bickel (1975) proposes the one-step Newton-Raphson update estimator

$$\widehat{\boldsymbol{\theta}}_{one-step} = \widehat{\boldsymbol{\theta}}_{ini} - \nabla^2 f^{IPW}(\widehat{\boldsymbol{\theta}}_{ini}, \widehat{\boldsymbol{\gamma}}_{ini})^{-1} \nabla f^{IPW}(\widehat{\boldsymbol{\theta}}_{ini}, \widehat{\boldsymbol{\gamma}}_{ini}),$$

and shows that it is as efficient as global estimator if initial estimate $\widehat{\boldsymbol{\theta}}_{ini}$ is accurate enough. This idea of one-step update can be easily implemented in distributed setting as the additivity of the derivatives. In randomly distributed settings, the average estimator—obtained by simply averaging all local estimators—serves as an effective initial value. However, this method is inefficient under non-randomly distributed mechanisms, where local estimators exhibit significant heterogeneity. Consequently, constructing a proper initial estimator is a key challenge in non-randomly distributed optimization problems.

To address this issue, we mitigate data non-randomness by constructing a pilot sample to compute pilot estimators for use as initial values. We then employ a one-step update strategy to refine these pilot estimators. The proposed CPOU1 method implements this approach through the following three steps.

First, we construct the pilot sample of size n_0 and compute the corresponding pilot estimates. Specifically, we first randomly sample $\tilde{n}_k$ data points from k-th local machines without replacement, where for all k, we requires $\tilde{n}_k/n_k = n_0/N$. Denote the sample index set of subsample selected from the k-th local machine as $\mathcal{P}_k$. Then, the pilot sample index $\mathcal{P} = \bigcup_{k=1}^{K} \mathcal{P}_k$. Let $G_{\mathcal{P}}(\boldsymbol{\gamma})$ and $f_{\mathcal{P}}^{IPW}(\boldsymbol{\theta}, \boldsymbol{\gamma})$ denote the loss functions based on the pilot sample:

$$G_{\mathcal{P}}(\boldsymbol{\gamma}) = \frac{1}{n_0} \sum_{i \in \mathcal{P}} g(\mathbf{W}_i; \boldsymbol{\gamma}) \quad \text{and} \quad f_{\mathcal{P}}^{IPW}(\boldsymbol{\theta}, \boldsymbol{\gamma}) = \frac{1}{n_0} \sum_{i \in \mathcal{P}} \frac{\delta_i}{\pi(\mathbf{W}_i; \boldsymbol{\gamma})} \ell(\mathbf{Z}_i; \boldsymbol{\theta}).$$

The pilot estimates are obtained via

$$\widehat{\boldsymbol{\theta}}_{\mathcal{P}} = \arg\min_{\boldsymbol{\theta} \in \Theta} f_{\mathcal{P}}^{IPW}(\boldsymbol{\theta}, \boldsymbol{\gamma}), \quad \widehat{\boldsymbol{\gamma}}_{\mathcal{P}} = \arg\min_{\boldsymbol{\gamma} \in \Gamma} G_{\mathcal{P}}(\boldsymbol{\gamma}).$$

As illustrated in the left panel of Figure 1, this first step completes the initial communication round with a communication cost of $O(\tilde{n}_k(p+d))$ cost.

Second, we enhance the accuracy of $\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}$ through one-step Newton-Raphson update. Local machines compute the local gradients $\nabla G_k(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})$ and Hessian matrices $\nabla^2 G_k(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})$. Upon receiving these derivatives, the central machine calculates the one-step estimator

$$\widehat{\boldsymbol{\gamma}}_{os} = \widehat{\boldsymbol{\gamma}}_{\mathcal{P}} - \left(\sum_{k=1}^{K} \frac{n_k}{N} \nabla^2 G_k(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) \right)^{-1} \sum_{k=1}^{K} \frac{n_k}{N} \nabla G_k(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}).$$

The middle panel of Figure 1 illustrates this second step, which completes the second communication round with a communication cost of $O(d^2)$.

Finally, we conduct one-step Newton-Raphson update for $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$. Specifically, after obtaining $\widehat{\boldsymbol{\gamma}}_{os}$, the local machines compute the gradients $\nabla f_k^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{os})$ and Hessian matrices $\nabla^2 f_k^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{os})$, for $k = 1, \ldots, K$. The central machine updates $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$ using these derivatives:

$$\widehat{\boldsymbol{\theta}}_{CPOU1} = \widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \left(\sum_{k=1}^{K} \frac{n_k}{N} \nabla^2 f_k^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{os}) \right)^{-1} \sum_{k=1}^{K} \frac{n_k}{N} \nabla f_k^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{os}). \tag{4}$$

The right panel of Figure 1 illustrates this final step, which completes the third communication round at a communication cost of $O(p^2)$.

In summary, the communication cost of the CPOU1 algorithm is $O(\tilde{n}_k(p+d) + d^2 + p^2)$ for k-th local machine. Under a general unbalanced setting where $c \leq n_{k1}/n_{k2} \leq C$ holds for any pair (k_1, k_2) and for positive constants c and C, the communication cost becomes $O(\frac{n_0}{K}(p+d) + d^2 + p^2)$, which decreases as the number of clients K increases. The detailed steps of the CPOU1 algorithm are outlined in Algorithm 1 below.

3.2. The CPOU2 Algorithm

The CPOU1 algorithm requires transferring the Hessian matrices for updating both $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$ and $\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}$. Consequently, the communication complexity is $O(p^2 + d^2)$ bits, which can become prohibitively expensive if the dimensions of $\boldsymbol{\theta}$ or $\boldsymbol{\gamma}$ are large. Hence, we aim to updates the pilot estimate in a more communication-efficient manner.

Algorithm 1 The CPOU1 algorithm.

Round 1: Compute pilot Estimates

1. **for** local machine $k = 1, \ldots, K$ **do**:

(1) Randomly select $\tilde{n}_k$ samples without replacement from $\mathcal{S}_k$, denoted by $\mathcal{P}_k$;

(2) Send the observations indexed by $\mathcal{P}_k$ to the central machine;

2. **for** central machine **do**:

Compute and broadcast the pilot estimates using the pilot samples $\mathcal{P} = \bigcup_k \mathcal{P}_k$:

$$\widehat{\boldsymbol{\gamma}}_{\mathcal{P}} = \arg\min_{\boldsymbol{\gamma}\in\Gamma} \; n_0^{-1} \sum_{i\in\mathcal{P}} g\left(\mathbf{w}_i; \boldsymbol{\gamma}\right);$$

$$\widehat{\boldsymbol{\theta}}_{\mathcal{P}} = \arg\min_{\boldsymbol{\theta}\in\Theta} \; n_0^{-1} \sum_{i\in\mathcal{P}} \frac{\delta_i}{\pi(\mathbf{w}_i;\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})} \ell(\mathbf{z}_i; \boldsymbol{\theta}).$$

Round 2: One-step update for $\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}$

1. **for** local machine $k = 1, \ldots, K$ **do**:

Compute and broadcast the gradient vector and Hessian matrix at the $\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}$:

$$\nabla G_k(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) = n_k^{-1} \sum_{i\in\mathcal{S}_k} \nabla g(\mathbf{w}_i; \widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) \;\text{ and }\; \nabla^2 G_k(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) = n_k^{-1} \sum_{i\in\mathcal{S}_k} \nabla^2 g(\mathbf{w}_i; \widehat{\boldsymbol{\gamma}}_{\mathcal{P}}).$$

2. **for** central machine **do**:

(1) Sum the derivatives respectively:

$$\nabla G(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) = \textstyle\sum_k \frac{n_k}{N} \nabla G_k(\mathbf{w}_i; \widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) \;\text{ and }\; \nabla^2 G(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) = \textstyle\sum_k \frac{n_k}{N} \nabla^2 G_k(\mathbf{w}_i; \widehat{\boldsymbol{\gamma}}_{\mathcal{P}});$$

(2) Compute the one-step Newton-Raphson update for $\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}$:

$$\widehat{\boldsymbol{\gamma}}_{os} = \widehat{\boldsymbol{\gamma}}_{\mathcal{P}} - \left(\nabla^2 G(\mathbf{w}_i; \widehat{\boldsymbol{\gamma}}_{\mathcal{P}})\right)^{-1} \nabla G(\mathbf{w}_i; \widehat{\boldsymbol{\gamma}}_{\mathcal{P}}).$$

Round 3: Compute the CPOU1 Estimate

1. **for** local machine $k = 1, \ldots, K$ **do**:

