Journal of the American Statistical Association

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/uasa20

First-Order Newton-Type Estimator for Distributed Estimation and Inference

Xi Chen, Weidong Liu & Yichen Zhang

To cite this article: Xi Chen, Weidong Liu & Yichen Zhang (2021): First-Order Newton-Type Estimator for Distributed Estimation and Inference, Journal of the American Statistical Association, DOI: 10.1080/01621459.2021.1891925

To link to this article: https://doi.org/10.1080/01621459.2021.1891925

JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION
2021, VOL. 00, NO. 0, 1–17: Theory and Methods
https://doi.org/10.1080/01621459.2021.1891925

First-Order Newton-Type Estimator for Distributed Estimation and Inference

Xi Chen[a], Weidong Liu[b], and Yichen Zhang[c]

[a]Stern School of Business, New York University, New York, NY; [b]School of Mathematical Sciences, Shanghai Jiao Tong University, Shanghai, China; [c]Krannert School of Management, Purdue University, West Lafayette, IN

ABSTRACT

This article studies distributed estimation and inference for a general statistical problem with a convex loss that could be nondifferentiable. For the purpose of efficient computation, we restrict ourselves to stochastic first-order optimization, which enjoys low per-iteration complexity. To motivate the proposed method, we first investigate the theoretical properties of a straightforward divide-and-conquer stochastic gradient descent approach. Our theory shows that there is a restriction on the number of machines and this restriction becomes more stringent when the dimension p is large. To overcome this limitation, this article proposes a new multi-round distributed estimation procedure that approximates the Newton step only using stochastic subgradient. The key component in our method is the proposal of a computationally efficient estimator of $\mathbf{\Sigma}^{-1}\boldsymbol{w}$, where $\mathbf{\Sigma}$ is the population Hessian matrix and $\boldsymbol{w}$ is any given vector. Instead of estimating $\mathbf{\Sigma}$ (or $\mathbf{\Sigma}^{-1}$) that usually requires the second-order differentiability of the loss, the proposed first-order Newton-type estimator (FONE) directly estimates the vector of interest $\mathbf{\Sigma}^{-1}\boldsymbol{w}$ as a whole and is applicable to nondifferentiable losses. Our estimator also facilitates the inference for the empirical risk minimizer. It turns out that the key term in the limiting covariance has the form of $\mathbf{\Sigma}^{-1}\boldsymbol{w}$, which can be estimated by FONE.

ARTICLE HISTORY
Received November 2019
Accepted February 2021

KEYWORDS
Distributed inference; Divide-and-conquer; Nonsmooth loss; Quantile regression; Stochastic gradient descent

1. Introduction

The development of modern technology has enabled data collection of unprecedented size, which poses new challenges to many statistical estimation and inference problems. First, given N samples with a very large N, a standard machine might not have enough memory to load the entire dataset all at once. Second, a deterministic optimization approach is computationally expensive. To address the storage and computation issues, distributed computing methods, originated from computer science literature, has been recently introduced into statistics. A general distributed computing scheme partitions the entire dataset into L parts, and then loads each part into the memory to compute a local estimator. The final estimator will be obtained via some communication and aggregation among local estimators.

To further accelerate the computation, we consider stochastic first-order methods (e.g., stochastic gradient/subgradient descent (SGD)), which have been widely adopted in practice. There are a few significant advantages of SGD. First, as a first-order method, it only requires the subgradient information. As compared to *second-order Newton-type* approaches, it is not only computationally efficient and more scalable but also has a wider range of applications to problems where the empirical Hessian matrix does not exist (e.g., when the loss is nonsmooth such as quantile regression). Second, a stochastic approach is usually more efficient than its deterministic counterpart. Although SGD has been widely studied in machine learning and optimization, using SGD for the purpose of statistical inference has not been sufficiently explored.

This article studies a general statistical estimation and inference problem under the distributed computing setup. As we mentioned, to achieve an efficient computation, we restrict ourselves to the use of only stochastic subgradient information. In particular, consider a general statistical estimation problem in the following risk minimization form,

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} F(\boldsymbol{\theta}) := \mathbb{E}_{\boldsymbol{\xi}\sim\Pi} f(\boldsymbol{\theta}, \boldsymbol{\xi}), \tag{1}$$

where $f(\cdot, \boldsymbol{\xi}) : \mathbb{R}^p \to \mathbb{R}$ is a convex loss function that can be nondifferentiable (e.g., in quantile regression), and $\boldsymbol{\xi}$ denotes the random sample from a probability distribution Π (e.g., $\boldsymbol{\xi} = (Y, \mathcal{X})$ in a regression setup). Our goal is to estimate $\boldsymbol{\theta}^* \in \mathbb{R}^p$ under the *diverging dimension case*, where the dimensionality p is allowed to go to infinity as the sample size grows (but p grows at a slower rate than the sample size). This regime is more challenging than the fixed p case. On the other hand, since this work does not make any sparsity assumption, the high dimensional setting where p could be potentially larger than the sample size is beyond our scope. For the ease of illustration, we will use two motivating examples throughout the article: (1) logistic regression with a differentiable loss, and (2) quantile regression with a nondifferentiable loss.

Given n iid samples[1] $\{\boldsymbol{\xi}_i\}_{i=1}^n$, a traditional nondistributed approach for estimating $\boldsymbol{\theta}^*$ is to minimize the empirical risk via

[1]With a slight abuse of notation, we use n to denote either the sample size in nondistributed settings or the local sample size of a single machine in distributed settings.

CONTACT Weidong Liu *weidongl@sjtu.edu.cn* School of Mathematical Sciences, Shanghai Jiao Tong University, Shanghai 200240, China.
Supplementary materials for this article are available online. Please go to *www.tandfonline.com/r/JASA*.

a deterministic optimization:

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{\theta}, \boldsymbol{\xi}_i). \tag{2}$$

Moreover, let $g(\boldsymbol{\theta}, \boldsymbol{\xi})$ be the gradient (when $f(\boldsymbol{\theta}, \boldsymbol{\xi})$ is differentiable) or a subgradient (when $f(\boldsymbol{\theta}, \boldsymbol{\xi})$ is nondifferentiable) of $f(\boldsymbol{\theta}, \boldsymbol{\xi})$ at $\boldsymbol{\theta}$. In terms of *statistical inference*, for many popular statistical models, the empirical risk minimizer (ERM) $\widehat{\boldsymbol{\theta}}$ has an asymptotic normal distribution. That is, under some regularity conditions, for a fixed unit length vector $\boldsymbol{w} \in \mathbb{R}^p$, as $n, p \to \infty$,

$$\frac{\sqrt{n}\boldsymbol{w}'(\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*)}{\sqrt{\boldsymbol{w}'\boldsymbol{\Sigma}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}^{-1}\boldsymbol{w}}} \to \mathcal{N}(0, 1), \tag{3}$$

where

$$\begin{aligned} \boldsymbol{\Sigma} &:= \nabla_{\boldsymbol{\theta}} \mathbb{E} g(\boldsymbol{\theta}, \boldsymbol{\xi})|_{\boldsymbol{\theta}=\boldsymbol{\theta}^*} \\ \boldsymbol{A} &= \operatorname{cov}(g(\boldsymbol{\theta}^*, \boldsymbol{\xi})) = \mathbb{E}\left[g(\boldsymbol{\theta}^*, \boldsymbol{\xi}) g(\boldsymbol{\theta}^*, \boldsymbol{\xi})'\right]. \end{aligned} \tag{4}$$

Under this framework, the main goal of our article is 2-fold:

1. Distributed estimation: Develop a distributed stochastic first-order method for estimating $\boldsymbol{\theta}^*$ in the case of diverging p, with the aim to achieve the best possible convergence rate (i.e., the rate of the pooled ERM estimator $\widehat{\boldsymbol{\theta}}$). The method should be applicable to nondifferentiable loss $f(\boldsymbol{\theta}, \boldsymbol{\xi})$ and only requires the local strong convexity of $F(\boldsymbol{\theta})$ at $\boldsymbol{\theta} = \boldsymbol{\theta}^*$ (instead of the strong convexity of $F(\boldsymbol{\theta})$ for any $\boldsymbol{\theta}$).
2. Distributed inference: Based on (3), develop a consistent estimator of the limiting variance $\boldsymbol{w}'\boldsymbol{\Sigma}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ to facilitate the inference.

Let us first focus on the distributed estimation problem. We will first investigate the theoretical proprieties of a straightforward method that combines the stochastic subgradient descent (SGD) and divide-and-conquer (DC) scheme and discuss the theoretical limitation of this method. To overcome the theoretical limitation, we propose a new method called the distributed first-order Newton-type estimator (FONE), where the key idea is to approximate the Newton step only using stochastic subgradient information in a distributed setting.

In a distributed setting, the DC strategy has been recently adopted in many statistical estimation problems (see, e.g., Li, Lin, and Li 2013; Chen and Xie 2014; Zhang, Duchi, and Wainwright 2015; Zhao, Cheng, and Liu 2016; Battey et al. 2018; Shi, Lu, and Song 2018; Banerjee, Durot, and Sen 2019; Huang and Huo 2019; Fan et al. 2019; Volgushev, Chao, and Cheng 2019). A standard DC approach estimates a local estimator for each local machine. and then aggregates the local estimators to obtain the final estimator. Combining the idea of DC with the mini-batch SGD naturally leads to a DC-SGD approach, where we run SGD on each local machine and then aggregate the obtained solutions by an averaging operation. In fact, DC-SGD is not the main focus/contribution of this article. It has been an existing popular distributed algorithm in practice for a long time. Nevertheless, the theoretical property of DC-SGD with mini-batch in the diverging dimension case has not been fully understood yet. We first establish the theoretical properties of DC-SGD and explain its limitations, which better motivates our distributed estimator (see below). For DC-SGD to achieve the optimal convergence rate, the number of machines L has to be $O(\sqrt{N/p})$ (see Section 3.1), where N is the total number of samples across L machines. The condition could be restrictive when the number of machines is large but each local machine has a limited storage (e.g., in a large-scale sensor network). Moreover, as compared to the standard condition $L = O(\sqrt{N})$ in a fixed p setting, the condition $L = O(\sqrt{N/p})$ becomes more stringent when p diverges. In fact, this constraint is not only for the case of DC-SGD. Since the averaging only reduces the variance but not the bias term, all the results for the standard DC approach in the literature inevitably involve a constraint on the number of machines, which aims to make the variance the dominating term.

To relax this condition on L and further improve the performance of DC-SGD, this article proposes a new approach called distributed FONE, which successively refines the estimator by multi-round aggregations. The starting point of our approach is the Newton-type method based on a consistent initial estimator $\widehat{\boldsymbol{\theta}}_0$:

$$\widetilde{\boldsymbol{\theta}} = \widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\Sigma}^{-1}\left(\frac{1}{n}\sum_{i=1}^{n} g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i)\right), \tag{5}$$

where $\boldsymbol{\Sigma}$ is the population Hessian matrix and $\left(\frac{1}{n}\sum_{i=1}^{n} g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i)\right)$ is the subgradient vector. However, the estimation of $\boldsymbol{\Sigma}$ is not easy when f is nondifferentiable and the empirical Hessian matrix does not exist.

To address this issue, our key idea is that instead of estimating $\boldsymbol{\Sigma}$ and computing its inverse, we propose an estimator of $\boldsymbol{\Sigma}^{-1}\boldsymbol{w} \in \mathbb{R}^p$ for any given vector $\boldsymbol{w} \in \mathbb{R}^p$, which solves (5) as a special case (with $\boldsymbol{w} = \frac{1}{n}\sum_{i=1}^{n} g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i)$). In fact, the estimator of $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ kills two birds with one stone: it not only constructs a Newton-type estimator of $\boldsymbol{\theta}^*$ but also provides an estimator for the asymptotic variance in (3), which facilitates the inference. In particular, the proposed FONE estimator of $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ is an iterative procedure that only uses the mini-batches of subgradient to approximate the Newton step. It is also worthwhile noting that our method extends the recent work by Jordan, Lee, and Yang (2019) and Wang and Zhang (2017), which approximates the Newton step by using local Hessian matrix computed on a single machine. However, to compute the local Hessian matrix, their method requires the second-order differentiability on the loss function and thus is not applicable to problems such as quantile regression. In contrast, our approach approximates the Newton step via stochastic subgradient and thus can handle the nondifferentiability in the loss function. We also note that the idea of approximating Newton step has been applied to specific statistical learning problems, such as SVM (Wang et al. 2019), quantile regression (Chen, Liu, and Zhang 2019; Chen et al. 2020), and PCA (Chen et al. 2021). This article provides a general framework for both smooth and nonsmooth loss functions. Moreover, our method subsumes a recently developed stochastic first-order approach—stochastic variance reduced gradient (SVRG, see, e.g., Johnson and Zhang 2013; Lee et al. 2017; Wang and Zhang 2017; Li et al. 2018, and references therein) as a special case. While SVRG requires $\boldsymbol{w}$ to be the averaged gradient and its theory only applies to strongly convex smooth loss functions, we allow a general $\boldsymbol{w}$ vector and nonsmooth losses.

Based on FONE, we further develop a multi-round distributed version of FONE which successively refines the estimator and does not impose any strict condition on the number of machines L. Theoretically, we show that for a smooth loss, when the number of rounds K exceeds a constant threshold K_0, the obtained distributed FONE $\widehat{\boldsymbol{\theta}}_{\text{dis},K}$ achieves the optimal convergence rate. For a nonsmooth loss, such as quantile regression, our convergence rate only depends on the sample size of one local machine with the largest subsample size. This condition is weaker than the case of DC-SGD since the bottleneck in the convergence of DC-SGD is the local machine with the smallest subsample size.

We further apply the developed FONE to obtain a consistent estimator of the asymptotic variance in (3) for the purpose of inference. Note that the term $\boldsymbol{A}$ can be easily estimated via replacing the expectation by its sample version. Instead of estimating $\boldsymbol{\Sigma}^{-1}$ in (3), our method directly estimates $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ for any fixed unit length vector $\boldsymbol{w}$ (please see Section 4 for more details).

The remainder of the article is organized as follows: Section 2.1 describes the mini-batch SGD algorithm with diverging dimension and the DC-SGD estimator. We further propose FONE and distributed FONE in Section 2.2. Section 3 presents the theoretical results. Section 4 discusses the application of FONE to the inference problem. In Section 5, we demonstrate the performance of the proposed estimators by simulation experiments and real data analysis, followed by conclusions in Section 6. Some additional theoretical results, technical proofs and additional experimental results are provided in the supplementary materials.

In this article, we denote the Euclidean norm for a vector $\boldsymbol{x} \in \mathbb{R}^p$ by $\|\boldsymbol{x}\|_2$, and we denote the spectral norm for a matrix $\mathcal{X}$ by $\|\mathcal{X}\|$. In addition, since the distributed estimation and inference usually involve quite a few notations, we briefly summarize them here. We use N, L, $n = N/L$, and m to denote the total number of samples, the number of machines (or the number of data partitions), the sample size on each local machine (when evenly distributed), and the batch size for mini-batch SGD, respectively. When we discuss a problem in the classical single machine setting, we will also use n to denote the sample size. We will use $\boldsymbol{\theta}^*$, $\widehat{\boldsymbol{\theta}}$, and $\widehat{\boldsymbol{\theta}}_0$ to denote the minimizer of the popular risk, the ERM, and the initial estimator, respectively. The random sample will be denoted by $\boldsymbol{\xi}$ and in a regression setting $\boldsymbol{\xi} = (Y, \mathcal{X})$.

2. Methodology

In this section, we first introduce the standard DC-SGD algorithm. The main purpose of introducing this DC-SGD algorithm is to better motivate the proposed FONE and its distributed version.

