JUSTC

Article

http://justc.ustc.edu.cn

Received: December 05, 2022; Accepted: February 27, 2023

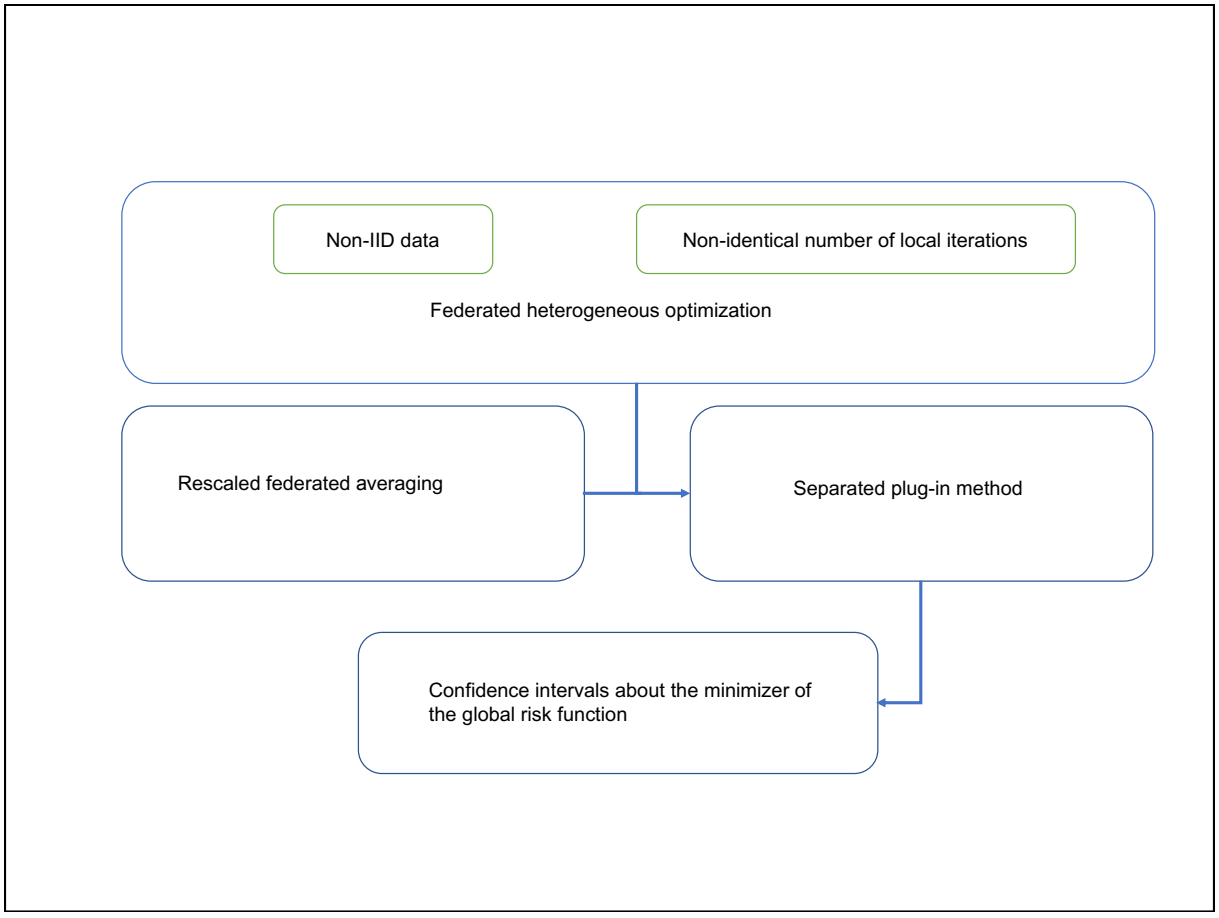
Online confidence interval estimation for federated heterogeneous optimization

Yu Wang, Wenquan Cui ✉, and Jianjun Xu ✉

International Institute of Finance, School of Management, University of Science and Technology of China, Hefei 230026, China

✉Correspondence: Wenquan Cui, E-mail: wqcui@ustc.edu.cn; Jianjun Xu, E-mail: xjj1994@mail.ustc.edu.cn

Graphical abstract



An online confidence interval estimation method called separated plug-in via rescaled federated averaging.

Public summary

■ Develop online statistical inference in federated learning for heterogeneous optimization.

■ Propose an online confidence interval estimation method being separated plug-in by rescaled federated averaging.

■ Establish the asymptotic normality and show the asymptotic covariance being inversely proportional to the client participation rate.