Compute and broadcast the gradient vector and Hessian matrix at the $\widehat{\boldsymbol{\gamma}}_{os}, \widehat{\boldsymbol{\theta}}_{\mathcal{P}}$:

$$\nabla f_k^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{os}) = n_k^{-1} \sum_{i\in\mathcal{S}_k} \frac{\delta_i}{\pi(\mathbf{w}_i;\widehat{\boldsymbol{\gamma}}_{os})} \nabla \ell(\mathbf{z}_i; \widehat{\boldsymbol{\theta}}_{\mathcal{P}}) \;\text{ and}$$

$$\nabla^2 f_k^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{os}) = n_k^{-1} \sum_{i\in\mathcal{S}_k} \frac{\delta_i}{\pi(\mathbf{w}_i;\widehat{\boldsymbol{\gamma}}_{os})} \nabla^2 \ell(\mathbf{z}_i; \widehat{\boldsymbol{\theta}}_{\mathcal{P}});$$

2. **for** central machine **do**:

(1) Sum the derivatives respectively:

$$\nabla f^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{os}) = \textstyle\sum_k \frac{n_k}{N} \nabla f_k^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}) \;\text{ and}$$

$$\nabla^2 f^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{os}) = \textstyle\sum_k \frac{n_k}{N} \nabla^2 f_k^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}});$$

(2) Compute the one-step Newton-Raphson update for $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$:

$$\widehat{\boldsymbol{\theta}}_{CPOU1} = \widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \left(\nabla^2 f^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{os})\right)^{-1} \nabla f^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{os}).$$

In the randomly distributed setting without missing values, Jordan, Lee, and Yang (2019) proposed the Communication-efficient Surrogate Likelihood (CSL) method, which avoids transmitting high-order derivatives. Starting from an initial estimator $\widehat{\boldsymbol{\theta}}_{ini}$, CSL constructs a surrogate loss

$$\tilde{f}(\boldsymbol{\theta}) \doteq f_1(\boldsymbol{\theta}) - \left\langle \nabla f_1(\widehat{\boldsymbol{\theta}}_{ini}) - \nabla f(\widehat{\boldsymbol{\theta}}_{ini}), \boldsymbol{\theta} \right\rangle,$$

where $f_1(\boldsymbol{\theta}) = \frac{1}{n_1} \sum_{i\in\mathcal{S}_1} \ell(\mathbf{Z}_i; \boldsymbol{\theta})$ is the loss on the first local machine. The estimator is then defined as $\widehat{\boldsymbol{\theta}}_{CSL} = \arg\min_{\boldsymbol{\theta}\in\Theta} \tilde{f}(\boldsymbol{\theta})$. CSL is simple and communication-efficient, but its validity requires $\nabla f_1(\widehat{\boldsymbol{\theta}}_{ini}) - \nabla f(\widehat{\boldsymbol{\theta}}_{ini}) = o_P(1)$, an assumption that may fail under unbalanced or non-random data distributions, leading to inefficiency or inconsistency.

This section introduces a communication-efficient estimator for non-randomly distributed data with missing values, named the CPOU2 algorithm. Building on the framework of the CSL method and pilot sampling, the algorithm achieves efficiency by constructing quadratic approximations of the original loss functions. Its implementation consists of three steps.

The first round of the CPOU2 algorithm involves constructing a pilot sample and obtaining pilot estimates $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$ and $\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}$, which is identical to the procedure in the CPOU1 algorithm. The right panel of Figure 2 illustrates this first round.

Second, we upgrade $\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}$ using a quadratic approximation loss function. By Taylor expansion, $G(\boldsymbol{\gamma})$ is approximated as follows:

$$G(\boldsymbol{\gamma}) \approx G(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) + \left\langle \nabla G(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}), \boldsymbol{\gamma} - \widehat{\boldsymbol{\gamma}}_{\mathcal{P}} \right\rangle + \frac{1}{2} \left\langle \boldsymbol{\gamma} - \widehat{\boldsymbol{\gamma}}_{\mathcal{P}}, \nabla^2 G(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})(\boldsymbol{\gamma} - \widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) \right\rangle.$$

Evaluating the second-order term $\nabla^2 G(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})$ in a communication round necessitates $O(d^2)$ bits from each machine. Inspired by the CSL approach, we replace $\nabla^2 G(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})$ with the Hessian matrix $\nabla^2 G_{\mathcal{P}}(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})$ on the central machine. Thus, we obtain a pilot sample-based surrogate loss function:

$$\tilde{G}(\boldsymbol{\gamma}) = \left\langle \nabla G(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}), \boldsymbol{\gamma} - \widehat{\boldsymbol{\gamma}}_{\mathcal{P}} \right\rangle$$

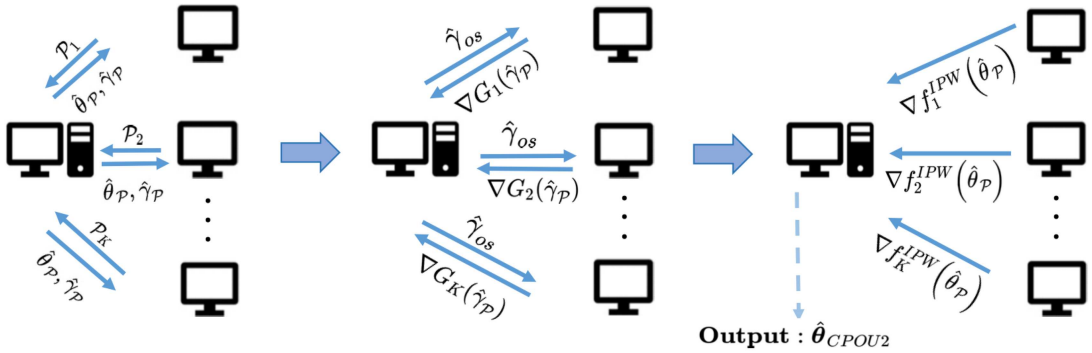

Figure 2. Illustration of the CPOU2 algorithm.

$$+\frac{1}{2}\left\langle \boldsymbol{\gamma}-\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}, \nabla^2 G_{\mathcal{P}}(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})(\boldsymbol{\gamma}-\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})\right\rangle. \quad (5)$$

By minimizing (5), we obtain the closed-form estimator of $\boldsymbol{\gamma}$:

$$\widehat{\boldsymbol{\gamma}}_{cos} = \widehat{\boldsymbol{\gamma}}_{\mathcal{P}} - \nabla^2 G_{\mathcal{P}}(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})^{-1}\nabla G(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}).$$

The right panel of Figure 2 illustrates the second round.

Third, we update $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$ in a communication-efficient way. By the similar arguments for $G(\boldsymbol{\gamma})$, we can construct a quadratic surrogate loss function:

$$\tilde{f}^{IPW}(\boldsymbol{\theta}, \widehat{\boldsymbol{\gamma}}_{cos}) = \left\langle \nabla f^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{cos}), \boldsymbol{\theta}-\widehat{\boldsymbol{\theta}}_{\mathcal{P}}\right\rangle + \frac{1}{2}\left\langle \boldsymbol{\theta}-\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \nabla^2 f_{\mathcal{P}}^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{cos})(\boldsymbol{\theta}-\widehat{\boldsymbol{\theta}}_{\mathcal{P}})\right\rangle. \quad (6)$$

Minimizing (6) yields the closed-form estimator

$$\widehat{\boldsymbol{\theta}}_{CPOU2} = \widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \nabla^2 f_{\mathcal{P}}^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{cos})^{-1}\nabla f^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{cos}); \quad (7)$$

see the right panel of Figure 2.

Compared with the CPOU1 algorithm in Section 3.1, the computation of $\widehat{\boldsymbol{\theta}}_{CPOU2}$ requires only the transfer of gradient vectors. This reduces the communication cost from $O(\tilde{n}_k(p+d)+d^2+p^2)$ to $O(\tilde{n}_k(p+d)+d+p)$ for high-dimensional cases. Hence, the communication advantage of CPOU2 method addresses high-dimensional settings where transmitting second-order statistics becomes the primary bottleneck. In particular, CPOU2 is advantageous when $p \gg \sqrt{\frac{n_0}{K}}$. The detailed algorithm is provided in Algorithm 2.

4. Theoretical Results

This section establishes the consistency and asymptotic normality of the proposed estimators. We begin by introducing necessary notations and assumptions, with the proofs of the main theorems provided in the Supplementary Material.