2.1. Divide-and-Conquer SGD (DC-SGD) Algorithm

Before we introduce our DC-SGD algorithm, we first present the mini-batch SGD algorithm for solving the stochastic optimization in (1) on a single machine with total n samples. In particular, we consider the setting when the dimension $p \to \infty$ but at a slower rate than n, that is, $p \le n^\kappa$ for some $\kappa \in (0, 1)$. Given n iid samples $\{\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_n\}$, we partition the index set $\{1, \ldots, n\}$ into s disjoint mini-batches $H_1, \ldots, H_s$, where each mini-batch has the size $|H_i| = m$ (for $i = 1, 2, \ldots, s$), and $s = n/m$ is the number of mini-batches. The mini-batch SGD algorithm starts from a consistent initial estimator $\widehat{\boldsymbol{\theta}}_0$ of $\boldsymbol{\theta}^*$. Let $\boldsymbol{z}_0 = \widehat{\boldsymbol{\theta}}_0$. The mini-batch SGD iteratively updates $\boldsymbol{z}_i$ from $\boldsymbol{z}_{i-1}$ as follows and outputs $\widehat{\boldsymbol{\theta}}_{\text{SGD}} = \boldsymbol{z}_s$ as its final estimator,

$$\boldsymbol{z}_i = \boldsymbol{z}_{i-1} - \frac{r_i}{m} \sum_{j \in H_i} g(\boldsymbol{z}_{i-1}, \boldsymbol{\xi}_j), \quad \text{for} \quad i = 1, 2, \ldots, s, \tag{6}$$

where we set the step-size $r_i = c_0 / \max(i^\alpha, p)$ for some $0 < \alpha \le 1$ and c_0 is a positive constant. It is worthwhile that a typical choice of r_i in the literature is $r_i = c_0 \cdot i^{-\alpha}$ (Polyak and Juditsky 1992; Chen et al. 2020). Since we are considering a diverging p case, our step-size incorporates the dimension p. As one can see, this mini-batch SGD algorithm only uses one pass of the data and enjoys a low per-iteration complexity.

We provide two examples on logistic regression and quantile regression to illustrate the subgradient function $g(\boldsymbol{\theta}, \boldsymbol{\xi})$ in our mini-batch SGD and will refer to these examples throughout the article.

Example 2.1 (Logistic regression). Consider a logistic regression model with the response $Y \in \{-1, 1\}$, where

$$\mathbb{P}(Y = 1|\mathcal{X}) = 1 - \mathbb{P}(Y = -1|\mathcal{X}) = \frac{1}{1 + \exp(-\mathcal{X}'\boldsymbol{\theta}^*)},$$

and $\boldsymbol{\theta}^* \in \mathbb{R}^p$ is the true model parameter. Define $\boldsymbol{\xi} = (Y, \mathcal{X})$. We have the smooth loss function $f(\boldsymbol{\theta}, \boldsymbol{\xi}) = \log(1 + \exp(-Y\mathcal{X}'\boldsymbol{\theta}))$ and its gradient $g(\boldsymbol{\theta}, \boldsymbol{\xi}) = -Y\mathcal{X}\big(1 + \exp(Y\mathcal{X}'\boldsymbol{\theta})\big)^{-1}$.

Example 2.2 (Quantile regression). Consider a quantile regression model $Y = \mathcal{X}'\boldsymbol{\theta}^* + \epsilon$, where we assume that $\mathcal{X} = (1, X_1, \ldots, X_{p-1})'$ and $\mathbb{P}(\epsilon \le 0|\mathcal{X}) = \tau$ is the so-called quantile level. Define $\boldsymbol{\xi} = (Y, \mathcal{X})$. We have the nonsmooth quantile loss function $f(\boldsymbol{\theta}, \boldsymbol{\xi}) = \ell_\tau(Y - \mathcal{X}'\boldsymbol{\theta})$ and $\ell_\tau(x) = x(\tau - I\{x \le 0\})$. A subgradient of the quantile loss is given by $g(\boldsymbol{\theta}, \boldsymbol{\xi}) = \mathcal{X}(I\{Y \le \mathcal{X}'\boldsymbol{\theta}\} - \tau)$.

The bias and L_2-estimation error of the mini-batch SGD will be provided in Theorem B.1 (see Section B in the supplementary materials). In particular, in the diverging dimension setting, it is necessary to have a consistent initial estimator to guarantee the consistency of obtained solution from the mini-batch SGD (see Proposition B.2 in the supplementary materials).

Given the mini-batch SGD, we are ready to introduce the DC-SGD. For the ease of illustration, suppose that the entire sample with the size N is evenly distributed on L machines (or split into L parts) with the subsample size $n = N/L$ on each local machine. For the ease of presentation, we assume that N/L is a positive integer. On each machine $k = 1, 2, \ldots, L$, we run the mini-batch SGD with the batch size m in (6). Let $\mathcal{H}_k$ be the indices of the data points on the kth machine, which is further split into s mini-batches $\{H_{k,i}, i = 1, 2, \ldots, s\}$ with $|H_{k,i}| = m$ and $s = n/m$. On the kth machine, we run our mini-batch SGD in (6) and obtain the local estimator $\widehat{\boldsymbol{\theta}}^{(k)}_{\text{SGD}}$. The final estimator is

Algorithm 1 DC-SGD algorithm

Input: The initial estimator $\widehat{\boldsymbol{\theta}}_0 \in \mathbb{R}^p$, the step-size sequence $r_i = c_0/\max(i^\alpha, p)$ for some $0 < \alpha \leq 1$, the mini-batch size m.

1: Distribute the initial estimator $\widehat{\boldsymbol{\theta}}_0$ to each local machine $k = 1, 2, \ldots, L$.
2: **for** each local machine $k = 1, 2, \ldots, L$ **do**
3: Set the starting point $\boldsymbol{z}_0^{(k)} = \widehat{\boldsymbol{\theta}}_0$.
4: **for** each iteration $i = 1, \ldots, s$ **do**
5: Update

$$\boldsymbol{z}_i^{(k)} = \boldsymbol{z}_{i-1}^{(k)} - \frac{r_i}{m} \sum_{j \in H_{k,i}} g(\boldsymbol{z}_{i-1}^{(k)}, \boldsymbol{\xi}_j),$$

6: **end for**
7: Set $\widehat{\boldsymbol{\theta}}_{\mathrm{SGD}}^{(k)} = \boldsymbol{z}_s^{(k)}$ as the local SGD estimator on the machine k.
8: **end for**
9: Aggregate the local estimators $\widehat{\boldsymbol{\theta}}_{\mathrm{SGD}}^{(k)}$ by averaging and compute the final estimator:

$$\widehat{\boldsymbol{\theta}}_{\mathrm{DC}} = \frac{1}{L} \sum_{k=1}^{L} \widehat{\boldsymbol{\theta}}_{\mathrm{SGD}}^{(k)}.$$

10: **Output:** $\widehat{\boldsymbol{\theta}}_{\mathrm{DC}}$.

aggregated by averaging the local estimators from L machines, that is,

$$\widehat{\boldsymbol{\theta}}_{\mathrm{DC}} = \frac{1}{L} \sum_{k=1}^{L} \widehat{\boldsymbol{\theta}}_{\mathrm{SGD}}^{(k)}. \tag{7}$$

Note that the DC-SGD algorithm only involves one round of aggregation. The details of the DC-SGD are presented in Algorithm 1.

In Theorem 3.1, we establish the convergence rate of the DC-SGD in terms of the dimension p, the number of machines L, the total sample size N and the mini-batch size m. Moreover, we show that for the DC-SGD to achieve the same rate as the mini-batch SGD running on the entire dataset, it requires a condition on the number of machines L.This condition on L is essential because the averaging scheme in a DC approach only reduces the variance but not the bias term.

2.2. First-Order Newton-Type Estimator (FONE)

To relax the condition on the number of machines L, one idea is to perform a Newton-type step in (5). However, as we have pointed out, the estimation of $\boldsymbol{\Sigma}$ requires the second-order differentiability of the loss function. Moreover, a typical Newton method successively refines the estimator of $\boldsymbol{\Sigma}$ based on the current estimate of $\boldsymbol{\theta}^*$ and thus requires the computation of matrix inversion in (5) for multiple iterations, which could be computationally expensive when p is large.

In this section, we propose a new FONE that directly estimates $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$ (for any given vector $\boldsymbol{a}$) only using the stochastic first-order information. Then for a given initial estimator $\widehat{\boldsymbol{\theta}}_0$, we can perform the Newton-type step in (5) as

$$\tilde{\boldsymbol{\theta}} = \widehat{\boldsymbol{\theta}}_0 - \widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{a}}, \qquad \boldsymbol{a} = \left(\frac{1}{n} \sum_{i=1}^{n} g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i) \right), \tag{8}$$

where $\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{a}}$ is our estimator of $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$.

To estimate $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$, we note that $\boldsymbol{\Sigma}^{-1}\boldsymbol{a} = \sum_{i=0}^{\infty}(I - \eta\boldsymbol{\Sigma})^i \eta \boldsymbol{a}$ for small enough η such that $\|\eta\boldsymbol{\Sigma}\| < 1$. Then we can use the following iterative procedure $\{\tilde{\boldsymbol{z}}_t\}$ to approximate $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$:

$$\tilde{\boldsymbol{z}}_t = \tilde{\boldsymbol{z}}_{t-1} - \eta(\boldsymbol{\Sigma}\tilde{\boldsymbol{z}}_{t-1} - \boldsymbol{a}), \ 1 \leq t \leq T, \tag{9}$$

where η here can be viewed as a constant step-size. To see that (9) leads to an approximation of $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$, when T is large enough, we have

$$\begin{aligned} \tilde{\boldsymbol{z}}_T &= \tilde{\boldsymbol{z}}_{T-1} - \eta(\boldsymbol{\Sigma}\tilde{\boldsymbol{z}}_{T-1} - \boldsymbol{a}) = (I - \eta\boldsymbol{\Sigma})\tilde{\boldsymbol{z}}_{T-1} + \eta\boldsymbol{a} \\ &= (I - \eta\boldsymbol{\Sigma})^2 \tilde{\boldsymbol{z}}_{T-2} + (I - \eta\boldsymbol{\Sigma})\eta\boldsymbol{a} + \eta\boldsymbol{a} \\ &= (I - \eta\boldsymbol{\Sigma})^{T-1}\tilde{\boldsymbol{z}}_1 + \sum_{i=0}^{T-2}(I - \eta\boldsymbol{\Sigma})^i \eta\boldsymbol{a} \approx \boldsymbol{\Sigma}^{-1}\boldsymbol{a}. \end{aligned}$$

As the iterate $\tilde{\boldsymbol{z}}_t$ approximates $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$, let us define $\boldsymbol{z}_t = \widehat{\boldsymbol{\theta}}_0 - \tilde{\boldsymbol{z}}_t$, which is the quantity of interest (see the left-hand side of the Newton-type step in (8)). To avoid estimating $\boldsymbol{\Sigma}$ in the recursive update in (9), we adopt the following first-order approximation:

$$-\boldsymbol{\Sigma}\tilde{\boldsymbol{z}}_{t-1} = \boldsymbol{\Sigma}(\boldsymbol{z}_{t-1} - \widehat{\boldsymbol{\theta}}_0) \approx g_{B_t}(\boldsymbol{z}_{t-1}) - g_{B_t}(\widehat{\boldsymbol{\theta}}_0), \tag{10}$$

where $g_{B_t}(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i \in B_t} g(\boldsymbol{\theta}, \boldsymbol{\xi}_i)$ is the averaged stochastic subgradient over a subset of the data indexed by $B_t \subseteq \{1, 2, \ldots, n\}$. Here B_t is randomly chosen from $\{1, \ldots, n\}$ with replacement for every iteration.

Given (10), we construct our FONE of $\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\Sigma}^{-1}\boldsymbol{a}$ by the following recursive update from $t = 1, 2, \ldots, T$:

$$\boldsymbol{z}_t = \boldsymbol{z}_{t-1} - \eta\{g_{B_t}(\boldsymbol{z}_{t-1}) - g_{B_t}(\widehat{\boldsymbol{\theta}}_0) + \boldsymbol{a}\}, \quad \boldsymbol{z}_0 = \widehat{\boldsymbol{\theta}}_0. \tag{11}$$

The obtained $\boldsymbol{z}_T$, as an estimator of $\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\Sigma}^{-1}\boldsymbol{a}$ can be directly used in the Newton-type step in (8). The choices of the input parameters and the convergence rate of our FONE will be proved in Propositions 3.1 and 3.2. Also note that for constructing the estimator of $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$, we can simply use $\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{z}_T$ and the procedure is summarized in Algorithm 2.

2.3. Distributed FONE for Estimating θ^*

Based on the FONE for $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$, we present a distributed FONE for estimating $\boldsymbol{\theta}^*$. Suppose the entire dataset with N samples is distributed on L local machines $\{\mathcal{H}_1, \mathcal{H}_2, \ldots, \mathcal{H}_L\}$ (not necessarily evenly distributed). Our distributed FONE is a multi-round approach with K rounds, where K is a prespecified constant. For each round $j = 1, 2, \ldots, K$, with the initialization $\widehat{\boldsymbol{\theta}}_{j-1}$, we first calculate $\boldsymbol{a} = \frac{1}{N}\sum_{i=1}^{N} g(\widehat{\boldsymbol{\theta}}_{j-1}, \boldsymbol{\xi}_i)$ by averaging the subgradients from each local machine. Then we apply FONE (Algorithm 2) with $\boldsymbol{a}$ on the local machine with the largest subsample size. Since FONE is performed on one local machine, this iterative procedure does not incur any extra communication cost. The detailed algorithm is given in Algorithm 3. In fact, the presented Algorithm 3 is essentially estimating $\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\Sigma}^{-1}\boldsymbol{a}$ with $\boldsymbol{a} = \frac{1}{N}\sum_{i=1}^{N} g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i)$ and $\widehat{\boldsymbol{\theta}}_0$ is a pregiven initial estimator.

Algorithm 2 First-order Newton-type estimator (FONE) of $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$

Input: Dataset $\{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_n\}$, the initial estimator $\widehat{\boldsymbol{\theta}}_0$, step-size η, the batch-size m, and a given vector $\boldsymbol{a} \in \mathbb{R}^p$.

1: Set $\boldsymbol{z}_0 = \widehat{\boldsymbol{\theta}}_0$.
2: **for** each $t = 1, 2, \ldots, T$ **do**
3: Choose B_t to be m distinct elements uniformly from $\{1, 2, \ldots, n\}$.
4: Calculate

$$g_{B_t}(\boldsymbol{z}_{t-1}) = \frac{1}{m}\sum_{i\in B_t} g(\boldsymbol{z}_{t-1}, \boldsymbol{\xi}_i),$$

$$g_{B_t}(\boldsymbol{z}_0) = \frac{1}{m}\sum_{i\in B_t} g(\boldsymbol{z}_0, \boldsymbol{\xi}_i).$$

5: Update

$$\boldsymbol{z}_t = \boldsymbol{z}_{t-1} - \eta\{g_{B_t}(\boldsymbol{z}_{t-1}) - g_{B_t}(\boldsymbol{z}_0) + \boldsymbol{a}\}.$$

6: **end for**
7: **Output:**

$$\widehat{\boldsymbol{\theta}}_{\mathrm{FONE}} = \widehat{\boldsymbol{\theta}}_0 - \boldsymbol{z}_T. \tag{12}$$

It is worthwhile noting that in contrast to DC-SGD where each local machine plays the same role, distributed FONE performs the update in (14) only on one local machine. The convergence rate of distributed FONE will depend on the subsample size of this machine (see Theorems 3.2 and 3.3). Therefore, to achieve the best convergence rate, we perform the update in (14) on the machine with the largest subsample size and *index it by the first machine* without loss of generality. We also note that since the first machine collects the gradient and performs FONE, the distributed FONE can be easily implemented in a de-centralized setting.