Let $B_\varepsilon = \{\boldsymbol{\theta} \in \mathbb{R}^p : |\boldsymbol{\theta}-\boldsymbol{\theta}^*|_2 \le \varepsilon\} \subset \Theta$ and $B_\varrho = \{\boldsymbol{\gamma} \in \mathbb{R}^d : |\boldsymbol{\gamma}-\boldsymbol{\gamma}^*|_2 \le \varrho\} \subset \Gamma$ be balls of radii ε and ϱ around $\boldsymbol{\theta}^*$ and $\boldsymbol{\gamma}^*$, respectively. For a vector $\mathbf{v} \in \mathbb{R}^d$, define the Euclidean norm as $|\mathbf{v}|_2 = (\sum_{i=1}^d v_i^2)^{1/2}$. For a matrix $\mathbf{v} \in \mathbb{R}^{d\times d}$, let $|\mathbf{v}|_2$ denote its spectral norm (i.e., its maximum singular value), which satisfies $|\mathbf{v}|_2 = \sup_{\mathbf{u}\in\mathbb{R}^d:|\mathbf{u}|\le 1} |\mathbf{V}\mathbf{u}|_2$. Let $\mathbf{I}_d$ denote the $d \times d$ identity matrix.

Assumption 1. ***(Local convexity)***:

(a) The population loss function for interest parameter $\boldsymbol{\theta}$ is ρ_1-strongly convex at $\boldsymbol{\theta}^*$, that is, the population information matrix $I(\boldsymbol{\theta}^*) = \mathbb{E}[\nabla^2 \ell(\mathbf{Z};\boldsymbol{\theta}^*)] \ge \rho_1 \mathbf{I}_p$.
(b) The population loss function for nuisance parameter $\boldsymbol{\gamma}$ is ρ_2-strongly convex at $\boldsymbol{\gamma}^*$, that is, the population information matrix $H(\boldsymbol{\gamma}^*) = \mathbb{E}[\nabla^2 g(\mathbf{W};\boldsymbol{\gamma}^*)] \ge \rho_2 \mathbf{I}_d$.

Assumption 2. ***(Smoothness for the loss function)***:

(a) *In some neighborhood of* $\boldsymbol{\theta}^*$, *for each* $\mathbf{Z} \in \mathcal{Z}$, $\nabla^2_{\boldsymbol{\theta}}\ell(\mathbf{Z},\cdot)$ *is Lipschitz continuous with Lipschitz constant* $L_1(\mathbf{Z})$ *satisfying* $\mathbb{E}[L_1^2(\mathbf{Z})] \le \infty$. *That is,* $|\nabla^2_{\boldsymbol{\theta}}\ell(\mathbf{Z};\boldsymbol{\theta}) - \nabla^2_{\boldsymbol{\theta}}\ell(\mathbf{Z};\boldsymbol{\theta}')|_2 \le L_1(\mathbf{Z})|\boldsymbol{\theta}-\boldsymbol{\theta}'|_2$, *for all* $\boldsymbol{\theta}, \boldsymbol{\theta}' \in B_\varepsilon$.
(b) *In some neighborhood of* $\boldsymbol{\gamma}^*$, *for every* $\mathbf{W} \in \mathcal{W}$, $\nabla^2_{\boldsymbol{\gamma}} g(\mathbf{W},\cdot)$ *is Lipschitz continuous with Lipschitz constant* $L_2(\mathbf{W})$ *satisfying* $\mathbb{E}[L_2^2(\mathbf{W})] \le \infty$. *That is,* $|\nabla^2_{\boldsymbol{\gamma}} g(\mathbf{W};\boldsymbol{\gamma}) - \nabla^2 g(\mathbf{W};\boldsymbol{\gamma}')|_2 \le L_2(\mathbf{W})|\boldsymbol{\gamma}-\boldsymbol{\gamma}'|_2$, for all $\boldsymbol{\gamma}, \boldsymbol{\gamma}' \in B_\varrho$.

Assumption 3. ***(Smoothness for propensity score)***:

(a) *For every* $\mathbf{W} \in \mathcal{W}$, $\pi(\mathbf{W},\cdot)$ *is continuous on the compact set* Γ *and twice continuously differentiable on the interior of the* Γ.
(b) *For constant* $\eta > 0$, $\pi(\mathbf{W},\boldsymbol{\gamma}) \ge \eta > 0$. *Moreover,* $L_3(\mathbf{W}) \doteq \sup_{\boldsymbol{\gamma}\in\Gamma} \nabla_{\boldsymbol{\gamma}}\pi(\mathbf{W},\gamma)$ *and* $L_4(\mathbf{W}) \doteq \sup_{\boldsymbol{\gamma}\in\Gamma} \nabla^2_{\boldsymbol{\gamma}}\pi(\mathbf{W},\boldsymbol{\gamma})$ *satisfy* $\mathbb{E}\left[L_3^2(\mathbf{W})\right] < \infty$ *and* $\mathbb{E}\left[L_4^2(\mathbf{W})\right] < \infty$, *respectively.*

Assumption 1 is a local identifiability condition, ensuring that $\boldsymbol{\theta}^*$ and $\boldsymbol{\gamma}^*$ are local minima of $F(\boldsymbol{\theta})$ and $G(\boldsymbol{\gamma})$, respectively. Assumption 2 controls moments of higher-order derivatives of the loss functions, and allows us to obtain high-probability bounds on the estimation error. Assumptions 1-2 are standard conditions in distributed statistical inference (Zhang, Wainwright, and Duchi 2012; Fan et al. 2019; Jordan, Lee, and Yang 2019). Assumption 3 is a regular condition ensuring the asymptotic normality of the IPW estimator (Robins, Rotnitzky, and Zhao 1994; Wooldridge 2007).

Theorem 1. Suppose that Assumptions 1–3 hold, then

$$\widehat{\boldsymbol{\theta}}_{CPOU1} - \boldsymbol{\theta}^* = -I(\boldsymbol{\theta}^*)^{-1}\frac{1}{N}\sum_{k=1}^K \sum_{i\in\mathcal{S}_k} \frac{\delta_i}{\pi(\mathbf{W}_i;\boldsymbol{\gamma}^*)}\nabla_{\boldsymbol{\theta}}\ell(\mathbf{Z}_i;\boldsymbol{\theta}^*) + \mathbf{I}(\boldsymbol{\theta}^*)^{-1}\mathbf{C}_0(\widehat{\boldsymbol{\gamma}}_{os}-\boldsymbol{\gamma}^*)$$

Algorithm 2 The CPOU2 algorithm.

Round 1: Compute pilot Estimates

1. **for** local machine $k = 1, \ldots, K$ **do**:
   (1) Randomly select $\tilde{n}_k$ samples without replacement from $\mathcal{S}_k$, denoted by $\mathcal{P}_k$;
   (2) Send the observations indexed by $\mathcal{P}_k$ to the central machine;
2. **for** central machine **do**:
   Compute and broadcast the pilot estimates using the pilot samples $\mathcal{P} = \bigcup_k \mathcal{P}_k$:

$$\widehat{\boldsymbol{\gamma}}_{\mathcal{P}} = \arg\min_{\boldsymbol{\gamma}\in\Gamma} \; n_0^{-1} \sum_{i\in\mathcal{P}} g\left(\mathbf{w}_i; \boldsymbol{\gamma}\right);$$

$$\widehat{\boldsymbol{\theta}}_{\mathcal{P}} = \arg\min_{\boldsymbol{\theta}\in\Theta} \; n_0^{-1} \sum_{i\in\mathcal{P}} \frac{\delta_i}{\pi(\mathbf{w}_i;\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})} \ell(\mathbf{z}_i; \boldsymbol{\theta});$$

3. Compute the Hessian matrix $\nabla^2 G_{\mathcal{P}}(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})$ based on pilot samples:

$$\nabla^2 G_{\mathcal{P}}(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) = \frac{1}{n_0} \sum_{i\in\mathcal{P}} \nabla^2 g(\mathbf{w}_i; \widehat{\boldsymbol{\gamma}}_{\mathcal{P}});$$

Round 2: Surrogate One-step update for $\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}$

1. **for** local machine $k = 1, \ldots, K$ **do**:
   Compute and broadcast the gradient vectors at the $\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}$:

$$\nabla G_k(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) = n_k^{-1} \sum_{i\in\mathcal{S}_k} \nabla g(\mathbf{w}_i; \widehat{\boldsymbol{\gamma}}_{\mathcal{P}}).$$

2. **for** central machine **do**:
   (1) Sum the gradient vectors:

$$\nabla G(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) = \textstyle\sum_k \frac{n_k}{N} \nabla G_k(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}) \; ;$$

   (2) Compute the one-step update for $\widehat{\boldsymbol{\gamma}}_{\mathcal{P}}$:

$$\widehat{\boldsymbol{\gamma}}_{cos} = \widehat{\boldsymbol{\gamma}}_{\mathcal{P}} - \left(\nabla^2 G_{\mathcal{P}}(\widehat{\boldsymbol{\gamma}}_{\mathcal{P}})\right)^{-1} \nabla G(\hat{\boldsymbol{\gamma}}_{\mathcal{P}}).$$

Round 3: Compute the CPOU2 Estimate

1. **For** local machine $k = 1, \ldots, K$ **do**:
   Compute and broadcast the gradient vector $\nabla f_k^{IPW}(\hat{\boldsymbol{\theta}}_{\mathcal{P}})$:

$$\nabla f_k^{IPW}(\hat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{cos}) = n_k^{-1} \sum_{i\in\mathcal{S}_k} \frac{\delta_i}{\pi(\mathbf{w}_i;\widehat{\boldsymbol{\gamma}}_{cos})} \nabla \ell(\mathbf{z}_i; \widehat{\boldsymbol{\theta}}_{\mathcal{P}}).$$

2. **For** central machine **do**:
   (1) Compute the Hessian matrix $f_{\mathcal{P}}^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}})$ based on pilot samples:

$$\nabla^2 f_{\mathcal{P}}^{IPW}(\hat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{cos}) = \frac{1}{n_0} \sum_{i\in\mathcal{P}} \frac{\delta_i}{\pi(\mathbf{w}_i;\widehat{\boldsymbol{\gamma}}_{cos})} \nabla^2 \ell(\mathbf{z}_i; \widehat{\boldsymbol{\theta}}_{\mathcal{P}});$$

   (2) Sum the gradient vectors:

$$\nabla f^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{cos}) = \textstyle\sum_k \frac{n_k}{N} \nabla f_k^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{cos});$$

   (3) Compute the one-step update for $\hat{\boldsymbol{\theta}}_{\mathcal{P}}$:

$$\widehat{\boldsymbol{\theta}}_{CPOU2} = \widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \left(\nabla^2 f_{\mathcal{P}}^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{cos})\right)^{-1} \nabla f^{IPW}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}}, \widehat{\boldsymbol{\gamma}}_{cos}).$$

$$+ O_p\left(||\widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \boldsymbol{\theta}^*||_2^2\right) + o_p\left(||\widehat{\boldsymbol{\gamma}}_{os} - \boldsymbol{\gamma}^*||_2\right), \tag{8}$$

where $\mathbf{C_0} = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \ell(\mathbf{Z}; \boldsymbol{\theta}^*) \nabla_{\boldsymbol{\gamma}} \log \pi(\mathbf{W}; \boldsymbol{\gamma}^*)^\top\right]$.

Theorem 1 establishes the estimation error of the CPOU1 estimator relative to the true parameter. The first term on the right-hand side of (1) resembles the empirical process term in the Bahadur representation under the centralized setting and converges at the $\sqrt{N}$ rate. The overall convergence rate of $\widehat{\boldsymbol{\theta}}_{CPOU1}$ is therefore driven by the convergence rates of the one-step nuisance estimator $\widehat{\boldsymbol{\gamma}}_{os}$ and the pilot estimator $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$.

According to Lemma 1 in Supplementary Material, we have $|\widehat{\boldsymbol{\gamma}}_{os} - \boldsymbol{\gamma}^*|_2 = O_P(N^{-1/2})$ and $|\widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \boldsymbol{\theta}^*|_2^2 = O_P(n_0^{-1})$. Hence, Theorem 1 implies that the ℓ_2-estimation error of the CPOU1 estimator satisfies

$$|\widehat{\boldsymbol{\theta}}_{CPOU1} - \boldsymbol{\theta}^*|_2 = O_P(N^{-1/2} + n_0^{-1}).$$

Moreover, if the pilot sample size satisfies $\sqrt{N}/n_0 = o(1)$, the bias from the pilot estimator becomes negligible, and $\widehat{\boldsymbol{\theta}}_{CPOU1}$ achieves the same efficiency as the global estimator. This result is summarized in the following theorem.

Corollary 1. Under Assumptions 1–3, if the pilot sample size n_0 and the total sample size N satisfy $\sqrt{N}/n_0 = o(1)$, then as $N \to \infty$,

$$\sqrt{N}(\widehat{\boldsymbol{\theta}}_{CPOU1} - \boldsymbol{\theta}^*) \to \mathcal{N}(\mathbf{0}, \mathbf{I}(\boldsymbol{\theta}^*)^{-1}\mathbf{A}\mathbf{I}(\boldsymbol{\theta}^*)^{-1}), \tag{9}$$

where $\mathbf{A} = \mathbb{E}[\mathbf{e}\mathbf{e}^\top]$, $\mathbf{e} = \frac{\delta}{\pi(\mathbf{W};\boldsymbol{\gamma}^*)} \nabla_{\boldsymbol{\theta}} \ell(\mathbf{Z}; \boldsymbol{\theta}^*) - \mathbf{C}_0 H(\boldsymbol{\gamma}^*)^{-1} \nabla_{\boldsymbol{\gamma}} g(\mathbf{W}; \boldsymbol{\gamma}^*)$, *and* $\mathbf{C}_0$ *is defined in the Proposition 1, respectively.*

The similar theoretical results for CPOU2 estimator are established.

Theorem 2. Suppose that Assumptions 1–3 hold, then

$$\begin{aligned}\widehat{\boldsymbol{\theta}}_{CPOU2} - \boldsymbol{\theta}^* = & -I(\boldsymbol{\theta}^*)^{-1}\frac{1}{N}\sum_{k=1}^{K}\sum_{i\in\mathcal{S}_k}\frac{\delta_{ki}}{\pi\left(\mathbf{W}_{ki};\boldsymbol{\gamma}^*\right)}\nabla_{\boldsymbol{\theta}}\ell\left(\mathbf{Z}_i;\boldsymbol{\theta}^*\right)\\ & + \mathbf{I}\left(\boldsymbol{\theta}^*\right)^{-1}\mathbf{C}_0\left(\widehat{\boldsymbol{\gamma}}_{cos} - \boldsymbol{\gamma}^*\right)\\ & + O_p\left(||\widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \boldsymbol{\theta}^*||_2^2 + ||\widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \boldsymbol{\theta}^*||_2\left\|\nabla^2 f_{\mathcal{P}}^{IPW}\right.\right.\\ & \left.\left.\left(\boldsymbol{\theta}^*,\boldsymbol{\gamma}^*\right) - \nabla^2 f^{IPW}(\boldsymbol{\theta}^*,\boldsymbol{\gamma}^*)\right\|_2\right)\\ & + o_P(||\widehat{\boldsymbol{\gamma}}_{cos} - \boldsymbol{\gamma}^*||_2),\end{aligned}$$

where $\mathbf{C_0} = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}}\,\ell(\mathbf{Z};\boldsymbol{\theta}^*)\nabla_{\boldsymbol{\gamma}}\,\log\pi(\mathbf{W};\boldsymbol{\gamma}^*)^\top\right]$.

Under the conditions of Theorem 2, it can be shown that $|\nabla^2 f_{\mathcal{P}}^{IPW}(\boldsymbol{\theta}^*,\boldsymbol{\gamma}^*) - \nabla^2 f^{IPW}(\boldsymbol{\theta}^*,\boldsymbol{\gamma}^*)| = O_P(n_0^{-1/2})$ and $|\widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \boldsymbol{\theta}^*|_2 = O_P(n_0^{-1/2})$ (see Lemma 1 in Supplementary Material). Consequently, $|\widehat{\boldsymbol{\theta}}_{CPOU2} - \boldsymbol{\theta}^*|_2 = O_P(N^{-1/2})$. The asymptotic normality of $\widehat{\boldsymbol{\theta}}_{CPOU2}$ is formally stated in the following theorem.

Corollary 2. Under Assumptions 1–3, if $\sqrt{N}/n_0 = o(1)$, *then as* $N \to \infty$,

$$\sqrt{N}(\widehat{\boldsymbol{\theta}}_{CPOU2} - \boldsymbol{\theta}^*) \to \mathcal{N}(\mathbf{0}, \mathbf{I}(\boldsymbol{\theta}^*)^{-1}\mathbf{A}\mathbf{I}(\boldsymbol{\theta}^*)^{-1}), \tag{10}$$

where $\mathbf{A} = \mathbb{E}[\mathbf{e}\mathbf{e}^\top]$, $\mathbf{e} = \frac{\delta}{\pi(\mathbf{W};\boldsymbol{\gamma}^*)}\nabla_{\boldsymbol{\theta}}\,\ell(\mathbf{Z};\boldsymbol{\theta}^*) - \mathbf{C}_0 H(\boldsymbol{\gamma}^*)^{-1}\nabla_{\boldsymbol{\gamma}}\,g(\mathbf{W};\boldsymbol{\gamma}^*)$.