Instead of using the first machine to compute FONE, one can further leverage the idea of the DC in the distributed FONE. In particular, one may let each local machine run FONE based on the aggregated gradient $\boldsymbol{a}$ simultaneously, and then take the average of all the local estimators on L machines as $\widehat{\boldsymbol{\theta}}_j$ for the jth round.

Finally, we make a brief comment on the communication cost of distributed FONE. First, both the distributed FONE and DC-SGD are transmitting p-dimensional vectors from the local machines, which are usually considered as *communication-efficient* distributed protocols in the literature. More precisely, DC-SGD only has one round communication and thus the communication cost on each local machine is $O(p)$. For distributed FONE, which requires K rounds of communication, the total communication cost on each local machine is $O(Kp)$. However, we note that from Theorems 3.2 and 3.3 in our theoretical results below, the number of rounds K only needs be a constant (instead of diverging to infinity). Therefore, the communication of the distributed FONE is on the same (asymptotic) order as DC-SGD.

Algorithm 3 Distributed FONE for estimating $\boldsymbol{\theta}^*$ in (1)

Input: The total sample size N, the entire data $\{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_N\}$ is distributed into L machines/parts $\{\mathcal{H}_k\}$ for $k = 1, 2, \ldots, L$ with $|\mathcal{H}_k| = n_k$. Initial estimator $\widehat{\boldsymbol{\theta}}_0 \in \mathbb{R}^p$, the batch size m, step-size η. Number of rounds K.

1: **for** each round $j = 1, 2, \ldots, K$ **do**
2: **for** each local machine $k = 1, 2, \ldots, L$ **do**
3: Calculate $\sum_{i\in\mathcal{H}_k} g(\widehat{\boldsymbol{\theta}}_{j-1}, \boldsymbol{\xi}_i)$.
4: **end for**
5: Collect $\sum_{i\in\mathcal{H}_k} g(\widehat{\boldsymbol{\theta}}_{j-1}, \boldsymbol{\xi}_i)$ from each local machine to compute their average:

$$\boldsymbol{a} = \frac{1}{N}\sum_{k=1}^{L}\sum_{i\in\mathcal{H}_k} g(\widehat{\boldsymbol{\theta}}_{j-1}, \boldsymbol{\xi}_i) = \frac{1}{N}\sum_{i=1}^{N} g(\widehat{\boldsymbol{\theta}}_{j-1}, \boldsymbol{\xi}_i). \tag{13}$$

6: Send $\boldsymbol{a}$ to the first machine (the local machine with the largest subsample size).
7: Set $\boldsymbol{z}_0 = \widehat{\boldsymbol{\theta}}_{j-1}$
8: **for** each $t = 1, 2, \ldots, T$ **do**
9: Choose B_t to be m distinct elements uniformly drawn from the data on the first machine $\mathcal{H}_1$.
10: Calculate

$$g_{B_t}(\boldsymbol{z}_{t-1}) = \frac{1}{m}\sum_{i\in B_t} g(\boldsymbol{z}_{t-1}, \boldsymbol{\xi}_i),$$

$$g_{B_t}(\boldsymbol{z}_0) = \frac{1}{m}\sum_{i\in B_t} g(\boldsymbol{z}_0, \boldsymbol{\xi}_i).$$

11: Update

$$\boldsymbol{z}_t = \boldsymbol{z}_{t-1} - \eta\{g_{B_t}(\boldsymbol{z}_{t-1}) - g_{B_t}(\boldsymbol{z}_0) + \boldsymbol{a}\}. \tag{14}$$

12: **end for**
13: Set $\widehat{\boldsymbol{\theta}}_j = \boldsymbol{z}_T$.
14: **end for**
15: **Output:** $\widehat{\boldsymbol{\theta}}_{\mathrm{dis},K} = \widehat{\boldsymbol{\theta}}_K$.

3. Theoretical Results

In this section, we provide theoretical results for mini-batch SGD in the diverging p case, DC-SGD, the newly proposed FONE and its distributed version. We first note that in most cases, the minimizer $\boldsymbol{\theta}^*$ in (1) is also a solution of the following estimating equation:

$$\mathbb{E}g(\boldsymbol{\theta}^*, \boldsymbol{\xi}) = 0, \tag{15}$$

where $g(\boldsymbol{\theta}, \boldsymbol{\xi})$ is the gradient or a subgradient of $f(\boldsymbol{\theta}, \boldsymbol{\xi})$ at $\boldsymbol{\theta}$. We will assume that (15) holds throughout our article. In fact, we can introduce (15) as our basic model (instead of (1)) as in the literature from stochastic approximation (see, e.g., Lai 2003). However, we choose to present the minimization form in (1) as it is more commonly used in statistical learning literature.

Now let us first establish the theory for the DC-SGD approach in the diverging p case.

3.1. Theory for DC-SGD

To establish the theory for DC-SGD, we first state our assumptions. The first assumption is on the relationship among the dimension p, the sample size n, and the mini-batch size m. Recall that α is the decaying rate in the step-size of SGD (see the input of Algorithm 1). (A1). Suppose that $p \to \infty$ and $p = O(n^{\kappa_1})$ for some $0 < \kappa_1 < 1$. The mini-batch size m satisfies $p \log n = o(m)$ and $n^{\tau_1} \le m \le n/p^{1/\alpha+\tau_2}$ for some $0 < \tau_1, \tau_2 < 1$.

Throughout this article, we define a loss function f to be smooth when f is continuously differentiable (e.g., logistic regression), and nonsmooth when f is nondifferentiable $f(\boldsymbol{\theta}, \boldsymbol{\xi})$ (e.g., quantile regression).

We give two separate conditions for smooth and nonsmooth loss functions, respectively. To simplify the illustration, we only present Conditions (A2-Log) and (A2-QR) for the two representative examples of smooth and nonsmooth loss functions. In particular, Condition (A2-Log) applies to logistic regression in Example 2.1, and Condition (A2-QR) applies to the quantile regression in Example 2.2. In Section A.1 in the supplementary materials, we provide very general conditions for smooth and nonsmooth loss functions, which are not necessarily restricted to the regression case. We will also verify that the logistic and quantile regressions satisfy our general conditions.

(A2-Log). (For logistic regression in Example 2.1.) Assume that

$$c_1 \le \lambda_{\min}(\mathbb{E}(\mathcal{X}\mathcal{X}')) \le \lambda_{\max}(\mathbb{E}(\mathcal{X}\mathcal{X}')) \le c_1^{-1},$$

and $\|\boldsymbol{\theta}^*\|_2 \le C_1$, $\sup_{\|\boldsymbol{v}\|_2=1} \mathbb{E}\exp(t_0(\boldsymbol{v}'\mathcal{X})^2) \le C_2$ for some $c_1, t_0, C_1, C_2 > 0$.

(A2-QR). (For quantile regression in Example 2.2.) Assume that $\epsilon|\mathcal{X}$ has density function $\rho_{\mathcal{X}}$ that is bounded and satisfies $|\rho_{\mathcal{X}}(x_1) - \rho_{\mathcal{X}}(x_2)| \le C_1|x_1 - x_2|$ for some $C_1 > 0$. Moreover,

$$c_1 \le \lambda_{\min}(\mathbb{E}[\mathcal{X}\mathcal{X}'\rho_{\mathcal{X}}(0)]) \le \lambda_{\max}(\mathbb{E}[\mathcal{X}\mathcal{X}'\rho_{\mathcal{X}}(0)]) \le c_1^{-1}$$

and $\sup_{\|\boldsymbol{v}\|_2=1} \mathbb{E}\exp(t_0|\boldsymbol{v}'\mathcal{X}|) \le C_2$ for some $c_1, t_0, C_2 > 0$.

Conditions (A2-Log) and (A2-QR) are standard regularity conditions (i.e., covariance and moment conditions) on the covariates of a regression model. The minimum eigenvalue conditions ensure that the population risk $F(\boldsymbol{\theta})$ is locally strongly convex at $\boldsymbol{\theta} = \boldsymbol{\theta}^*$. We note that the general version of these two conditions (A2-Log) and (A2-QR) for arbitrary loss functions $f(\boldsymbol{\theta}, \boldsymbol{\xi})$ will be provided in Section A.1 in the supplementary materials.

Due to the space constraint, we introduce the theory of mini-batch SGD in Section B of the supplementary materials. Despite the simplicity and wide applicability of the mini-batch SGD, the theoretical investigation of the asymptotic properties of this approach, especially in the diverging p case, is still quite limited. In fact, our theoretical analysis reveals several interesting phenomena of the mini-batch SGD when p is diverging, which also leads to useful practical guidelines when implementing mini-batch SGD. A natural starting point in a standard mini-batch SGD is random initialization. However, we show that when p diverges to infinity, a random initialized SGD will no longer converge to $\boldsymbol{\theta}^*$, with the L_2-estimation error being a polynomial of p (see Proposition B.2 in the supplementary materials). To address the challenge arising from $p \to \infty$, a consistent initial estimator $\widehat{\boldsymbol{\theta}}_0$ is both sufficient and necessary to ensure the convergence of SGD (see Theorem B.1 and Proposition B.2 in the supplementary materials). Since DC-SGD is built on the mini-batch SGD, a consistent initialization is also required in DC-SGD, which can be easily constructed by running a deterministic optimization on a small batch of data. Given that, we provide the convergence result of the DC-SGD estimator $\widehat{\boldsymbol{\theta}}_{\mathrm{DC}}$ in (7) (see Algorithm 1). For the ease of presentation, we assume that the data are evenly distributed, where each local machine has $n = N/L$ samples.

Theorem 3.1. Assume Conditions (A1) and (A2-Log) or (A2-QR) hold, suppose the initial estimator $\widehat{\boldsymbol{\theta}}_0$ is independent to $\{\boldsymbol{\xi}_i, i = 1, 2, \ldots, N\}$. On the event $\{\|\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\theta}^*\|_2 \le d_n\}$ with $d_n \to 0$, the DC-SGD estimator achieves the following convergence rate:

$$\mathbb{E}_0\|\widehat{\boldsymbol{\theta}}_{\mathrm{DC}} - \boldsymbol{\theta}^*\|_2^2 = O\left(\frac{p}{L^{1-\alpha}m^{1-\alpha}N^{\alpha}} + \frac{p^2L^{2\alpha}}{m^{2-2\alpha}N^{2\alpha}}\right). \quad (16)$$

Again, we note that throughout the theoretical results, Condition (A2-Log) can be generalized to Conditions (C2) and (C3) in the supplementary materials (and correspondingly (A2-QR) to (C2) and (C3*)).

The convergence rate in (16) contains two terms. The first term comes from the variance of the DC-SGD estimator, while the second one comes from the squared bias term. Note that $n = N/L$, the squared bias term in (16) can be written as $\left(\frac{p}{m^{1-\alpha}n^{\alpha}}\right)^2$, which is the same as the square of the bias from the mini-batch SGD on one machine (see Theorem B.1 in the supplementary materials). This is because the averaging of the local estimators from L machines cannot reduce the bias term. On the other hand, the variance term is reduced by a factor of $1/L$ by averaging over L machines. Therefore, when L is not too large, the variance will become the dominating term and gives the optimal convergence rate. An upper bound on L is a universal condition in the DC scheme to achieve the optimal rate in a statistical estimation problem (see, e.g., Li, Lin, and Li 2013; Chen and Xie 2014; Zhang, Duchi, and Wainwright 2015; Zhao, Cheng, and Liu 2016; Lee et al. 2017; Battey et al. 2018; Huang and Huo 2019; Volgushev, Chao, and Cheng 2019). In particular, let us consider the optimal step-size r_i where $\alpha = 1$. When the number of machines $L = O(\sqrt{N/p})$, the rate in (16) becomes $O(p/N)$, which is a classical optimal rate when using all the N samples.

We next show on the two motivating examples that the constraint on the number of machines $L = O(\sqrt{N/p})$ is necessary to achieve the optimal rate by DC-SGD. To this end, we provide the lower bounds on our two examples for the bias of the SGD estimator on each local machine.

Example 2.1 (Continued). For a logistic regression model with $\boldsymbol{\xi} = (Y, \mathcal{X})$, let $\mathcal{X} = (1, X_1, \ldots, X_{p-1})'$ with $\mathbb{E}X_i = 0$ for all $1 \le i \le p-1$ and $\boldsymbol{\theta}^* = (1, 0, \ldots, 0)$. Suppose that $\mathbb{E}\|\mathcal{X}\|_2^2 \ge cp$ for some $c > 0$ and $\sup_{\|\boldsymbol{v}\|_2=1} \mathbb{E}\exp(t_0|\boldsymbol{v}'\mathcal{X}|) \le C$. Suppose the initial estimator $\widehat{\boldsymbol{\theta}}_0$ is independent to $\{\mathcal{X}_i, i = 1, 2, \ldots, n\}$. On the event $\{\|\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\theta}^*\|_2 \le d_n\}$ with $d_n \to 0$, we have $\|\mathbb{E}_0(\widehat{\boldsymbol{\theta}}_{\mathrm{SGD}}) - \boldsymbol{\theta}^*\|_2 \ge \frac{cp}{m^{1-\alpha}n^{\alpha}}$.

Example 2.2 (Continued). For a quantile regression model, assume that ϵ is independent with $\mathcal{X}$ and $\mathbb{E}X_i = 0$ for all $1 \le i \le p-1$. Let $F(x)$ be the cumulative distribution function of ϵ. Suppose that $\mathbb{E}\|\mathcal{X}\|_2^2 \ge cp$ for some $c > 0$ and $\sup_{\|\boldsymbol{v}\|_2=1} \mathbb{E}\exp(t_0|\boldsymbol{v}'\mathcal{X}|) \le C$. Suppose the initial estimator $\widehat{\boldsymbol{\theta}}_0$ is independent to $\{\mathcal{X}_i, i = 1, 2, \ldots, n\}$, and assume that $F(\cdot)$ has bounded third-order derivatives and $F'(0)$, $F''(0)$ are positive. On the event $\{\|\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\theta}^*\|_2 \le d_n\}$ with $d_n \to 0$, we have $\|\mathbb{E}_0(\widehat{\boldsymbol{\theta}}_{\text{SGD}}) - \boldsymbol{\theta}^*\|_2 \ge \frac{cp}{m^{1-\alpha}n^{\alpha}}$.

For the DC-SGD estimator $\widehat{\boldsymbol{\theta}}_{\text{DC}}$, it is easy to see that the mean squared error $\mathbb{E}_0\|\widehat{\boldsymbol{\theta}}_{\text{DC}} - \boldsymbol{\theta}^*\|_2^2 \ge \|\mathbb{E}_0(\widehat{\boldsymbol{\theta}}_{\text{DC}}) - \boldsymbol{\theta}^*\|_2^2$ (the squared bias of $\widehat{\boldsymbol{\theta}}_{\text{DC}}$). Recall that the bias of $\widehat{\boldsymbol{\theta}}_{\text{DC}}$ is the average over local machines, and each local machine induces the same bias $\|\mathbb{E}_0(\widehat{\boldsymbol{\theta}}_{\text{SGD}}) - \boldsymbol{\theta}^*\|_2$ (see the bias in the above two examples). Therefore, for logistic regression and quantile regression, when $\alpha = 1$ and $\sqrt{N/p} = o(L)$, we have

$$\frac{\mathbb{E}_0\|\widehat{\boldsymbol{\theta}}_{\text{DC}} - \boldsymbol{\theta}^*\|_2^2}{p/N} \ge \frac{\|\mathbb{E}_0(\widehat{\boldsymbol{\theta}}_{\text{SGD}}) - \boldsymbol{\theta}^*\|_2^2}{p/N} \ge \frac{c^2p^2/n^2}{p/N} = c^2\frac{L^2}{N/p} \to \infty.$$

This shows that when the number of machines L is much larger than $\sqrt{N/p}$, the convergence rate of DC-SGD will no longer be optimal.