Corollary 2 indicates that $\widehat{\boldsymbol{\theta}}_{CPOU2}$ is as efficient as the $\widehat{\boldsymbol{\theta}}_{CPOU1}$ for relatively large N and n_0, while $\widehat{\boldsymbol{\theta}}_{CPOU2}$ requires less communication cost. Furthermore, both the $\widehat{\boldsymbol{\theta}}_{CPOU1}$ and $\widehat{\boldsymbol{\theta}}_{CPOU2}$ achieve optimal statistical convergence rate without any restrictive condition on the randomness and completeness of distributed data.

To guarantee asymptotic normality, Corollaries 1 and 2 require the pilot sample size n_0 to grow faster than $\sqrt{N}$. This imposes a constraint on the communication cost, which is of order $O(\frac{n_0}{K}(p+d))$ in the general unbalanced setting (i.e., $c_1 \le \frac{n_{k_1}}{n_{k_2}} \le c_2$ for all pairs (k_1,k_2) and some positive constants c_1 and c_2). Notably, the statistical error rates of the proposed estimators depend on n_0 and N, with no explicit asymptotic constraint on the number of clients K. This is a distinct advantage of our pilot-based framework, as it allows the system to scale to a large number of clients K without compromising statistical efficiency, while also maintaining lower communication costs per machine.

5. Numerical Simulation

In this section, we investigate the performance of the proposed CPOU algorithms through numerical simulation studies. We generate the i.i.d. data $(\mathbf{x}_i, y_i)_{i=1}^N$ as follows. $\mathbf{x}_i = (1, \tilde{\mathbf{x}}_i^\top)^\top$ with $\tilde{\mathbf{x}}_i = (x_1, x_2, \ldots, x_{p-1})^\top N(\mathbf{0}_{p-1}, \Sigma)$, where $\Sigma \in \mathbb{R}^{(p-1)\times(p-1)}$ equals to the identity matrix. y_i takes a value in 0 or 1 with $P\left(y_i = 1|\mathbf{x}_i\right) = \exp(\mathbf{x}_i^\top\boldsymbol{\theta})/\left\{1+\exp(\mathbf{x}_i^\top\boldsymbol{\theta})\right\}$. The dimension is fixed at $p = 5$, and the coefficient vector $\boldsymbol{\theta} = c_0\boldsymbol{\tau}/\|\boldsymbol{\tau}\|_2$, where $\boldsymbol{\tau} \in \mathbb{R}^p$ is drawn from the standard normal distribution and $c_0 = 1.5$. We consider the scenario where some covariates are missing. Specifically, x_1 is missing at random, and the propensity function is specified as $\text{logit}(\pi(y,x_2)) = 5/4 + 1/4y - x_2$, where the $\text{logit}(x) = \log(\frac{x}{1-x})$. For simplicity, let sample size of each local machine $n_k = n = N/K$, where N is the total sample size and K is the number of local machines. To study the performance of the proposed estimators under various data allocation mechanisms, we consider the following cases:

Case I. The data on different local machines are randomly distributed.

Case II. The storage location of each observation depends on its second covariate. Specifically, let $x_{(i)2}$ be the i-th order index of x_{i2}, $i = 1,\ldots,N$, and the (i)-th observation $(\mathbf{x}_{(i)}, y_{(i)})$ is allocated on k-th local machine if it satisfy $(i) \in [(k-1)n+1, kn]$.

Case III. Let $J_i = y_i + \mathbf{x}_i^\top\boldsymbol{\beta}$ with $\boldsymbol{\beta} = (1,\ldots,1)^\top \in \mathbb{R}^p$. Then, J_i is seen as an index, according to which we sort the whole sample. That is, let $J_{(1)} \le \cdots \le J_{(N)}$ be the order index statistic of the J_i s, which leads a new index order (i) for $1 \le i \le N$. The (i)-th observation $(\mathbf{x}_{(i)}, y_{(i)})$ is assigned on k-th local machine if it satisfies $(i) \in [(k-1)n+1, kn]$.

In the first case, data are randomly distributed across different local machines, which represents the ideal scenario assumed by most distributed algorithms. In Case II, the data distribution mechanism is influenced by the covariates. Case III allows that the data distribution mechanism is influenced by both the response and the covariates. Both Case II and III lead to non-random allocations of the entire dataset across K local machines.

For each case, we repeat $B = 200$ independent simulations and report the mean squared estimation error(MSE), namely, $\text{MSE}(\widehat{\boldsymbol{\theta}}) = B^{-1}\sum_{b=1}^{B}|\widehat{\boldsymbol{\theta}}_{(b)} - \boldsymbol{\theta}|_2^2$. We mainly compare the proposed approaches with the following five methods:

- Simultaneous Algorithm (SA): proposed by Shi et al. (2021).
- Distributed Quasi-Newton method (DQN): proposed by Wu, Huang, and Wang (2023).
- Complete-Case (CC): only use the global complete data for parameter estimation.
- Local: use the local data of a local machine for parameter estimation using the IPW method.
- GLO: the global estimator, which uses IPW method to estimate the global likelihood function.

5.1. Effect of the Total Sample Size

In this section, we examine the impact of total sample sizes on the performance of CPOU methods in logistic regression model. The number of local machines K is fixed at $K = 10$, and the total sample size N varies from 1×10^4 to 7×10^4, and the pilot sample size $n_0 = 5\% \times N$.

Figure 3 displays the logarithm of the MSE for all estimators against varying total sample sizes in logistic regression model. In Case I, the whole data is split randomly. The data allocation mechanism in Case II depends solely on $\mathbf{x}$, while in Case III, it depends on both $\mathbf{x}$ and y. In Cases II and III, the SA algorithm fails to produce a result. To display its performance in the figure, we assign the MSE value which equals to log(2) for the SA algorithm. In Case III, the LOCAL estimator fails to converge and it return the result at the maximum of iteration (1000) as its final result. In Case I (randomly distributed setting), the MSE curves of proposed methods and the SA algorithm are

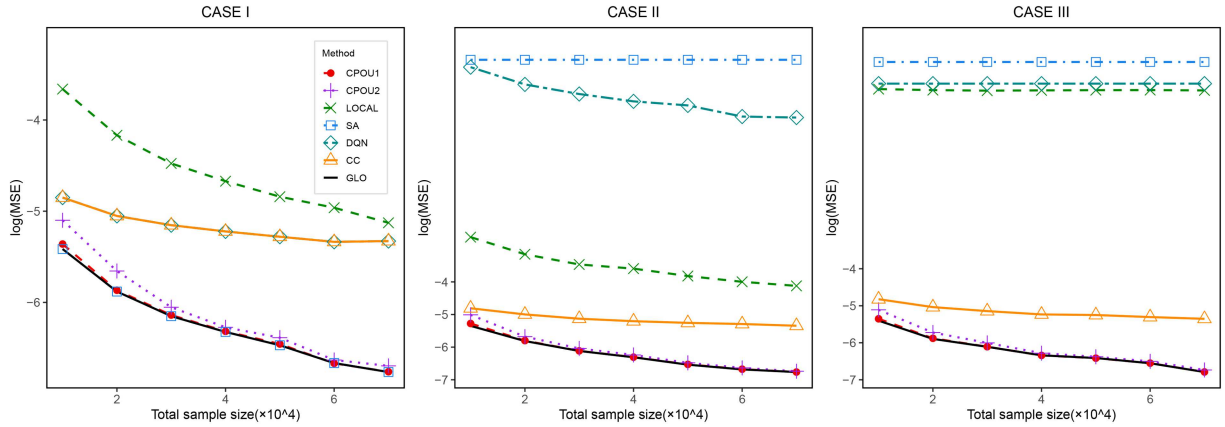

Figure 3. The logarithm of the MSE for all estimators in logistic regression varies with the total sample size N, where the number of local machine K is fixed at $K = 10$. In all cases, each point represents an average of 200 replications.

Table 1. The MSE of the different estimators in the logistic regression model versus the number of machines K ranging from 10 to 400.

p	Case	Method	K				
			10	50	100	200	400
5	I	CPOU1	0.0078	0.0077	0.0077	0.0077	0.0080
		CPOU2	0.0079	0.0077	0.0079	0.0079	0.0082
		SA	0.0078	0.0077	0.0077	0.0077	0.0080
		DQN	0.0241	0.0238	0.0236	0.0234	0.0237
		LOCAL	0.0256	0.0540	0.0789	0.1080	0.1651
		CC	0.0241	0.0238	0.0236	0.0233	0.0237
		GLO	0.0078	0.0077	0.0077	0.0077	0.0080
	II	CPOU1	0.0077	0.0076	0.0079	0.0078	0.0078
		CPOU2	0.0077	0.0078	0.0080	0.0081	0.0080
		SA	—	—	—	—	—
		DQN	1.6596	—	—	—	—
		LOCAL	0.0490	0.1689	0.2137	0.4337	0.6682
		CC	0.0242	0.0237	0.0239	0.0243	0.0234
		GLO	0.0077	0.0076	0.0079	0.0077	0.0078
	III	CPOU1	0.0081	0.0080	0.0081	0.0081	0.0072
		CPOU2	0.0083	0.0081	0.0082	0.0083	0.0074
		SA	—	—	—	—	—
		DQN	—	—	—	—	—
		LOCAL	1.7691	3.0768	3.7360	4.4973	5.1963
		CC	0.0237	0.0236	0.0234	0.0237	0.0233
		GLO	0.0081	0.0080	0.0081	0.0082	0.0072
100	I	CPOU1	0.0075	0.0077	0.0076	0.0075	0.0077
		CPOU2	0.0101	0.0104	0.0101	0.0102	0.0107
		SA	0.0075	0.0076	—	—	—
		DQN	0.0082	0.0090	0.0103	0.0130	—
		LOCAL	0.0242	0.0589	0.0944	0.1836	13.5339
		CC	0.0080	0.0081	0.0081	0.0080	0.0081
		GLO	0.0075	0.0076	0.0075	0.0075	0.0076
	II	CPOU1	0.0076	0.0076	0.0076	0.0075	0.0076
		CPOU2	0.0103	0.0102	0.0102	0.0102	0.0105
		SA	—	—	—	—	—
		DQN	0.0866	5.0002	—	—	—
		LOCAL	0.0254	0.0629	0.1026	0.2107	16.2699
		CC	0.0081	0.0080	0.0081	0.0080	0.0081
		GLO	0.0075	0.0076	0.0076	0.0075	0.0075
	III	CPOU1	0.0077	0.0076	0.0076	0.0077	0.0076
		CPOU2	0.0104	0.0103	0.0104	0.0105	0.0104
		SA	—	—	—	—	—
		DQN	—	—	—	—	—
		LOCAL	0.1153	0.2383	0.3513	1.1049	7.4776
		CC	0.0082	0.0081	0.0081	0.0081	0.0081
		GLO	0.0076	0.0075	0.0075	0.0076	0.0075

almost overlap, demonstrating the performance of these three estimators perform as well as the GLO estimator. In contrast, the CC and DQN methods perform poorly due to systematic differences between observed and missing individuals. The LOCAL estimator shows the worst performance. In Cases II-III (non-randomly distributed framework), both the CPOU1 and CPOU2 estimators maintain performance comparable to the GLO estimator, while all the other estimators (SA, CC, DQN, LOCAL) are inconsistent. Compared to the CPOU2 estimator, the MSEs of the CPOU1 estimator are slightly smaller when $N <$

Table 2. Different missing rates.

$\gamma_0 = -2$	$\gamma_0 = -1$	$\gamma_0 = 0$	$\gamma_0 = 1$	$\gamma_0 = 2$
82.80%	67.13%	47.07%	27.84%	13.93%

6×10^4. Hence, leveraging the global second-order derivatives from all local machines results in better performance. It is worth noting that the proposed algorithms perform well because the pilot sample is a random subsample representing the entire data distribution, regardless of how the subsequent data is split.

5.2. Effect of the Number of Machines

In this experiment, we examine the effect of the number of local machines on the performance of the proposed algorithms and competing methods. The total sample size is fixed at $N = 10 \times 10^4$, with the number of local machines K varying from 10 to 400, and the pilot sample size set to $n_0 = 0.05 \times N$. We consider two settings with covariate dimension $p = 5$ and $p = 100$.

As shown in Table 1, the proposed estimators achieve similar performance with that of the GLO estimator, and their performance remains remarkably stable as the number of machines K increases, for both $p = 5$ and $p = 100$. The LOCAL estimator deteriorates rapidly with increasing K, particularly in Case III. The SA and DQN methods display substantial numerical instability, failing to converge or producing extremely large errors in Cases II-III. Finally, the CC estimator yields relatively stable but consistently larger errors than GLO and CPOU1 estimators.

Regarding communication cost, the SA and DQN methods require $O(p + d)$ and $O(p)$, respectively, while CPOU1 requires $O\left(\frac{n_0}{K}(p + d) + p^2 + d^2\right)$ and CPOU2 requires $O\left(\frac{n_0}{K}(p + d) + p + d\right)$. Although SA and DQN are more communication-efficient, they are ineffective under non-randomly distributed data.

Compared to CPOU1, CPOU2 achieves significantly higher communication efficiency, especially when K is large, and this advantage is further enhanced by increases in both in both K and the covariate dimension p. This is concretely illustrated by the scenario with $K = 400$ and $p = 100$ in Table 1: while CPOU1 must transmit a full 100×100 matrix of second-order information, CPOU2, by employing a pilot-sample Hessian approximation, only needs to transmit data of dimension 12×100, where $[n_0/K] = 12$. This example demonstrates how CPOU2 effectively alleviates the communication burden typically associated with second-order methods in large-scale, high-dimensional settings.

5.3. Effect of Different Missing Rates

In this section, we analyze how the missing rate affects the performance of the estimators. The propensity score function is $\text{logit}(\pi(\mathrm{y}, \mathrm{x}_2)) = \gamma_0 + 1/4\mathrm{y} - \mathrm{x}_2$ with varying $\gamma_0 \in \{-2, -1, 0, 1, 2\}$.

The corresponding missing rates are detailed in Table 2. We set the total sample size $N = 5 \times 10^4$ and the number of machines $K = 10$. The results are presented in Table 3. Table 3 reveals that the MSEs of the CPOU1, CPOU2, and CC estimators decrease as the missing rate reduces in all cases, while the MSEs of the SA, DQN, and LOCAL estimators decrease only in Case I. Furthermore, our proposed methods exhibit MSE values similar to those of the GLO estimator in all Cases.

5.4. Wrongly Specified Missing Model

We also consider the scenario where the missing mechanism model is mis-specified. Specifically, the true propensity score function is not linear in x_2: $\text{logit}(\pi(\mathrm{y}, \mathrm{x}_2)) = 5/4 + 1/4\mathrm{y} - \text{sgn}(\mathrm{x}_2)\sqrt{|\mathrm{x}_2|}$, where $\text{sgn}(\cdot)$ denotes sign function. However, the working model remains $\text{logit}(\pi(\mathrm{y}, \mathrm{x}_2)) = \gamma_0 + \gamma_1\mathrm{y} + \gamma_2\mathrm{x}_2$. Figure 4 records the log (MSE) of all estimators. The results demonstrate that the proposed CPOU1 and CPOU2 methods exhibit robust performance, even under model mis-specification, when compared with the other competing estimators (e.g., SA, CC, DQN, and LOCAL).

6. Real Data Example

As a real-world example, we analyze the 2015 U.S. Behavioral Risk Factor Surveillance System (BRFSS) dataset (https://www.cdc.gov/brfss/index.html), which contains 432,683 respondents after preprocessing (Shi et al. 2021). Our main focus is the relationship between obesity and economic status, where obesity is defined by body mass index (BMI) >25, and income is represented by four dummy variables. Additional covariates include race, sex, and age. BMI and income are partially observed with a missing rate of 21.67%, and we use education level as an auxiliary variable to model the missingness. Table 4 provides detailed information of all variables.

The dataset covers respondents from 53 U.S. states and territories, which we partition into $K = 53$ regional subsets to account for potential regional heterogeneity. Figure 5 illustrates the dataset characteristics, showing substantial variation in both sample sizes and obesity rates across local machines.

We then apply a logistic regression model to examine the factors associated with obesity status. The model is specified as follows:

$$\log \frac{\mathbb{P}(\text{obesity} = 1)}{1 - \mathbb{P}(\text{obesity} = 1)} = \text{intercept} + ic_1 \times \theta_{11} + ic_2 \times \theta_{12} + ic_3 \times \theta_{13} + ic_4 \times \theta_{14} + \text{sex} \times \theta_2 + \text{age} \times \theta_3 + \text{race} \times \theta_4.$$

The pilot sample proportion is set at 5%. For comparison, we also compute the GLO estimator on this dataset. Table 5 presents the regression results obtained by different methods. All predictors are statistically significant with p-values below 0.001. Moreover, the coefficient estimates from our proposed estimators are very close to those of the GLO estimator. These results indicate that, after controlling for race, sex, and age, individuals with higher incomes face a lower risk of obesity.