Remark 3.1. It is worthwhile noting that convergence rate of stochastic gradient estimators can be improved by the use of averaging (Polyak and Juditsky 1992). In particular, given the SGD iterates $\{\boldsymbol{z}_i\}_{i=1}^s$ in (6), an average stochastic gradient (ASGD) algorithm outputs $\widehat{\boldsymbol{\theta}}_{\text{ASGD}} = \frac{1}{s}\sum_{i=1}^s \boldsymbol{z}_i$ instead of $\widehat{\boldsymbol{\theta}}_{\text{SGD}} = \boldsymbol{z}_s$. ASGD is known to achieve a faster convergence rate than SGD when $\alpha < 1$. Similar to $\widehat{\boldsymbol{\theta}}_{\text{DC}}$ in (7), we may implement the DC scheme on ASGD estimator, denoted by $\frac{1}{L}\sum_{k=1}^{L}\widehat{\boldsymbol{\theta}}_{\text{ASGD}}^{(k)}$. Assuming the mini-batch size $m = 1$, we have

$$\mathbb{E}_0\|\widehat{\boldsymbol{\theta}}_{\text{DC-ASGD}} - \boldsymbol{\theta}^*\|_2^2 = O\left(\frac{p}{N} + \frac{p^2L^2}{N^2}\right).$$

As compared to the convergence rate of DC-SGD in Theorem 3.1, when the exponent in the stepsize $\alpha < 1$, the convergence rate of DC-ASGD is faster than the that of DC-SGD. In other words, the averaging idea indeed accelerates the DC-SGD approach. Nevertheless, the DC-ASGD estimator requires the same condition as $\widehat{\boldsymbol{\theta}}_{\text{DC}}$ on the number of machines (i.e., $L = O(\sqrt{N/p})$) to achieve optimal rate.

3.2. Theory for FONE

We provide our main theoretical results on FONE for estimating $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$ and the distributed FONE for estimating $\boldsymbol{\theta}^*$. The smooth loss and nonsmooth loss functions are discussed separately in Sections 3.2.1 and 3.2.2.

Recall that n denotes the sample size used in FONE in the single machine setting (see Algorithm 2). In our theoretical results, we denote the step-size in FONE by η_n (instead of η in Algorithms 2 and 3) to highlight the dependence of the step-size on n. For the FONE method, Condition `(A1)` can be further weakened to the following condition:

`(A1*)` . Suppose that $p \to \infty$ and $p = O(n^{\kappa_1})$ for some $0 < \kappa_1 < 1$. The mini-batch size m satisfies $p\log n = o(m)$ with $m = O(n^{\kappa_2})$ for some $0 < \kappa_2 < 1$.

3.2.1. Smooth Loss Function f

To establish the convergence rate of our distributed FONE, we first provide a consistency result for $\widehat{\boldsymbol{\theta}}_{\text{FONE}}$ in (12).

Proposition 3.1 (On $\widehat{\boldsymbol{\theta}}_{\text{FONE}}$ for $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$ for smooth loss function f). Assume Conditions `(A1*)` and `(A2-Log)` (or Conditions `(C2)` and `(C3)` in the supplementary materials) hold. Suppose that the initial estimator satisfies $\|\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\theta}^*\|_2 = O_{\mathbb{P}}(d_n)$, and $\|\boldsymbol{a}\|_2 = O(\tau_n)$ (or $O_{\mathbb{P}}(\tau_n)$ for the random case). The iteration number T and step-size η_n satisfy $\log n = o(\eta_n T)$ and $T = O(n^A)$ for some $A > 0$. We have

$$\|\widehat{\boldsymbol{\theta}}_{\text{FONE}} - \boldsymbol{\Sigma}^{-1}\boldsymbol{a}\|_2 = O_{\mathbb{P}}\Big(\tau_n d_n + \tau_n^2 + \sqrt{\frac{p\log n}{n}}\tau_n + \sqrt{\eta_n}\tau_n + n^{-\gamma}\Big) \tag{17}$$

for any large $\gamma > 0$.

The relationship between η_n and T (i.e., $\log n = o(\eta_n T)$) is intuitive since when the step-size η_n is small, Algorithm 2 requires more iterations T to converge. The consistency of the estimator requires that the length of the vector $\boldsymbol{a}$ goes to zero, that is, $\tau_n = o(1)$, since τ_n^2 appears in the convergence rate in (17). In Section 4, we further discuss a slightly modified FONE that deals with any vector $\boldsymbol{a}$, which applies to the estimation of the limiting variance of $\widehat{\boldsymbol{\theta}}$ in (3). When $\tau_n = o(1)$, $d_n = o(1)$, and $\eta_n = o(1)$, each term in $O_{\mathbb{P}}$ in (17) goes to zero and thus the proposition guarantees that $\widehat{\boldsymbol{\theta}}_{\text{FONE}}$ is a consistent estimator of $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$. Moreover, since Proposition 3.1 will be used as an intermediate step for establishing the convergence rate of the distributed FONE, to facilitate the ease of use of Proposition 3.1, we leave d_n, τ_n, and η_n unspecified here and discuss their magnitudes in Theorem 3.2. A practical choice of η_n is further discussed in the experimental section.

Given Proposition 3.1, we now provide the convergence result for the multi-round distributed FONE for estimating $\boldsymbol{\theta}^*$ and approximating $\widehat{\boldsymbol{\theta}}$ in Algorithm 3. To this end, let us first provide some intuitions on the improvement for one-round distributed FONE from the initial estimator $\widehat{\boldsymbol{\theta}}_0$ to $\widehat{\boldsymbol{\theta}}_{\text{dis},1}$. For the first round in Algorithm 3, the algorithm essentially estimates $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$ with $\boldsymbol{a} = \frac{1}{N}\sum_{i=1}^N g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i)$. When $f(\boldsymbol{\theta}, \boldsymbol{\xi})$ is differentiable and noting that $\frac{1}{N}\sum_{i=1}^N g(\widehat{\boldsymbol{\theta}}, \boldsymbol{\xi}_i) = 0$ (where $\widehat{\boldsymbol{\theta}}$ is the ERM in (2)), we can prove that (see more details in the proof of Theorem 3.2),

$$\begin{aligned}\boldsymbol{a} &= \frac{1}{N}\sum_{i=1}^N \Big(g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i) - g(\widehat{\boldsymbol{\theta}}, \boldsymbol{\xi}_i)\Big) \\ &= G(\widehat{\boldsymbol{\theta}}_0) - G(\widehat{\boldsymbol{\theta}}) + \frac{1}{N}\sum_{i=1}^N \Big\{[g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i) - g(\widehat{\boldsymbol{\theta}}, \boldsymbol{\xi}_i)] \\ &\quad - [G(\widehat{\boldsymbol{\theta}}_0) - G(\widehat{\boldsymbol{\theta}})]\Big\}\end{aligned} \tag{18}$$

$$= \boldsymbol{\Sigma}(\widehat{\boldsymbol{\theta}}_0 - \widehat{\boldsymbol{\theta}}) + O_{\mathbb{P}}(1)\left(\|\widehat{\boldsymbol{\theta}}_0 - \widehat{\boldsymbol{\theta}}\|_2\|\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\theta}^*\|_2 + \|\widehat{\boldsymbol{\theta}}_0 - \widehat{\boldsymbol{\theta}}\|_2^2\right) \tag{19}$$

$$+ O_{\mathbb{P}}(1)\left(\sqrt{\frac{p\log N}{N}}\|\widehat{\boldsymbol{\theta}}_0 - \widehat{\boldsymbol{\theta}}\|_2 + N^{-\gamma}\right), \tag{20}$$

for any $\gamma > 0$. Note that in Algorithm 3, the FONE procedure is executed on the first machine. For the ease of plugging the result in Proposition 3.1 on FONE, we let $n := n_1$ to denote the subsample size on the first machine.

Assuming that the initial estimator $\widehat{\boldsymbol{\theta}}_0$ and $\widehat{\boldsymbol{\theta}}$ satisfy $\|\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\theta}^*\|_2 + \|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2 = O_{\mathbb{P}}(n^{-\delta_1})$ for some $\delta_1 > 0$, then by (18), we have $\|\boldsymbol{a}\|_2 = O_{\mathbb{P}}(n^{-\delta_1})$ (i.e., the length $\tau_n = O(n^{-\delta_1})$ in Proposition 3.1). Moreover, we can further choose d_n in Proposition 3.1 to be $d_n = O(n^{-\delta_1})$. Let the step-size $\eta_n = n^{-\delta_2}$ for some $\delta_2 > 0$. After one round of distributed FONE in Algorithm 3, by Proposition 3.1 and the proof of Theorem 3.2 in the supplementary materials, we can obtain that $\|\widehat{\boldsymbol{\theta}}_{\text{dis},K=1} - \widehat{\boldsymbol{\theta}}\|_2 = O_{\mathbb{P}}(n^{-\delta_1-\delta_0})$ with $\delta_0 = \min(\delta_1, \delta_2/2, (1-\kappa_1)/2)$, where κ_1 is the parameter in our assumption $p = O(n^{\kappa_1})$ (see Condition (A1*)). Now we show the convergence rate of the Kth round distributed FONE by induction. In the Kth round, the output of the $(K-1)$th round $\widehat{\boldsymbol{\theta}}_{\text{dis},K-1}$ is used as the initial estimator $\widehat{\boldsymbol{\theta}}_0$ where $\|\widehat{\boldsymbol{\theta}}_0 - \widehat{\boldsymbol{\theta}}\|_2 = n^{-\delta_1-(K-1)\delta_0}$. Therefore, we can choose d_n and τ_n in Proposition 3.1 by $d_n = O(n^{-\delta_1})$ and $\tau_n = n^{-\delta_1-(K-1)\delta_0}$ from (18). As a result of Proposition 3.1, we obtain that $\|\widehat{\boldsymbol{\theta}}_{\text{dis},K} - \widehat{\boldsymbol{\theta}}\|_2 = O_{\mathbb{P}}(n^{-\delta_1-K\delta_0})$. This convergence result of distributed FONE is formally stated in the next theorem.

Theorem 3.2 (Distributed FONE for smooth loss function f). Assume Conditions (A1*) and (A2-Log) (or Conditions (C2) and (C3) in the supplementary materials) hold, $N = O(n^A)$ for some $A > 0$. Suppose that $\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2 + \|\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\theta}^*\|_2 = O_{\mathbb{P}}(n^{-\delta_1})$ for some $\delta_1 > 0$. Let $\eta_n = n^{-\delta_2}$ for some $\delta_2 > 0$, $\log n = o(\eta_n T)$, $T = O(n^A)$ for some $A > 0$, and $p\log n = o(m)$. For any $\gamma > 0$, there exists $K_0 > 0$ such that, for any $K \geq K_0$, we have $\|\widehat{\boldsymbol{\theta}}_{\text{dis},K} - \widehat{\boldsymbol{\theta}}\|_2 = O_{\mathbb{P}}(n^{-\gamma})$.

As we illustrate before Theorem 3.2, since $\|\widehat{\boldsymbol{\theta}}_{\text{dis},K} - \widehat{\boldsymbol{\theta}}\|_2 = O_{\mathbb{P}}(n^{-\delta_1-K\delta_0})$ with $\delta_0 = \min(\delta_1, \delta_2/2, (1-\kappa_1)/2)$. We have $K_0 = (\gamma - \delta_1)/\delta_0$ in Theorem 3.2. We recall that $n = n_1$ denotes the number of samples on the first machine. Note that γ in Theorem 3.2 can be arbitrarily large. Under some regular conditions, it is typical that $\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2 = O_{\mathbb{P}}(\sqrt{p/N})$. Therefore, for a smooth loss function f, our distributed FONE achieves the optimal rate $O_{\mathbb{P}}(\sqrt{p/N})$. Note that it does not need any condition on the number of machines L. Given the step-size $\eta_n = n^{-\delta_2}$, by the condition $\log n = o(\eta_n T)$, we can choose the number of iterations $T = n^{\delta_2}(\log n)^2$ in the distributed FONE. Therefore, the computation complexity of distributed FONE is $O(np + n^{\delta_2}(\log n)^2 mp)$ for each round, on the first machine. We also note that n is the subsample size on the first machine, which is much smaller than the total sample size N. In terms of the communication cost, each machine only requires to transmit an $O(p)$ vector for each round.

We note that although DC-SGD assumes the independence between the initial estimator and the sample, such a condition is no longer required in our distributed FONE for both smooth loss and nonsmooth loss. Therefore, one can use the subsample on one local machine to construct the initial estimator. We also note that, in contrast to DC-SGD, it is unknown how the averaging scheme would benefit the Newton approach. Therefore, as compared to DC-ASGD in Remark 3.1, it is unclear whether the use of averaging in Dis-FONE could improve the convergence rate. We leave it to future investigation.

3.2.2. Nonsmooth Loss Function f

For a nonsmooth loss, we provide the following convergence rate of the FONE of $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$ under Conditions (A1*) and (A2-QR).

Proposition 3.2 (On $\widehat{\boldsymbol{\theta}}_{\text{FONE}}$ for $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$ for nonsmooth loss function f). Assume Conditions (A1*) and (A2-QR) (or Conditions (C2) and (C3*) in the supplementary materials) hold. Suppose that the initial estimator satisfies $\|\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\theta}^*\|_2 = O_{\mathbb{P}}(d_n)$, and $\|\boldsymbol{a}\|_2 = O(\tau_n)$ (or $O_{\mathbb{P}}(\tau_n)$ for the random case). The iteration number T and step-size η_n satisfy $\log n = o(\eta_n T)$ and $T = O(n^A)$ for some $A > 0$. We have

$$\|\widehat{\boldsymbol{\theta}}_{\text{FONE}} - \boldsymbol{\Sigma}^{-1}\boldsymbol{a}\|_2 = O_{\mathbb{P}}\left(\tau_n d_n + \tau_n^2 + \sqrt{\frac{p\log n}{n}}\sqrt{\tau_n} + \frac{p\log n}{m}\sqrt{\eta_n} + \sqrt{\eta_n}\tau_n + \frac{p\log n}{n}\right). \tag{21}$$

Compared to Proposition 3.1, the mini-batch size m appears in the error bound of Proposition 3.2 due to the discontinuity of the gradient in the nonsmooth setting. Consequently, the average gradient $\frac{1}{m}\sum_{i=1}^m g_i(\boldsymbol{\theta}, \boldsymbol{\xi}_i)$ has a nonnegligible bias that enters into the error bound.

With Proposition 3.2 in hand, we now provide the convergence rate of the distributed FONE in Algorithm 3 under Condition (A2-QR). It is worthwhile noting that when $f(\boldsymbol{\theta}, \boldsymbol{\xi})$ is nondifferentiable, then $\frac{1}{N}\sum_{i=1}^N g(\widehat{\boldsymbol{\theta}}, \boldsymbol{\xi}_i)$ can be nonzero due to the discontinuity in the function $g(\boldsymbol{\theta}, \boldsymbol{\xi})$, where $\widehat{\boldsymbol{\theta}}$ is the ERM in (2). Therefore, we need to assume that

$$\sum_{i=1}^N g(\widehat{\boldsymbol{\theta}}, \boldsymbol{\xi}_i) = O_{\mathbb{P}}(q_N) \tag{22}$$

with $q_N = O(N^{\kappa_3})$ for some $0 < \kappa_3 < 1$. For example, for a quantile regression, $q_N = O(p^{3/2}\log N)$ (He and Shao 2000), which satisfies this condition when $p = o(N^{\kappa_4})$ with $0 < \kappa_4 < 2/3$.

Given (A1*), (A2-QR) and (22), we have the following convergence rate of $\widehat{\boldsymbol{\theta}}_{\text{dis},K}$:

Theorem 3.3 (Distributed FONE for nonsmooth loss function f). Suppose Conditions (A1*) and (A2-QR) (or Conditions (C2) and (C3*) in the supplementary materials) and (22) hold, $N = O(n^A)$ and $T = O(n^A)$ for some $A > 0$. Suppose that $\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2 + \|\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\theta}^*\|_2 = O_{\mathbb{P}}(n^{-\delta_1})$ for some $\delta_1 > 0$. Let $\eta_n = n^{-\delta_2}$ for some $\delta_2 > 0$, $\log n = o(\eta_n T)$, and $p\log n = o(m)$. For any $\frac{1}{2} < \gamma < 1$, there exists $K_0 > 0$ such that, for any $K \geq K_0$, we have

$$\|\widehat{\boldsymbol{\theta}}_{\text{dis},K} - \widehat{\boldsymbol{\theta}}\|_2 = O_{\mathbb{P}}\left(\frac{q_N}{N} + \sqrt{\eta_n}\frac{p\log n}{m} + \left(\frac{p\log n}{n}\right)^{\gamma}\right). \tag{23}$$

As one can see from (23), the distributed FONE has a faster convergence rate when the subsample size on the first machine

n_1 is large (recall that $n := n_1$). In practice, it is usually affordable to increase the memory and computational resources for only one local machine. This is different from the case of DC-SGD, where the convergence rate actually depends on the smallest subsample size among local machines.[2]

The parameter γ in the exponent of the last term serves as a target rate of convergence. More specifically, the convergence rate after K rounds is (see Section D.4 for more details),

$$\|\widehat{\boldsymbol{\theta}}_{\mathrm{dis},K} - \widehat{\boldsymbol{\theta}}\|_2 = O_{\mathbb{P}}\Big(n^{-\delta_1 - K\delta_0} + \frac{q_N}{N} + \sqrt{\eta_n}\frac{p\log n}{m} + \Big(\frac{p\log n}{n}\Big)^{\gamma}\Big), \quad \text{where } \delta_0 = \min\left\{\frac{\delta_1(1-\gamma)}{2\gamma-1}, \frac{\delta_1}{2}, \frac{\delta_2}{4}\right\}.$$

Therefore, when $K > K_0 := (\kappa_1\gamma - \delta_1)/\delta_0$, the term $n^{-\delta_1 - K\delta_0}$ is bounded by the term of in the convergence rate $\big(\frac{p\log n}{n}\big)^{\gamma}$, where κ_1 is the parameter in the assumption of $p = O(n^{\kappa_1})$ in Condition (A1*). For the second last term in the right-hand side of (23), we can choose the step-size η_n and the batch size m such that $\sqrt{\eta_n}/m \le (p\log n)^{\gamma-1}/n^{\gamma}$, and the convergence rate of $\|\widehat{\boldsymbol{\theta}}_{\mathrm{dis},K} - \widehat{\boldsymbol{\theta}}\|_2$ is thus dominated by $q_N/N + \big((p\log n)/n\big)^{\gamma}$. Due to q_N in (22), the relationship between these two terms depends on the specific model. Usually, under some conditions on the dimension p, $\|\widehat{\boldsymbol{\theta}}_{\mathrm{dis},K} - \widehat{\boldsymbol{\theta}}\|_2$ achieves a faster rate than $\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2$, which makes $\widehat{\boldsymbol{\theta}}_{\mathrm{dis},K}$ attain the optimal rate for estimating $\boldsymbol{\theta}^*$. Let us take the quantile regression as an example, where the ERM $\widehat{\boldsymbol{\theta}}$ has an error rate of $\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2 = O_{\mathbb{P}}(\sqrt{p/N})$ and $q_N = O(p^{3/2}\log N)$ (He and Shao 2000). Assuming that $p = O(\sqrt{N}/\log N)$ and $n \ge cN^{\frac{1}{2\gamma}}p^{1-\frac{1}{2\gamma}}\log n$, we have $\|\widehat{\boldsymbol{\theta}}_{\mathrm{dis},K} - \boldsymbol{\theta}^*\|_2 = O_{\mathbb{P}}(\sqrt{p/N})$.

Similar to the smooth case, the computation complexity is $O(np + n^{\delta_2}(\log n)^2 mp)$ for each round, on the first machine. Assuming the second term of (23) is dominated by the third term, we may specify $m = \sqrt{\eta_n}n\log n$ and the corresponding computation complexity becomes $O(n^{1+\delta_2/2}(\log n)^3 p)$. Again, each machine only requires to transmit an $O(p)$ vector for each round.

4. Inference: Application of FONE to the Estimation of $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ With $\|\boldsymbol{w}\|_2 = 1$

An important application of the proposed FONE is to conduct the inference of $\boldsymbol{\theta}^*$ in the diverging p case. To provide asymptotic valid inference, we only need a consistent estimator of the limiting variance in (3).

To estimate the limiting variance, we note that $\boldsymbol{A}$ can be easily estimated by $\widehat{\boldsymbol{A}} = \frac{1}{n}\sum_{i=1}^{n} g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i)g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i)'$. Therefore, we only need to estimate $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$. The challenge here is the theory of our Propositions 3.1 and 3.2 only applies to the case $\boldsymbol{\Sigma}^{-1}\boldsymbol{a}$ with $\|\boldsymbol{a}\|_2 = o(1)$ or $o_{\mathbb{P}}(1)$. However, in the inference application, we have $\|\boldsymbol{w}\|_2 = 1$. To address this challenge, given the unit length vector $\boldsymbol{w}$, we define $\boldsymbol{a} = \tau_n\boldsymbol{w}$, where $\|\boldsymbol{a}\|_2 = \tau_n = o(1)$ and its rate will be specified later in our theoretical results in Theorems 4.1 and 4.2. We run Algorithm 2 and its output $\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{z}_T$ is an estimator of $\tau_n\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$. Then the estimator of $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ can be naturally constructed as,

$$\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}} = \frac{\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{z}_T}{\tau_n}, \quad \text{where in Algorithm 2} \quad \boldsymbol{a} = \tau_n\boldsymbol{w}. \tag{24}$$

We note that the initial estimator $\widehat{\boldsymbol{\theta}}_0$ for estimating $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ needs to be close to the target parameter $\boldsymbol{\theta}^*$. In a nondistributed setting, we could choose the ERM $\widehat{\boldsymbol{\theta}}$ as $\widehat{\boldsymbol{\theta}}_0$ for inference, while in the distributed setting, we use $\widehat{\boldsymbol{\theta}}_{\mathrm{dis},K}$ from distributed FONE in Algorithm 3 with a sufficiently large K.

Furthermore, we briefly comment on an efficient implementation for computing the limiting variance $\boldsymbol{w}'\boldsymbol{\Sigma}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$. Instead of explicitly constructing the estimator of $\boldsymbol{A}$ by a $p \times p$ matrix $\widehat{\boldsymbol{A}} = \frac{1}{n}\sum_{i=1}^{n} g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i)g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i)'$, we can directly compute the estimator by

$$(\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}})'\widehat{\boldsymbol{A}}(\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}}) = \frac{1}{n}\sum_{i=1}^{n}\Big(g(\widehat{\boldsymbol{\theta}}_0, \boldsymbol{\xi}_i)'\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}}\Big)^2, \tag{25}$$

where $\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}}$ is precomputed by FONE. The implementation in (25) only incurs a computation cost of $O(np)$.

We next provide the theoretical results of the estimator in (24) for two cases: f is smooth and f is nonsmooth. We note that for the purpose of asymptotic valid inference, we only need $\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}}$ in (24) to be a consistent estimator of $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$. To show the consistency of our estimator, we provide the convergence rates in the following Theorems 4.1 and 4.2 for smooth and nonsmooth loss functions, respectively:

Theorem 4.1 (Estimating $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ for a smooth loss function f). Under the conditions of Proposition 3.1, let $\tau_n = \sqrt{(p\log n)/n}$. Assuming that $\|\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\theta}^*\|_2 = O_{\mathbb{P}}(d_n)$ and $\log n = o(\eta_n T)$, we have

$$\|\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}} - \boldsymbol{\Sigma}^{-1}\boldsymbol{w}\|_2 = O_{\mathbb{P}}\Big(\sqrt{\frac{p\log n}{n}} + \sqrt{\eta_n} + d_n\Big). \tag{26}$$

From Theorem 4.1, the estimator $\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}}$ is consistent as long as $d_n = o(1)$ and the step-size $\eta_n = o(1)$. Let us further provide some discussion on the convergence rate in (26). If we choose a good initiation such that $d_n = O(\sqrt{(p\log n)/n})$, the term d_n in (26) will be a smaller order term. For example, the initialization rate $d_n = O(\sqrt{(p\log n)/n})$ can be easily satisfied by using either the ERM $\widehat{\boldsymbol{\theta}}$ or $\widehat{\boldsymbol{\theta}}_{\mathrm{dis},K}$ from distributed FONE with a sufficiently large K. Moreover, we can specify η_n to be small (e.g., $\eta_n = O((p\log n)/n)$). Then the rate in (26) is $\sqrt{(p\log n)/n}$, which almost matches the parametric rate for estimating a p dimensional vector.

For nonsmooth loss function f, we have the following convergence rate of $\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}}$:

Theorem 4.2 (Estimating $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ for nonsmooth loss function f). Under the conditions of Proposition 3.2, let $\tau_n = \big((p\log n)/n\big)^{1/3}$. Assuming that $\|\widehat{\boldsymbol{\theta}}_0 - \boldsymbol{\theta}^*\|_2 = O_{\mathbb{P}}(d_n)$ and

[2] Noting that although we present the evenly distributed setting for DC-SGD for the ease of illustration, one can easily see the convergence rate is actually determined by the smallest subsample size from the proof.

$\log n = o(\eta_n T)$, we have

$$\|\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}} - \boldsymbol{\Sigma}^{-1}\boldsymbol{w}\|_2 = O_{\mathbb{P}}\Big(\big(\frac{p\log n}{n}\big)^{1/3} + \sqrt{\eta_n}\big(\frac{n^{1/3}(p\log n)^{2/3}}{m} + 1\big) + d_n\Big). \tag{27}$$

To make d_n a smaller order term in the rate in (27), we choose a good initiation such that $d_n = O((p\log n)/n)^{1/3})$. As long as the step-size η_n is small such that $\eta_n = \min\big(\frac{(p\log n)^{2/3}}{n^{2/3}}, \frac{m^2}{(p\log n)^{2/3}n^{4/3}}\big)$, the convergence rate in (27) is $O_{\mathbb{P}}\big(((p\log n)/n)^{1/3}\big)$, which implies that $\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}}$ is a consistent estimator of $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$.

5. Experimental Results

In this section, we provide simulation studies and real data analysis to illustrate the performance of our methods on two statistical estimation problems in Examples 2.1 and 2.2, that is, logistic regression and quantile regression (QR).

5.1. Simulation Studies

For regression problems in the two motivating examples, let $\boldsymbol{\xi}_i = (Y_i, \mathcal{X}_i)$ for $i = 1, 2, \ldots, N$, where $\mathcal{X}_i = (1, X_{i,1}, X_{i,2}, \ldots, X_{i,p-1})' \in \mathbb{R}^p$ is a random covariate vector and N is the total sample size. Here $(X_{i,1}, X_{i,2}, \ldots, X_{i,p-1})$ follows a multivariate normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I}_{p-1})$, where $\mathbf{I}_{p-1}$ is a $p-1$ dimensional identity matrix. We also provide the simulation studies with correlated design $\mathcal{X}$ as well as the cases that the samples on local machines are nonidentically distributed, which are relegated to the supplementary materials due to space limitations (see Section G.1). The true coefficient $\boldsymbol{\theta}^*$ follows a uniform distribution $\text{Unif}([-0.5, 0.5]^p)$. For QR in Example 2.2, we follow the standard approach (see, e.g., Pang, Lu, and Wang 2012) that first generates the data from a linear regression model $Y_i = \mathcal{X}_i'\boldsymbol{\theta} + \epsilon_i$, where $\boldsymbol{\theta}$ follows a uniform distribution $\text{Unif}([-0.5, 0.5]^p)$ and $\epsilon_i \sim N(0, 1)$. For each quantile level τ, we need to compute the true QR coefficient $\boldsymbol{\theta}^*$ by shifting ϵ_i such that $\Pr(\epsilon_i \le 0) = \tau$. Thus, the true QR coefficient $\boldsymbol{\theta}^* = \boldsymbol{\theta} + (\Phi^{-1}(\tau), 0, 0, \ldots, 0)'$, where Φ is the CDF of the standard normal distribution. In our experiment, we set the quantile level $\tau = 0.25$. All of the data points are evenly distributed on L machines with subsample size $n = n_i = N/L$ for $i = 1, 2, \ldots, L$. We further discuss the imbalanced situation in Section 5.1.4.

In the following experiments, we evaluate the proposed distributed FONE (Dis-FONE, see Algorithm 3) in terms of the L_2-estimation errors, and compare its performance with that of the DC-SGD estimator (see Algorithm 1). In particular, we report the L_2-distance to the true coefficient $\boldsymbol{\theta}^*$ as well as the L_2-distance to the ERM $\widehat{\boldsymbol{\theta}}$ in (2), which is considered as the nondistributed "oracle" estimator. We also compare the methods with mini-batch SGD in (6) on the entire dataset in a nondistributed setting, which can be considered as a special case of DC-SGD when the number of machines $L = 1$. For all these methods, it is required to provide a consistent initial estimator $\widehat{\boldsymbol{\theta}}_0$. In our experiments below, we compute the initial estimator by minimizing the empirical risk function in (2) with a small batch of fresh samples (which is also used by ERM for the fair comparison). It is clear that as dimension p grows, it requires more samples to achieve the desired accuracy of the initial estimator. Therefore, we specify the size of the fresh samples as $n_0 = 10p$. We note that the fresh samples are used only because DC-SGD requires the independence between the initial estimator and the samples. This is not a requirement for our distributed FONE method. Also, although we allow p to diverge, the sample size $10p$ is still considered as a small batch of samples since the condition `(A1)` requires $p = o(n)$, that is, p grows much slower than n. We also discuss the effect of the accuracy of the initial estimator $\widehat{\boldsymbol{\theta}}_0$ by varying n_0 (see Section G.2 in the supplementary materials).

For DC-SGD, the step-size is set to $r_i = c_0/\max(i^\alpha, p)$ with $\alpha = 1$, and c_0 is a positive scaling constant. We use an intuitive data-driven approach to choose c_0. We first specify a set $\mathcal{C}$ of candidate choices for c_0 ranging from 0.001 to 1000. We choose the best c_0 that achieves the smallest objective function in (2) with $\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}}^{(1)}_{\text{SGD}}$ using data points from the first machine (see Algorithm 1), that is, $c_0 = \arg\min_{c\in\mathcal{C}} \frac{1}{n}\sum_{i=1}^n f(\widehat{\boldsymbol{\theta}}^{(1)}_{\text{SGD}}, \boldsymbol{\xi}_i^{(1)})$, where $\{\boldsymbol{\xi}_i^{(1)}, i = 1, 2, \ldots, n\}$ denotes the samples on the first machine. For Dis-FONE, the step-size is set to $\eta = c_0' m/n$, where c_0' is also selected from a set $\mathcal{C}$ of candidate constants. Similarly, we choose the best tuning constant that achieves the smallest objective in (2) with $\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}}_{\text{dis},1}$ and samples from the first machine. Here, $\widehat{\boldsymbol{\theta}}_{\text{dis},1}$ is the output of Dis-FONE after the first round of the algorithm. That is, $c_0' = \arg\min_{c\in\mathcal{C}} \frac{1}{n}\sum_{i=1}^n f(\widehat{\boldsymbol{\theta}}_1, \boldsymbol{\xi}_i^{(1)})$. More simulation results with different choices of the stepsizes are provided in Section G.3 of the supplementary materials.