7. Conclusion

In this study, we focus on distributed learning for non-randomly distributed data containing missing values, where data is non-randomly allocated across different machines and each local dataset contains a significant amount of missing values. Directly

Table 3. The MSE of the different estimators in the logistic regression model versus the intercept γ_0 in the missingness model ($N = 10^5, K = 10$).

Case	Method	Missing rates				
		$\gamma_0 = -2$	$\gamma_0 = -1$	$\gamma_0 = 0$	$\gamma_0 = 1$	$\gamma_0 = 2$
I	CPOU1	0.0242	0.0151	0.0100	0.0079	0.0074
	CPOU2	0.0319	0.0171	0.0105	0.0081	0.0074
	SA	0.0243	0.0150	0.0100	0.0079	0.0074
	DQN	0.0620	0.0519	0.0393	0.0266	0.0166
	LOCAL	0.0746	0.0470	0.0325	0.0256	0.0231
	CC	0.0620	0.0519	0.0393	0.0266	0.0166
	GLO	0.0243	0.0150	0.0100	0.0079	0.0074
II	CPOU1	0.0245	0.0151	0.0104	0.0080	0.0073
	CPOU2	0.0311	0.0168	0.0108	0.0082	0.0074
	SA	—	—	—	—	—
	DQN	3.6809	2.6886	2.3175	1.8761	1.5895
	LOCAL	0.1458	0.0986	0.0691	0.0499	0.0423
	CC	0.0632	0.0516	0.0398	0.0266	0.0168
	GLO	0.0240	0.0148	0.0104	0.0080	0.0073
III	CPOU1	0.0231	0.0146	0.0104	0.0083	0.0073
	CPOU2	0.0295	0.0165	0.0109	0.0085	0.0073
	SA	—	—	—	—	—
	DQN	—	—	—	—	—
	LOCAL	1.7718	1.7724	1.7640	1.7620	1.7644
	CC	0.0605	0.0516	0.0395	0.0269	0.0168
	GLO	0.0227	0.0145	0.0104	0.0083	0.0072

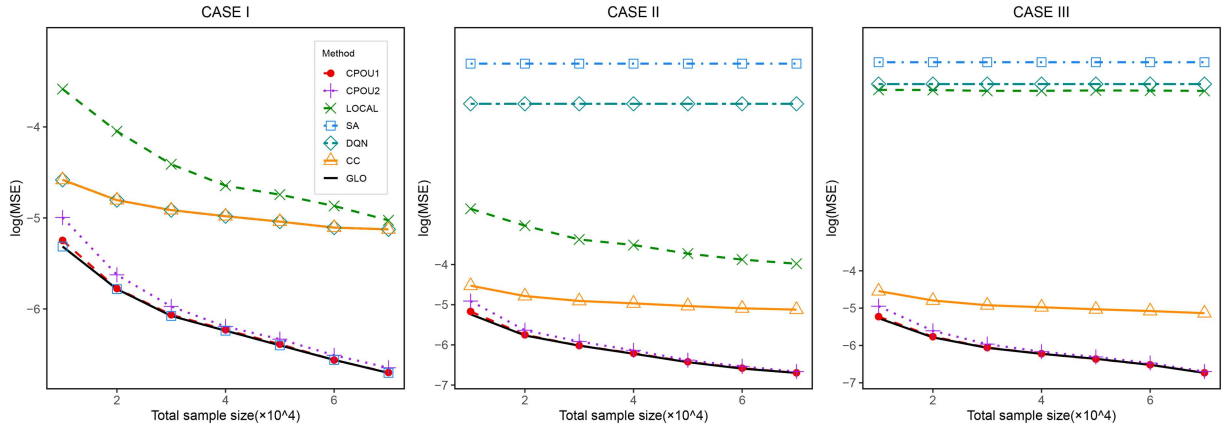

Figure 4. The logarithm of the MSE for all estimators in logistic regression varies with the total sample size N for mis-specified missing model. In all cases, each point represents an average of 200 replications.

Table 4. Description of the variables of the U.S. BRFSS dataset.

	Variable	Description
Response	BMI	Obesity index, whether the BMI $>$ 25 or not: 1 for Yes; 0 for No Yes: 60.7%; No: 31.4%; NA: 9.8%.
Predictors	Income	Household income with 5 levels: $0 - 15,000$: 8.6%; $15,000 - 25,000$: 13.5%; $25,000 - 35,000$: 8.9%; $35,000 - 50,000$: 11.9%; Over 50,000: 39.7%; NA: 17.4%.
	Sex	1 for Male; 0 for Female: Male: 42.2%; Female: 57.8%
	Age	Continuous variable ($18 - 80$).
	Race	Categorical variables with 2 levels:
Auxiliary	Education	Education level variable with 4 levels: Did not graduate High school: 7.7%; Graduated High school: 28.1% Attend College or Technical School: 27.4%; Graduated from College or Technical School: 36.8%

Note: "NA" represents the missingness.

applying existing distributed algorithms in such case can result in severely biased or even inconsistent estimates. We develop two CPOU methods, which handle non-randomly distributed data with missing values through three communication rounds. In the first round, we use IPW to weight the loss function, correcting the bias introduced by missing data. Then, we construct

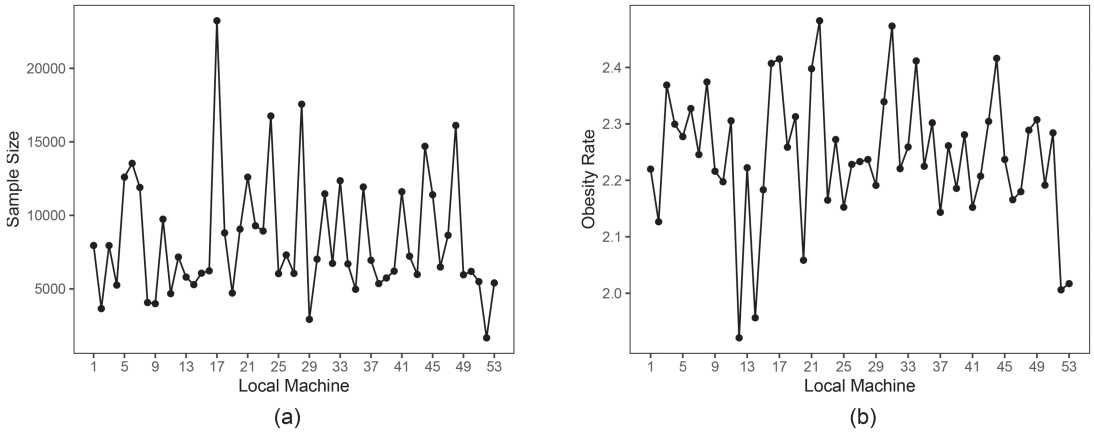

Figure 5. Data distribution characteristics: (a) sample sizes across local machines; (b) obesity rates. Significant disparities in both sample sizes and obesity rates emphasize the regional heterogeneity.

Table 5. Regression results under the GLO, CPOU1, CPOU2 methods.

Variables	GLO			CPOU1			CPOU2		
	Estimate	*SE*	*p*-Value	Estimate	*SE*	*p*-Value	Estimate	*SE*	*p*-Value
(Intercept)	0.1328	0.0147	<0.001	0.1321	0.0147	<0.001	0.1328	0.0147	<0.001
Income_1	0.0809	0.0131	<0.001	0.0819	0.0131	<0.001	0.0818	0.0131	<0.001
Income_2	0.0897	0.0109	<0.001	0.090	0.0109	<0.001	0.0909	0.0109	<0.001
Income_3	0.0843	0.0126	<0.001	0.0842	0.0126	<0.001	0.0835	0.0126	<0.001
Income_4	0.1221	0.1013	<0.001	0.1218	0.1013	<0.001	0.1207	0.1013	<0.001
Race	-0.2844	0.0096	<0.001	-0.2840	0.0096	<0.001	-0.2835	0.0096	<0.001
Sex	0.5969	0.0076	<0.001	0.5965	0.0076	<0.001	0.5967	0.0076	<0.001
Age	0.0091	0.0002	<0.001	0.0091	0.0002	<0.001	0.0091	0.0002	<0.001

Notes: Estimation results for the logistic regression under GLO, CPOU1, and CPOU2 methods. In each method, we report the estimated coefficient (Estimate), standard error (*SE*, $\times 10^2$), and *p*-values for all variables. The *SE* is estimated by the asymptotic standard error based on the global dataset.

an i.i.d. pilot sample and obtain consistent pilot estimates for both the nuisance and interest parameters, where the nuisance parameter is introduced by IPW. In the second and third rounds, we treat the pilot estimates as initial points and perform one-step updates for each parameter. To accommodate varying communication constraints, we adopt two different update strategies, resulting in two distinct methods. The proposed CPOU estimators are proven to be statistically as efficient as the global estimator, without imposing restrictive assumptions about data randomness or completeness. The superior performance is demonstrated through numerical simulations and a real-world dataset analysis.