Moreover, by Condition `(A1)`, we set the mini-batch size in DC-SGD (or the size of B_t in Dis-FONE, see Algorithm 3) as $m = \lfloor p\log n \rfloor$, where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x. For Dis-FONE, we first set the number of the iterations T in each round as $T = 20$ and the number of rounds $K = 20$ for logistic regression and $K = 80$ for quantile regression. Note that due to the nonsmoothness in the loss function of quantile regression, it requires more rounds of iterations K to ensure the convergence. In practice, we could also adopt a simple data-driven approach to determine the number of rounds K. In particular, we could stop the algorithm when the change (norm of the difference) between the estimators for two consecutive rounds is negligible. We carefully evaluate the effect of T and K (by considering different values of T and K) in Section 5.1.3. We also compare the methods with $\widehat{\boldsymbol{\theta}}_{\text{FONE}}$ (Algorithm 2), which corresponds to the Dis-FONE $\widehat{\boldsymbol{\theta}}_{\text{dis},K}$ with only $K = 1$ round. All results reported below are based on the average of 100 independent runs of simulations.

5.1.1. Effect of N and p

In Tables 1 and 2, we fix the number of machines $L = 20$ and vary the total sample size N from $\{10^5, 2\times10^5, 5\times10^5\}$ and dimension $p \in \{100, 200, 500\}$. Results for logistic regression are reported in Table 1 and results for quantile regression are in Table 2. In both tables, the left columns provide the L_2 estimation errors (with respect to the truth $\boldsymbol{\theta}^*$) of the DC-SGD estimator $\widehat{\boldsymbol{\theta}}_{\text{DC}}$, the SGD estimator $\widehat{\boldsymbol{\theta}}_{\text{SGD}}$, Dis-FONE $\widehat{\boldsymbol{\theta}}_{\text{dis},K}$, and the ERM $\widehat{\boldsymbol{\theta}}$. We note that both SGD and ERM are nondistributed

Table 1. Logistic regression: comparisons of L_2-errors when varying the total sample size N and dimension p changes.

	L_2-distance to the truth $\boldsymbol{\theta}^*$						L_2-distance to ERM $\widehat{\boldsymbol{\theta}}$		
p	$\widehat{\boldsymbol{\theta}}_0$	$\widehat{\boldsymbol{\theta}}_{\text{DC}}$	$\widehat{\boldsymbol{\theta}}_{\text{SGD}}$	$\widehat{\boldsymbol{\theta}}_{\text{FONE}}$	$\widehat{\boldsymbol{\theta}}_{\text{dis},K}$	$\widehat{\boldsymbol{\theta}}$	$\widehat{\boldsymbol{\theta}}_{\text{DC}}$	$\widehat{\boldsymbol{\theta}}_{\text{SGD}}$	$\widehat{\boldsymbol{\theta}}_{\text{dis},K}$
				$N = 10^5$					
100	1.251	0.447	0.148	0.724	0.103	0.093	0.445	0.116	0.038
200	1.899	1.096	0.523	1.046	0.168	0.153	1.091	0.494	0.049
500	4.509	3.853	3.111	2.154	0.338	0.301	3.748	3.021	0.085
				$N = 2 \times 10^5$					
100	1.303	0.390	0.100	0.594	0.072	0.067	0.386	0.074	0.025
200	2.094	1.248	0.315	0.821	0.115	0.109	1.235	0.286	0.034
500	4.717	3.920	2.189	1.725	0.222	0.211	3.891	2.133	0.045
				$N = 5 \times 10^5$					
100	1.342	0.313	0.081	0.347	0.046	0.042	0.304	0.069	0.018
200	1.833	0.874	0.169	0.749	0.073	0.068	0.868	0.152	0.023
500	4.835	3.885	1.006	1.413	0.141	0.130	3.859	0.989	0.036

NOTE: Here the number of machines $L = 20$. Denote by $\widehat{\boldsymbol{\theta}}_{\text{DC}}$ the DC-SGD estimator, $\widehat{\boldsymbol{\theta}}_{\text{SGD}}$ the SGD estimator on the entire dataset in a nondistributed setting, $\widehat{\boldsymbol{\theta}}_{\text{FONE}}$ the single-round FONE of Algorithm 2, and $\widehat{\boldsymbol{\theta}}_{\text{dis},K}$ the Dis-FONE with $K = 20$.

Table 2. Quantile regression: comparisons of L_2-errors when varying the total sample size N and dimension p.

	L_2-distance to the truth $\boldsymbol{\theta}^*$						L_2-distance to ERM $\widehat{\boldsymbol{\theta}}$		
p	$\widehat{\boldsymbol{\theta}}_0$	$\widehat{\boldsymbol{\theta}}_{\text{DC}}$	$\widehat{\boldsymbol{\theta}}_{\text{SGD}}$	$\widehat{\boldsymbol{\theta}}_{\text{FONE}}$	$\widehat{\boldsymbol{\theta}}_{\text{dis},K}$	$\widehat{\boldsymbol{\theta}}$	$\widehat{\boldsymbol{\theta}}_{\text{DC}}$	$\widehat{\boldsymbol{\theta}}_{\text{SGD}}$	$\widehat{\boldsymbol{\theta}}_{\text{dis},K}$
				$N = 10^5$					
100	0.450	0.079	0.063	0.191	0.047	0.043	0.073	0.050	0.020
200	0.715	0.114	0.109	0.342	0.082	0.071	0.106	0.097	0.035
500	1.278	0.198	0.176	0.554	0.144	0.126	0.176	0.142	0.062
				$N = 2 \times 10^5$					
100	0.450	0.070	0.037	0.130	0.035	0.030	0.067	0.021	0.015
200	0.726	0.101	0.067	0.246	0.059	0.054	0.098	0.037	0.027
500	1.287	0.176	0.118	0.379	0.098	0.076	0.157	0.065	0.046
				$N = 5 \times 10^5$					
100	0.451	0.043	0.030	0.114	0.029	0.025	0.037	0.017	0.014
200	0.719	0.067	0.047	0.157	0.041	0.037	0.064	0.026	0.020
500	1.294	0.105	0.076	0.264	0.074	0.057	0.091	0.041	0.035

NOTE: Here the number of machines $L = 20$. Denote by $\widehat{\boldsymbol{\theta}}_{\text{DC}}$ the DC-SGD estimator, $\widehat{\boldsymbol{\theta}}_{\text{SGD}}$ the SGD estimator on the entire dataset in a nondistributed setting, $\widehat{\boldsymbol{\theta}}_{\text{FONE}}$ the single-round FONE of Algorithm 2, and $\widehat{\boldsymbol{\theta}}_{\text{dis},K}$ the Dis-FONE with $K = 80$.

algorithms for pooled data. We will show that in many cases our distributed Dis-FONE estimator even outperforms the nondistributed SGD. For reference, we also report L_2-errors of the initial estimator $\widehat{\boldsymbol{\theta}}_0$. The right columns report the L_2-distances to the ERM $\widehat{\boldsymbol{\theta}}$.

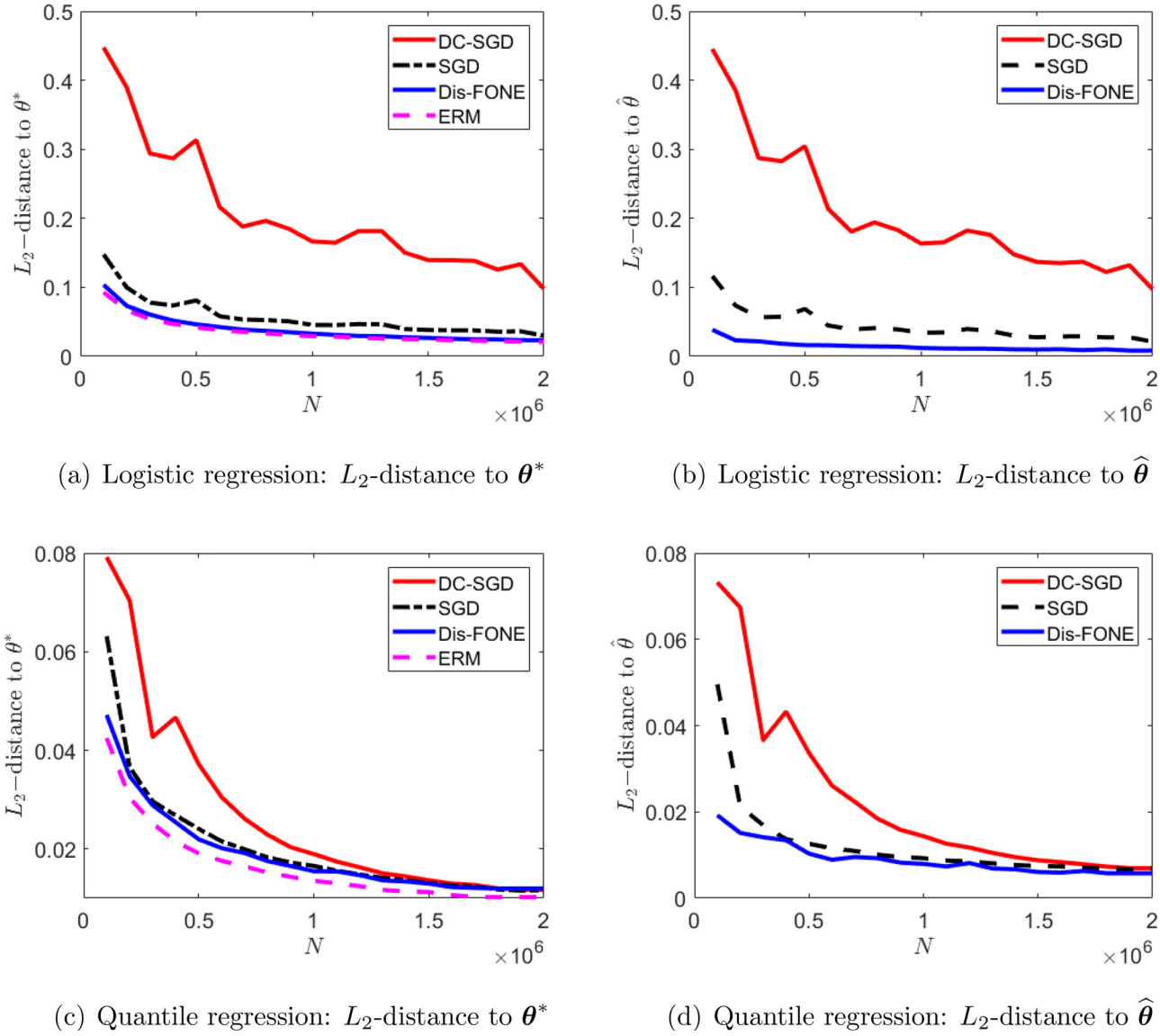
From Tables 1 and 2, we can see that the proposed Dis-FONE $\widehat{\boldsymbol{\theta}}_{\text{dis},K}$ achieves similar errors as the ERM $\widehat{\boldsymbol{\theta}}$ in all cases, and outperforms DC-SGD and SGD especially when p is large. We also provide Figure 1 that captures the performance of the estimators in terms of their L_2-errors when the total sample size N increases. From Figure 1, we can see that the estimation error for each method decreases as N increases. Moreover, the L_2-error of Dis-FONE is very close to the ERM as N increases, while there is a significant gap between DC-SGD and the ERM.

5.1.2. Effect on the Number of Machines L

For the effect on the number of machines L, we fix the total sample size $N = 10^5$ and the dimension $p = 100$ and vary the number of machines L from 1 to 200, and plot the L_2-errors in Figure 2. From Figure 2, the L_2-error of DC-SGD increases as L increases (i.e., each machine has fewer samples). In contrast, the L_2-error of Dis-FONE versus L is almost flat, and is quite close to ERM even when L is large. This is consistent with our theoretical result that DC-SGD will fail when L is large. The SGD estimator, which is the $L = 1$ case of DC-SGD (and thus its error is irrelevant of L and is presented by a horizontal line), provides moderate accuracy. Further increasing L would lead to an excessively small local sample size (e.g., when $N = 10^5$ and $L > 200$, we will have the local sample size $n < 500$, and thus is unrealistic in practice, considering that the dimensionality $p = 100$.

5.1.3. Effect of K and T in Dis-FONE

For Dis-FONE, we provide the comparison of the estimator errors with different numbers of rounds K and numbers of inner iterations T. In Figure 3, we fix the total sample size $N = 10^5$, the dimension $p = 100$, the number of machines $L = 20$ and vary T from $\{5, 20, 100\}$. The x-axis in Figure 3 is the number of rounds K. For all three cases of T, the performance of Dis-FONE is quite desirable and reaches the accuracy of the ERM when K becomes



(a) Logistic regression: L_2-distance to $\boldsymbol{\theta}^*$

(b) Logistic regression: L_2-distance to $\widehat{\boldsymbol{\theta}}$

(c) Quantile regression: L_2-distance to $\boldsymbol{\theta}^*$

(d) Quantile regression: L_2-distance to $\widehat{\boldsymbol{\theta}}$

Figure 1. Comparison of L_2-errors when N increases. The left column reports the L_2-errors with respect to the truth $\boldsymbol{\theta}^*$ and the right column reports the L_2-errors with respect to the ERM $\widehat{\boldsymbol{\theta}}$. Here the dimension $p = 100$ and the number of machines $L = 20$. In Dis-FONE, we set $K = 20$ in the logistic regression case and $K = 80$ in the quantile regression case.

larger. When T is smaller, it requires a larger K for Dis-FONE to converge. In other words, we need to perform more rounds of Dis-FONE to achieve the same accuracy.

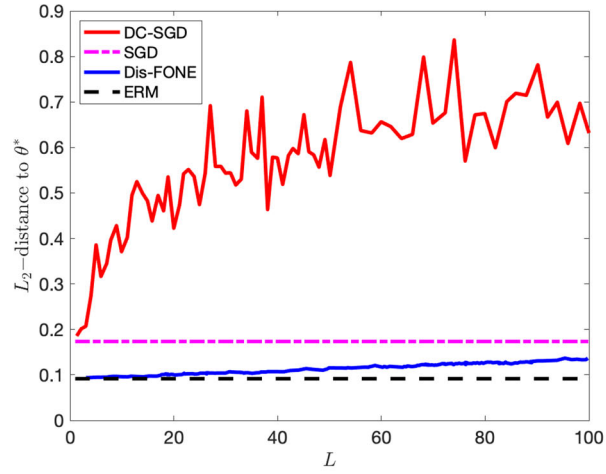
5.1.4. Effect on the Unbalanced Data Partition

In previous simulation studies, the entire dataset is evenly separated on different machines. As one can see from Algorithm 3 and Theorem 3.3, the subsample size on the first machine n_1 plays a different role in Dis-FONE than those on the other machines $n_2, n_3, \ldots, n_L$. In Figure 4, we investigate the effect of n_1 by varying n_1 from N/L (the case of evenly distributed) to $10 \times N/L$. Let the remaining data points be evenly distributed on the other machines, that is, $n_2 = n_3 = \cdots = n_L = (N-n_1)/(L-1)$. We set $N = 10^5$ and $L = 20$. From Figure 4, the L_2-error of Dis-FONE gets much closer to ERM $\widehat{\boldsymbol{\theta}}$ in (2) when the largest subsample size n_1 increases, which is consistent with our theoretical results.

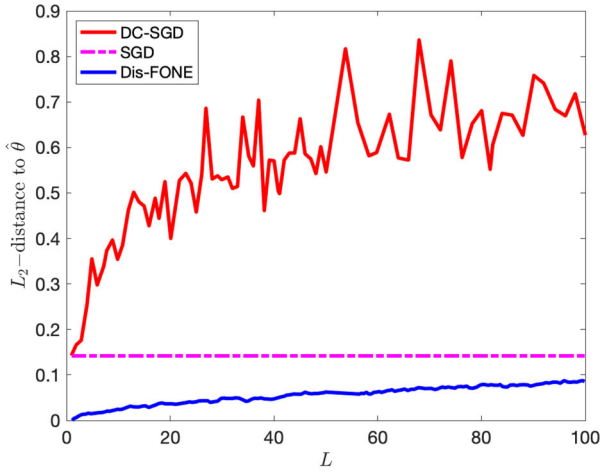
In the supplementary materials, we further investigate the cases of correlated design, the effect of the quality of the initial estimator, and different choices of the stepsizes (see Section G for more details). We also conduct simulations to compare our proposed methods (DC-SGD and Dis-FONE) to the existing methods in Section G.4 in terms of statistical accuracy and computation time. For logistic regression, we compare our algorithm with CSL (Jordan, Lee, and Yang 2019) for logistic regression. Both Dis-FONE and CSL methods achieve nearly optimal performance as compared to the ERM, while Dis-FONE accelerates CSL. For quantile regression, we compare our methods with DC-QR (Volgushev, Chao, and Cheng 2019). Dis-FONE outperforms DC-QR in terms of both computation time and statistical accuracy since DC-QR suffers from the restriction on the subsample size analogous to the case of DC-SGD.

5.1.5. Experiments on Statistical Inference

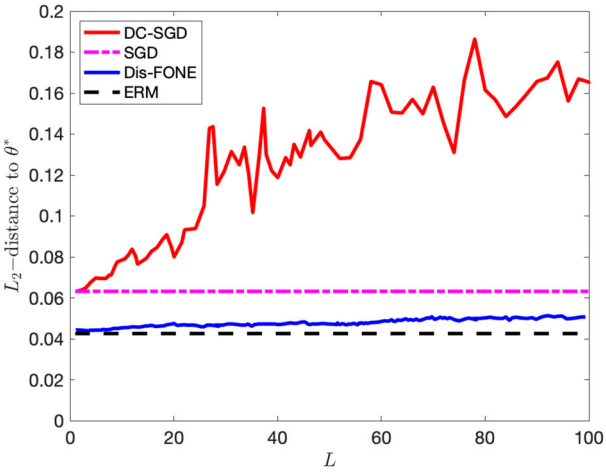
In this section, we provide simulation studies for estimating $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$, where $\boldsymbol{\Sigma}$ is the population Hessian matrix of the



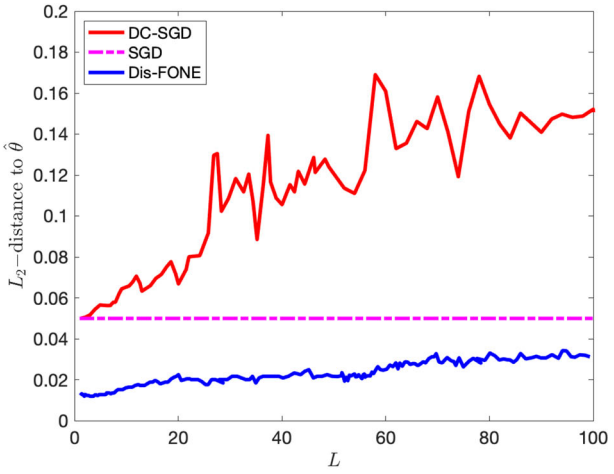
(a) Logistic regression: L_2-distance to $\boldsymbol{\theta}^*$



(b) Logistic regression: L_2-distance to $\widehat{\boldsymbol{\theta}}$



(c) Quantile regression: L_2-distance to $\boldsymbol{\theta}^*$



(d) Quantile regression: L_2-distance to $\widehat{\boldsymbol{\theta}}$

Figure 2. Comparison of L_2-errors when the number of machines L increases. Here the total sample size $N = 10^5$ and the dimension $p = 100$. Denote by $\widehat{\boldsymbol{\theta}}_{\text{dis},K}$ the Dis-FONE with $K = 20$ in the logistic regression case and $K = 80$ in the quantile regression case.

underlying regression model and $\|\boldsymbol{w}\|_2 = 1$. As we illustrate in Section 4, this estimator plays an important role in estimating the limiting variance of the ERM.

In this experiment, we specify $\boldsymbol{w} = \mathbf{1}_p/\sqrt{p}$, the sample size $n \in \{10^5, 2 \times 10^5, 5 \times 10^5\}$, the dimension $p \in \{100, 200, 500\}$. According to Theorems 4.1 and 4.2, we set the multiplier $\tau_n = ((p \log n)/n)^{1/2}$, the step-size $\eta_n = (p \log n)/n$ for logistic regression, and $\tau_n = ((p \log n)/n)^{1/3}$, $\eta_n = ((p \log n)/n)^{2/3}$ for quantile regression, respectively.

Given $\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}}$, we are able to compute the estimator of limiting variance $\boldsymbol{w}'\boldsymbol{\Sigma}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ using (25). Based on that and (3), we construct the 95% confidence interval for $\boldsymbol{w}'\boldsymbol{\theta}^*$ as follows,

$$\boldsymbol{w}'\widehat{\boldsymbol{\theta}} \pm \Phi^{-1}(0.975)\sqrt{\boldsymbol{w}'\boldsymbol{\Sigma}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}^{-1}\boldsymbol{w}/n}, \tag{28}$$

where $\Phi(\cdot)$ is the CDF of the standard normal random variable.

The left columns in Table 3 present the average coverage rates of the confidence intervals of $\boldsymbol{w}'\boldsymbol{\theta}^*$ constructed by (28) and their average interval lengths. In the right columns of Table 3, we report the square root of the ratio between the estimated variance and the true limiting variance, that is,

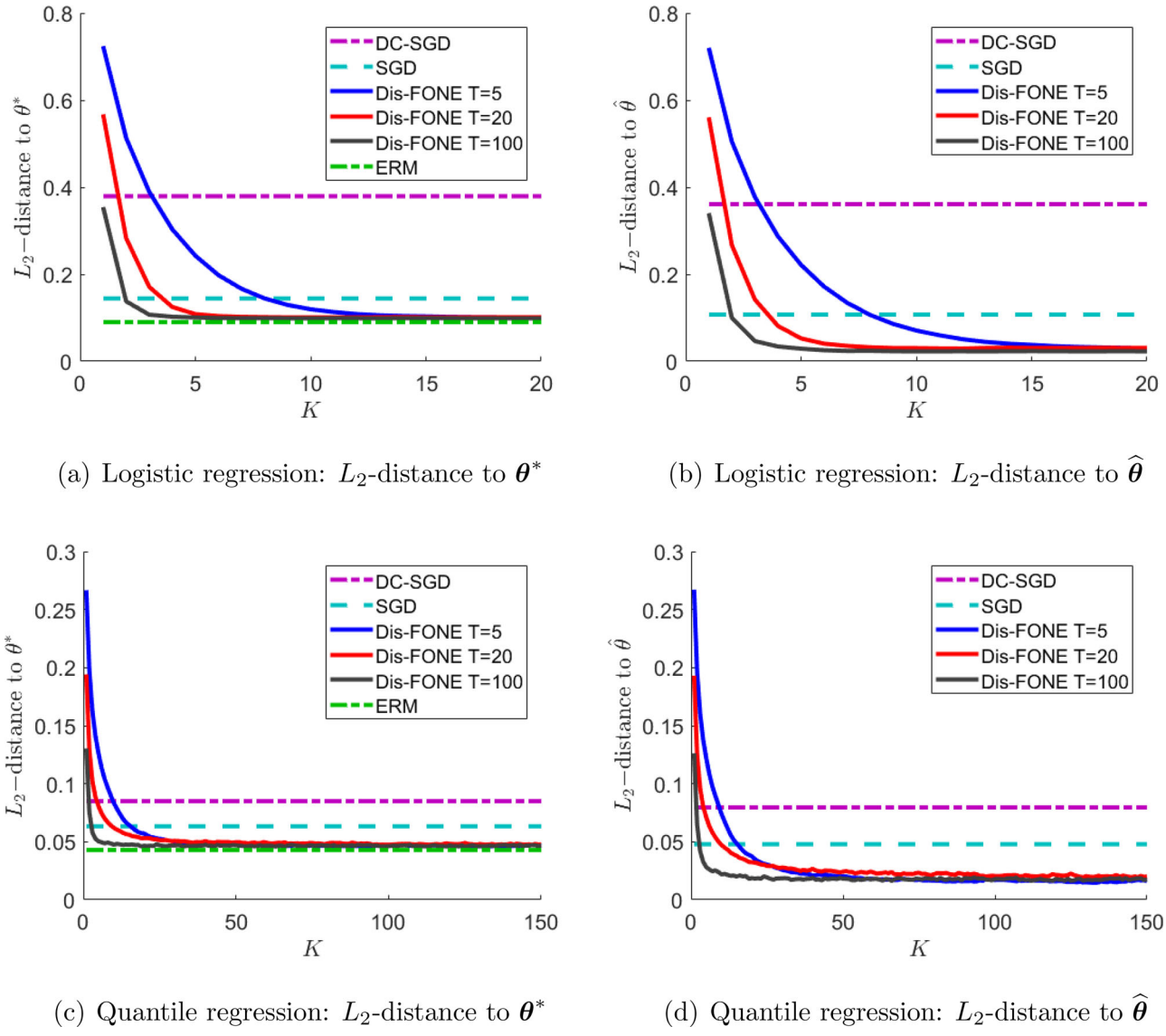
$$\sqrt{(\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}})'\widehat{\boldsymbol{A}}(\widehat{\boldsymbol{\Sigma}^{-1}\boldsymbol{w}})\Big/\boldsymbol{w}'\boldsymbol{\Sigma}^{-1}\boldsymbol{A}\boldsymbol{\Sigma}^{-1}\boldsymbol{w}}.$$

From Table 3, our estimator achieves good performance for both logistic and quantile regression models. As the sample size n increases, the coverage rates become closer to the nominal level and the ratios get closer to 1.

5.2. Real Data Analysis—Census 2000 Data

In this section, we provide real data analysis of our proposed methods. We consider the sampled U.S. 2000 Census dataset,[3] consisting of annual salary and related features on employed

[3] U.S. Census, *http://www.census.gov/census2000/PUMS5.html*



(a) Logistic regression: L_2-distance to $\boldsymbol{\theta}^*$

(b) Logistic regression: L_2-distance to $\widehat{\boldsymbol{\theta}}$

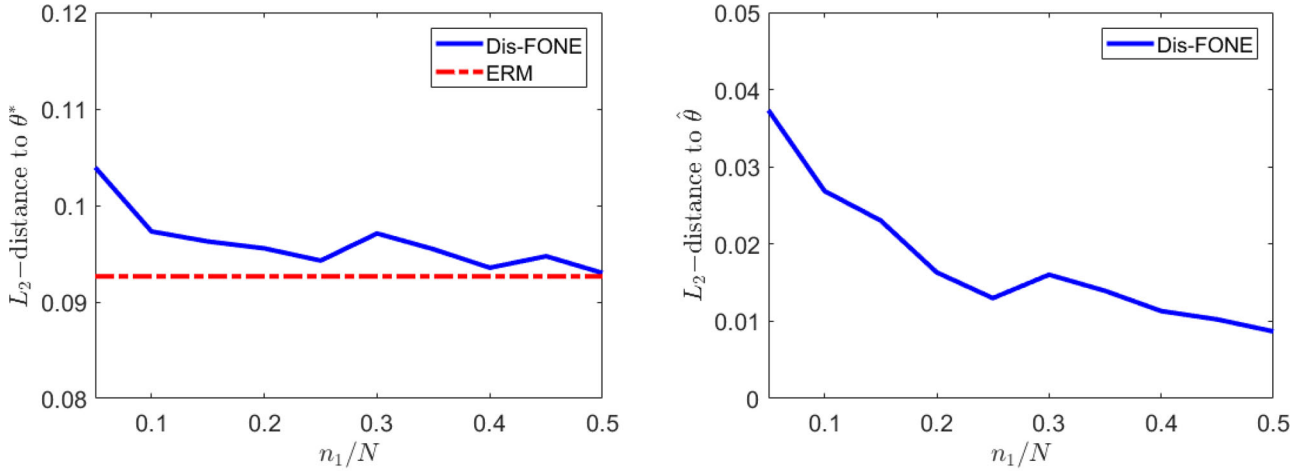
(c) Quantile regression: L_2-distance to $\boldsymbol{\theta}^*$

(d) Quantile regression: L_2-distance to $\widehat{\boldsymbol{\theta}}$

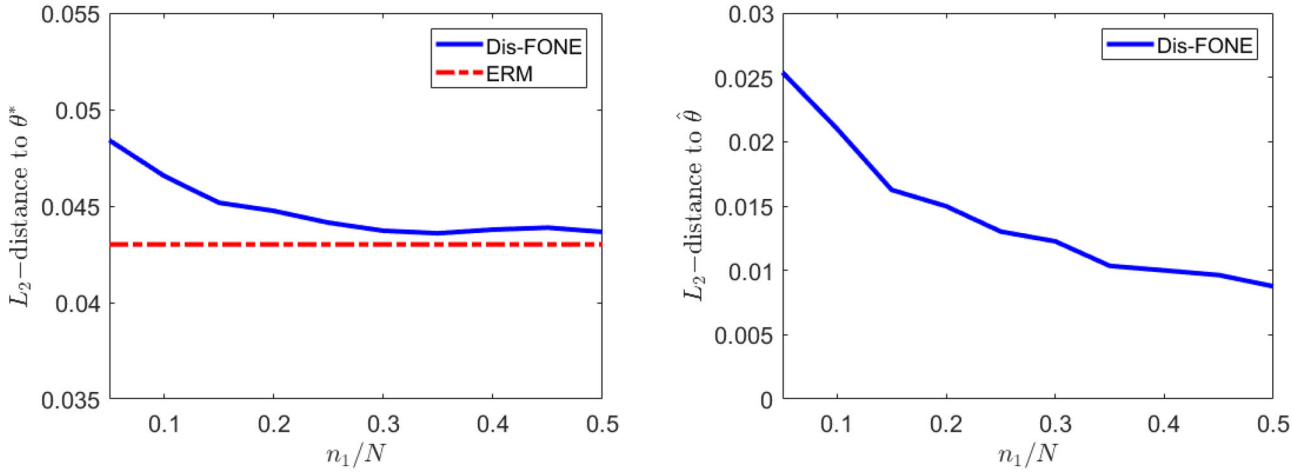
Figure 3. Comparison of L_2-errors when the number of rounds K in Dis-FONE increases. The x-axis is the number of rounds K in Dis-FONE. Here the total sample size $N = 10^5$, the dimension $p = 100$, and the number of machines $L = 20$. The errors of DC-SGD, SGD, and ERM are presented by the horizontal lines since their performance is irrelevant of K.

full-time wage and salary workers who reported that they worked 40 weeks or more and worked 35 hr or more per week. The U.S. 2000 Census dataset is a widely used dataset in quantile regression literature (see, e.g., Angrist, Chernozhukov, and Fernández-Val 2006; Yang, Meng, and Mahoney 2013). The entire sample size is 5×10^6 and the dimension $p = 11$. We perform a quantile regression on the dataset, which treats the annual salary as the response variable, with two different quantile levels $\tau = 0.25$ and $\tau = 0.5$ (i.e., the least absolute deviations regression). We use $L = 100$ machines/nodes and vary the total sample size N from 5×10^5 to 5×10^6 to compare our Dis-FONE with DC-SGD in a distributed environment. In DC-SGD, we set $\alpha = 1$ in the step-size $r_i = c_0/\max(i^\alpha, p)$. In Dis-FONE, we set $T = 100$, $K = 20$, and step-size $\eta_n = c_0' m/n$. The constants c_0 and c_0' is chosen in the say way as in the simulation study. More particularly, we choose the best c_0 and c_0' that achieves the smallest objective function in (2) with $\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}}^{(1)}_{\text{SGD}}$ and $\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}}_{\text{dis},1}$ using data points from the first machine. The ERM estimator is computed by solving the quantile regression using the interior-point method with pooled data on a single powerful machine.