The current CPOU methods assume that the missing data mechanism is homogeneous across all local machines, however, this assumption may not hold in practice. Extending the CPOU framework to account for heterogeneous missing mechanisms would be interesting. Additionally, in the initial round of the CPOU algorithms, a small subset of raw data is transferred to form a pilot sample, which is not feasible due to privacy concerns in many fields. Therefore, it will be of great importance to explore the privacy preserving distributed method for non-randomly distributed dataset. These promising directions will guide our future research.

Acknowledgments

The authors are grateful to the editors and the referees for their insightful comments and suggestions that considerably improved the quality of the paper.

Author Contributions

Liu developed the idea and its theoretical details and conducted all experiments and practical applications under Wu's supervision. The manuscript was mainly written by Liu, Wu, Yang, and Liu together.

Disclosure Statement

No potential conflict of interest was reported by the author(s).

Funding

The research is supported by the National Natural Science Foundation of China (No: 12271034) and the Open Fund Project of Key Laboratory of Market Regulation (No: 2023SYSKF02003).

References

Bickel, P. J. (1975), "One-Step Huber Estimates in the Linear Model," *Journal of the American Statistical Association*, 70, 428–434. DOI: 10.1080/01621459.1975.10479884. [3]

Cadena, J., Ray, P., Chen, H., Soper, B., Rajan, D., Yen, A., and Goldhahn, R. (2021), "Stochastic Gradient-Based Distributed Bayesian Estimation in Cooperative Sensor Networks," *IEEE Transactions on Signal Processing*, 69, 1713–1724. DOI: 10.1109/TSP.2021.3058765. [1]

Damiani, A., Vallati, M., Gatta, R., Dinapoli, N., Jochems, A., Deist, T., van Soest, J., Dekker, A., and Valentini, V. (2015), "Distributed Learning to Protect Privacy in Multi-Centric Clinical Studies," in *Artificial Intelligence in Medicine: 15th Conference on Artificial Intelligence in Medicine, AIME 2015, Proceedings 15*, June 17–20, 2015, Pavia, Italy, Springer, pp. 65–75. [1]

Fan, J. Q., Guo, Y. Y., and Wang, K. Z. (2023), "Communication-Efficient Accurate Statistical Estimation," *Journal of the American Statistical Association*, 118, 1000–1010. DOI: 10.1080/01621459.2021.1969238. [1]

Fan, J., Wang, D., Wang, K., and Zhu, Z. (2019), "Distributed Estimation of Principal Eigenspaces," *Annals of Statistics*, 47, 3009–3031. DOI: 10.1214/18-AOS1713. [5]

Guo, F., Yu, F. R., Zhang, H., Ji, H., Leung, V. C., and Li, X. (2020), "An Adaptive Wireless Virtual Reality Framework in Future Wireless Networks: A Distributed Learning Approach," *IEEE Transactions on Vehicular Technology*, 69, 8514–8528. DOI: 10.1109/TVT.2020.2995877. [1]

Horvitz, D. G., and Thompson, D. J. (1952), "A Generalization of Sampling without Replacement from a Finite Universe," *Journal of the American Statistical Association*, 47, 663–685. DOI: 10.1080/01621459.1952.10483446. [2]

Huang, C., and Huo, X. M. (2019), "A Distributed One-Step Estimator," *Mathematical Programming*, 174, 41–76. DOI: 10.1007/s10107-019-01369-0. [1]

Jordan, M. I., Lee, J. D., and Yang, Y. (2019), "Communication-Efficient Distributed Statistical Inference," *Journal of the American Statistical Association*, 114, 668–681. DOI: 10.1080/01621459.2018.1429274. [1,4,5]

Li, S., Wang, K., and Xu, Y. (2023), "Robust Estimation for Nonrandomly Distributed Data," *Annals of the Institute of Statistical Mathematics*, 75, 493–509. DOI: 10.1007/s10463-022-00852-4. [1]

Lin, N., and Xi, R. (2011), "Aggregated Estimating Equation Estimation," *Statistics and Its Interface*, 4, 73–83. DOI: 10.4310/SII.2011.v4.n1.a8. [1]

Little, R. J. and Rubin, D. B. (1987), *Statistical Analysis with Missing Data*, New York: John Wiley & Sons. [1]

Liu, Q., and Ihler, A. T. (2014), "Distributed Estimation, Information Loss and Exponential Families," *Advances in Neural Information Processing Systems*, 27, 1098–1106. [1]

Pan, R., Ren, T., Guo, B. S., Li, F., Li, G. D., and Wang, H. S. (2022), "A Note on Distributed Quantile Regression by Pilot Sampling and One-Step Updating," *Journal of Business & Economic Statistics*, 40, 1691–1700. DOI: 10.1080/07350015.2021.1961789. [1]

Pan, Y., Wang, H., Xu, K., and Huang, H. (2025), "Distributed Estimation for Linear Regression with Covariates Missing at Random," *Communications in Statistics-Simulation and Computation*, 54, 583–601. [1]

Robins, J. M., and Rotnitzky, A. (1995), "Semiparametric Efficiency in Multivariate Regression Models with Missing Data," *Journal of the American Statistical Association*, 90, 122–129. DOI: 10.1080/01621459.1995.10476494. [2]

Robins, J. M., Rotnitzky, A., and Zhao, L. P. (1994), "Estimation of Regression Coefficients When Some Regressors Are Not Always Observed," *Journal of the American Statistical Association*, 89, 846–866. DOI: 10.1080/01621459.1994.10476818. [1,5]

Shamir, O., Srebro, N., and Zhang, T. (2014), "Communication-Efficient Distributed Optimization Using an Approximate Newton-Type Method," in *International Conference on Machine Learning*, PMLR, pp. 1000–1008. [1]

Shi, J., Qin, G., Zhu, H., and Zhu, Z. (2021), "Communication-Efficient Distributed m-Estimation with Missing Data," *Computational Statistics & Data Analysis*, 161, 107251. DOI: 10.1016/j.csda.2021.107251. [1,2,7,9]

Wang, F. F., Zhu, Y. Q., Huang, D. Y., Qi, H. B., and Wang, H. S. (2021), "Distributed One-Step Upgraded Estimation for Non-uniformly and Non-randomly Distributed Data," *Computational Statistics & Data Analysis*, 162, 107265. DOI: 10.1016/j.csda.2021.107265. [1]

Wang, K., and Li, S. (2023), "Distributed Statistical Optimization for Non-Randomly Stored Big Data with Application to Penalized Learning," *Statistics and Computing*, 33, 73. DOI: 10.1007/s11222-023-10247-x. [1]

Wooldridge, J. M. (2007), "Inverse Probability Weighted Estimation for General Missing Data Problems," *Journal of Econometrics*, 141, 1281–1301. DOI: 10.1016/j.jeconom.2007.02.002. [5]

Wu, S., Huang, D., and Wang, H. (2023), "Quasi-Newton Updating for Large-Scale Distributed Learning," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 85, 1326–1354. DOI: 10.1093/jrsssb/qkad059. [7]

Xiong, R., Koenecke, A., Powell, M., Shen, Z., Vogelstein, J. T., and Athey, S. (2023), "Federated Causal Inference in Heterogeneous Observational Data," *Statistics in Medicine*, 42, 4418–4439. DOI: 10.1002/sim.9868. [1]

Zhang, X., Zhang, T., and Wang, L. (2023), "Two-Stage Communication-Efficient Distributed Sparse m-Estimation with Missing Data," *Statistics*, 57, 617–636. DOI: 10.1080/02331888.2023.2201505. [1]

Zhang, Y. C., Wainwright, M. J., and Duchi, J. C. (2012), "Communication-Efficient Algorithms for Statistical Optimization," *Advances in Neural Information Processing Systems*, 25, 1502–1510. [1,5]

Zhu, X. N., Li, F., and Wang, H. S. (2021), "Least-Square Approximation for a Distributed System," *Journal of Computational and Graphical Statistics*, 30, 1004–1018. DOI: 10.1080/10618600.2021.1923517.