From Figure 5, our proposed Dis-FONE $\widehat{\boldsymbol{\theta}}_{\text{dis},K}$ is very close to the ERM estimator and outperforms both DC-SGD and SGD estimators. As N increases, both DC-SGD and SGD estimators are closer to the ERM estimator. In addition to estimation accuracy, we further investigate the performance on the testing set. For a given N, we split the data into the training set (with $0.8N$ samples) and testing set (with $0.2N$). We estimate $\widehat{\boldsymbol{\theta}}_{\text{method}}$ using an estimation method, such as, DC-SGD, SGD, Dis-FONE, on the training set and evaluate the quantile objective value on the testing data. Denote the obtained quantile objective value on the testing set using a given method and the ERM by $\widehat{f}_{\text{method}}$ and $\widehat{f}_{\text{ERM}}$, respectively. We report the relative errors of objective values $|\widehat{f}_{\text{method}} - \widehat{f}_{\text{ERM}}|/|\widehat{f}_{\text{ERM}}|$ and from Figure 6, the relative errors of Dis-FONE are very close to zero.



(a) Logistic regression: L_2-distance to $\boldsymbol{\theta}^*$ (b) Logistic regression: L_2-distance to $\widehat{\boldsymbol{\theta}}$



(c) Quantile regression: L_2-distance to $\boldsymbol{\theta}^*$ (d) Quantile regression: L_2-distance to $\widehat{\boldsymbol{\theta}}$

Figure 4. Comparison of L_2-errors when the subsample size of the first machine n_1 in Dis-FONE increases. The x-axis is the ratio of n_1 to the total sample size N. Here the total sample size $N = 10^5$, the dimension $p = 100$, and the number of machines $L = 20$.

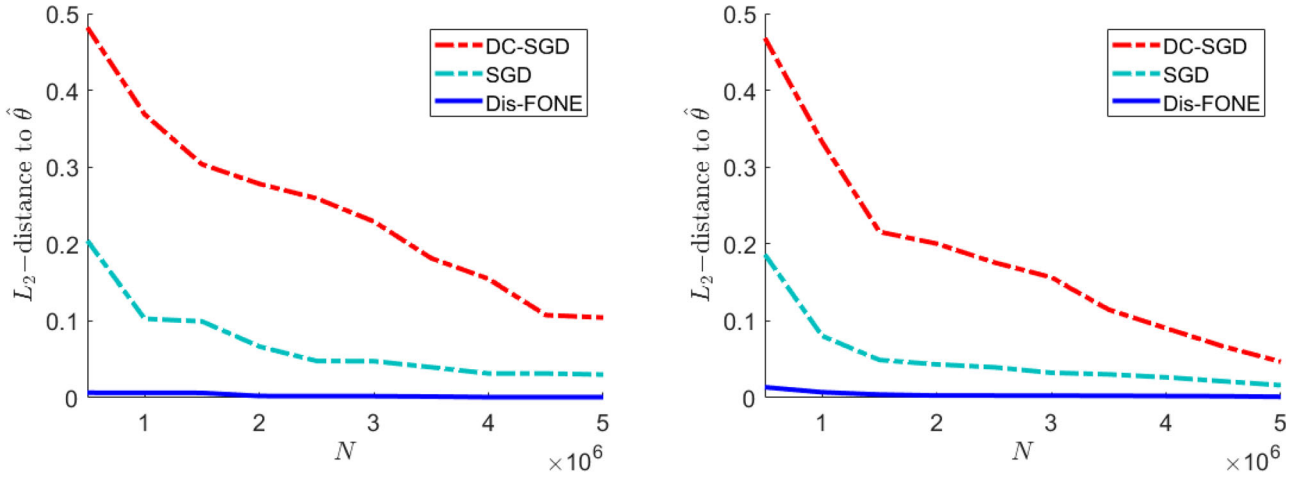
Table 3. Left columns: Coverage rates and average confidence interval length in the brackets; right columns: square roots of the ratios of the estimated variance to the true limiting variance of ERM $\widehat{\boldsymbol{\theta}}$.

		Coverage rates (Avg length)			Square root ratio		
Model	n	$p = 100$	$p = 200$	$p = 500$	$p = 100$	$p = 200$	$p = 500$
Logistic	$n = 10^5$	94.37 (0.127)	93.76 (0.174)	91.67 (0.271)	1.043	1.033	1.027
	$n = 2 \times 10^5$	94.49 (0.131)	94.10 (0.168)	92.14 (0.265)	1.041	1.027	1.019
	$n = 5 \times 10^5$	94.71 (0.128)	94.23 (0.167)	92.31 (0.254)	1.017	1.017	1.014
Quantile	$n = 10^5$	94.57 (0.160)	94.11 (0.221)	92.96 (0.358)	1.042	1.023	1.027
	$n = 2 \times 10^5$	94.64 (0.159)	94.24 (0.204)	93.04 (0.349)	1.007	1.004	1.004
	$n = 5 \times 10^5$	94.67 (0.154)	94.59 (0.191)	93.47 (0.344)	1.005	1.002	1.003

NOTE: The sample size $n \in \{10^5, 2 \times 10^5, 5 \times 10^5\}$ and dimension $p \in \{100, 200, 500\}$. The multiplier $\tau_n = ((p \log n)/n)^{1/2}$, the step-size $\eta_n = (p \log n)/n$ for logistic regression, and $\tau_n = ((p \log n)/n)^{1/3}$, $\eta_n = ((p \log n)/n)^{2/3}$ for quantile regression, respectively.

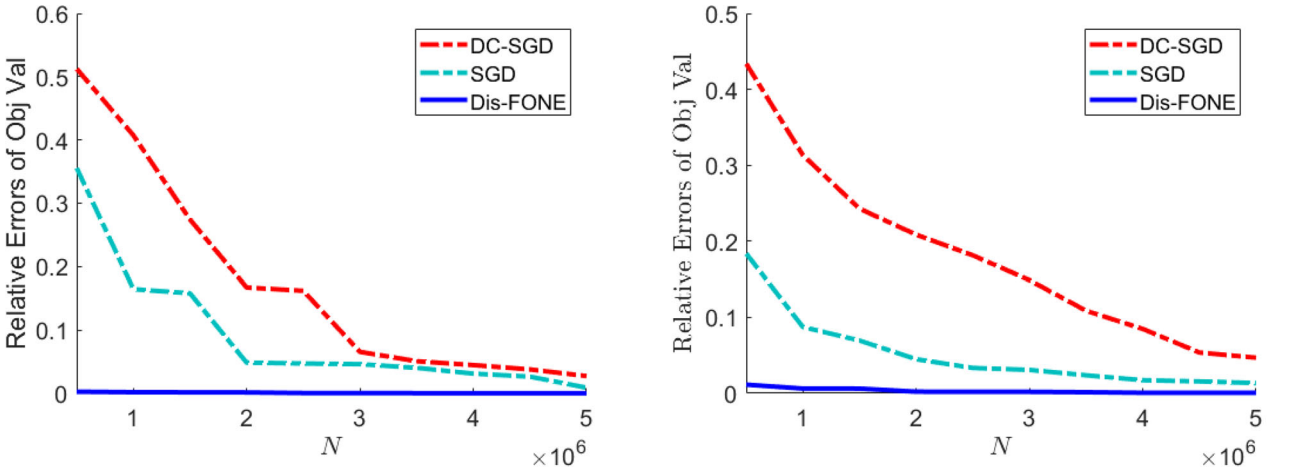
6. Conclusions

This article studies general distributed estimation and inference problems based on stochastic subgradient descent. We propose an efficient FONE for estimating $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ and its distributed version. The key idea behind our method is to use stochastic gradient information to approximate the Newton step. We further characterize the theoretical properties when using FONE for distributed estimation and inference with both smooth and nonsmooth loss functions. We also conduct numerical studies to demonstrate the performance of the proposed distributed FONE. The proposed FONE of $\boldsymbol{\Sigma}^{-1}\boldsymbol{w}$ is a general estimator,



(a) $\tau = 0.25$: L_2-distance to $\widehat{\boldsymbol{\theta}}$ (b) $\tau = 0.5$: L_2-distance to $\widehat{\boldsymbol{\theta}}$

Figure 5. Real data analysis: comparison of L_2-errors when the sample size N increases.



(a) $\tau = 0.25$: relative errors of objective values (b) $\tau = 0.5$: relative errors of objective values

Figure 6. Real data analysis: comparison of the relative errors of objective values $|\widehat{f}_{\text{method}} - \widehat{f}_{\text{ERM}}|/|\widehat{f}_{\text{ERM}}|$ on the testing data when the sample size N increases.

which could find applications to other statistical estimation problems. While this article focuses on convex loss functions, the proposed methods can be directly applied to nonconvex objectives. It would be an interesting future direction to derive the convergence rates for nonconvex settings.

Supplementary Materials

The supplementary material provides the verification of conditions, the theory of mini-batch SGD with diverging dimension, the proofs of all technical results in the main paper, and additional numerical experiments.

Acknowledgments

The authors are very grateful to anonymous referees and the associate editor for their detailed and constructive comments that considerably improved the quality of this article.

Funding

Weidong Liu is supported by National Program on Key Basic Research Project (973 Program, 2018AAA0100704), NSFC grant nos. 11825104 and 11690013, Youth Talent Support Program, and a grant from Australian Research Council.

References

Angrist, J., Chernozhukov, V., and Fernández-Val, I. (2006), "Quantile Regression Under Misspecification, With an Application to the US Wage Structure," *Econometrica*, 74, 539–563. [14]

Banerjee, M., Durot, C., and Sen, B. (2019), "Divide and Conquer in Non-Standard Problems and the Super-Efficiency Phenomenon," *The Annals of Statistics*, 47, 720–757. [2]

Battey, H., Fan, J., Liu, H., Lu, J., and Zhu, Z. (2018), "Distributed Estimation and Inference With Statistical Guarantees," *The Annals of Statistics*, 46, 1352–1382. [2,6]

Chen, X., Lee, J. D., Li, H., and Yang, Y. (2021), "Distributed Estimation for Principal Component Analysis: A Gap-Free Approach," *Journal of the American Statistical Association* (to appear). [2]

Chen, X., Lee, J. D., Tong, X. T., and Zhang, Y. (2020), "Statistical Inference for Model Parameters in Stochastic Gradient Descent," *The Annals of Statistics*, 48, 251–273. [3]

Chen, X., Liu, W., Mao, X., and Yang, Z. (2020), "Distributed High-Dimensional Regression Under a Quantile Loss Function," *Journal of Machine Learning Research*, 21, 1–43. [2]

Chen, X., Liu, W., and Zhang, Y. (2019), "Quantile Regression Under Memory Constraint," *The Annals of Statistics*, 47, 3244–3273. [2]

Chen, X., and Xie, M. (2014), "A Split-and-Conquer Approach for Analysis of Extraordinarily Large Data," *Statistica Sinica*, 24, 1655–1684. [2,6]

Fan, J., Wang, D., Wang, K., and Zhu, Z. (2019), "Distributed Estimation of Principal Eigenspaces," *The Annals of Statistics*, 47, 3009–3031. [2]

He, X., and Shao, Q.-M. (2000), "On Parameters of Increasing Dimensions," *Journal of Multivariate Analysis*, 73, 120–135. [8,9]

Huang, C., and Huo, X. (2019), "A Distributed One-Step Estimator," *Mathematical Programming*, 174, 41–76. [2,6]

Johnson, R., and Zhang, T. (2013), "Accelerating Stochastic Gradient Descent Using Predictive Variance Reduction," in *Advances in Neural Information Processing Systems*. [2]

Jordan, M. I., Lee, J. D., and Yang, Y. (2019), "Communication-Efficient Distributed Statistical Inference," *Journal of the American Statistical Association*, 114, 668–681. [2,12]

Lai, T. L. (2003), "Stochastic Approximation," *The Annals of Statistics*, 31, 391–406. [5]

Lee, J. D., Lin, Q., Ma, T., and Yang, T. (2017), "Distributed Stochastic Variance Reduced Gradient Methods by Sampling Extra Data With Replacement," *Journal of Machine Learning Research*, 18, 4404–4446. [2]

Lee, J. D., Liu, Q., Sun, Y., and Taylor, J. E. (2017), "Communication-Efficient Sparse Regression," *Journal of Machine Learning Research*, 18, 1–30. [6]

Li, R., Lin, D. K., and Li, B. (2013), "Statistical Inference in Massive Data Sets," *Applied Stochastic Models in Business and Industry*, 29, 399–409. [2,6]

Li, T., Kyrillidis, A., Liu, L., and Caramanis, C. (2018), "Approximate Newton-Based Statistical Inference Using Only Stochastic Gradients," arXiv no. 1805.08920. [2]

Pang, L., Lu, W., and Wang, H. J. (2012), "Variance Estimation in Censored Quantile Regression via Induced Smoothing," *Computational Statistics & Data Analysis*, 56, 785–796. [10]

Polyak, B. T., and Juditsky, A. B. (1992), "Acceleration of Stochastic Approximation by Averaging," *SIAM Journal on Control and Optimization*, 30, 838–855. [3,7]

Shi, C., Lu, W., and Song, R. (2018), "A Massive Data Framework for *M*-Estimators With Cubic-Rate," *Journal of the American Statistical Association*, 113, 1698–1709. [2]

Volgushev, S., Chao, S.-K., and Cheng, G. (2019), "Distributed Inference for Quantile Regression Processes," *The Annals of Statistics*, 47, 1634–1662. [2,6,12]

Wang, J., and Zhang, T. (2017), "Improved Optimization of Finite Sums With Minibatch Stochastic Variance Reduced Proximal Iterations," arXiv no. 1706.07001. [2]

Wang, X., Yang, Z., Chen, X., and Liu, W. (2019), "Distributed Inference for Linear Support Vector Machine," *Journal of Machine Learning Research*, 20, 1–41. [2]

Yang, J., Meng, X., and Mahoney, M. (2013), "Quantile Regression for Large-Scale Applications," in *International Conference on Machine Learning*. [14]

Zhang, Y., Duchi, J., and Wainwright, M. (2015), "Divide and Conquer Kernel Ridge Regression: A Distributed Algorithm With Minimax Optimal Rates," *Journal of Machine Learning Research*, 16, 3299–3340. [2,6]

Zhao, T., Cheng, G., and Liu, H. (2016), "A Partially Linear Framework for Massive Heterogeneous Data," *The Annals of Statistics*, 44, 1400–1437. [2,6]