Citation: Wang Y, Cui W Q, Xu J J. Online confidence interval estimation for federated heterogeneous optimization. *JUSTC*, **2023**, 53(11): 1103. DOI: 10.52396/JUSTC-2022-0179

JUSTC

Article

http://justc.ustc.edu.cn

Received: December 05, 2022; Accepted: February 27, 2023

Online confidence interval estimation for federated heterogeneous optimization

Yu Wang, Wenquan Cui ✉, and Jianjun Xu ✉

International Institute of Finance, School of Management, University of Science and Technology of China, Hefei 230026, China

✉Correspondence: Wenquan Cui, E-mail: wqcui@ustc.edu.cn; Jianjun Xu, E-mail: xjj1994@mail.ustc.edu.cn

Cite This: *JUSTC*, **2023**, 53(11): 1103 (10pp) Read Online SI Supporting Information

Abstract: From a statistical viewpoint, it is essential to perform statistical inference in federated learning to understand the underlying data distribution. Due to the heterogeneity in the number of local iterations and in local datasets, traditional statistical inference methods are not competent in federated learning. This paper studies how to construct confidence intervals for federated heterogeneous optimization problems. We introduce the rescaled federated averaging estimate and prove the consistency of the estimate. Focusing on confidence interval estimation, we establish the asymptotic normality of the parameter estimate produced by our algorithm and show that the asymptotic covariance is inversely proportional to the client participation rate. We propose an online confidence interval estimation method called separated plug-in via rescaled federated averaging. This method can construct valid confidence intervals online when the number of local iterations is different across clients. Since there are variations in clients and local datasets, the heterogeneity in the number of local iterations is common. Consequently, confidence interval estimation for federated heterogeneous optimization problems is of great significance.

Keywords: federated learning; confidence interval; number of local iterations; online learning

CLC number: O212; TP181 **Document code:** A

2020 Mathematics Subject Classification: 68T05; 62F30

1 Introduction

Federated learning is a privacy-preserving machine learning framework that allows multiple clients to collaboratively train a global model without transferring local data. For estimation and prediction problems, statistical inference is an indispensable method for measuring uncertainties[1–3]. However, computation constraints, memory restrictions, and communication budgets make traditional statistical estimation and inference methods incompetent under federated settings[4]. In addition, variations in the local datasets and heterogeneity in the number of local iterations make it difficult to perform confidence interval estimation under federated settings.

Suppose that there are N clients and a central server. Each client labels with an unique number in $[N] \triangleq \{1,2,\cdots,N\}$. The feature space $\mathcal{X}$ is a finite-dimensional Euclidean space $\mathbf{R}^d$ for some positive integer d. In regression, the response space is $\mathcal{Y}=\mathbf{R}$. For the classification problem, the label space is $\mathcal{Y}=\{1,2,\cdots,J\}$ for some positive integer J. The ith client has a local dataset consisted of identically and independently distributed (IID) samples from some unknown data distribution $\rho_i(z^i)$ over $\mathcal{Z}=\mathcal{X}\times\mathcal{Y}$, where $z^i=(x^i,y^i)$, $x^i\in\mathcal{X}$, and $y^i\in\mathcal{Y}$. The federated learning system intends to optimize a sum of risk functions, with only access to local stochastic gradient updates.

The federated optimization problem is to minimize the global risk function

$$R(\theta)=\sum_{i=1}^{N} p_i R_i(\theta), \tag{1}$$

where $R_i(\theta)=\mathbb{E}_{z^i\sim\rho_i} l_i(\theta;z^i)$ is the local risk function at the ith client. Here, $l_i(\theta;z^i)$ is a client-specified loss function, where $z^i\in\mathcal{Z}$ represents an independent realization from local data distribution ρ_i. The weight of the ith client is p_i, which satisfies $p_i>0$ and $\sum_{i=1}^{N} p_i=1$. In this research, θ is the model parameter and we assume that the parameter space is $\Theta\subset\mathbf{R}^p$. The minimizer of $R(\theta)$ is denoted by θ^* and the minimizer of $R_i(\theta)$ is denoted by θ_i^*. In the non-IID problem, the data distributions at each client $\{\rho_i\}_1^N$ may vary. Therefore, θ_i^* typically does not coincide with θ^* for $i\in[N]$.

A typical method to solve Eq. (1) is federated averaging (FedAvg)[5]. To reduce communication budges, only a subset of clients are selected in each communication round in FedAvg. The selected clients perform multiple local updates before these local models are aggregated to collaboratively train a global model. It is widely applied in many federated learning applications[6–11]. Federated learning has been widely studied on non-IID data. Zhao et al.[12] has shown that the accuracy of federated learning decreases significantly, by up to 55% for neural networks trained for highly skewed non-IID data, where each client trains only on a single class of data. Several approaches, such as CSFedAvg[13] and FL+HC[14], can

DOI: 10.52396/JUSTC-2022-0179

JUSTC, **2023**, 53(11): 1103

promote the accuracy of FedAvg on non-IID data. Some works are interested in the convergence of FedAvg on non-IID data. For instance, Li et al.[15] has developed convergence guarantees for FedAvg estimates on non-IID data under certain regularity conditions.

A batch of recent works[16–19] analyzing the convergence of federated optimization algorithms assume that E_i (number of local stochastic gradient descent (SGD) iterations at client i) is identical, i.e., $E_i = E_j$ for any $i, j \in [N]$. However, this assumption is unrealistic for real word datasets. The clients are usually very different. The size of datasets and the computation speeds of these clients typically vary. To make full use of the the local data, we perform SGD at each client with data arriving in a given time interval. Due to the streaming data, the number of SGD iterations is proportional to the number of arrived data. Since the size of their local datasets as well as their computation speeds is different, the number of local iterations is typically different. When the batch size is the same across clients, E_i is certainly proportional to n_i. In the seminal paper[5], McMahan et al. proposed federated averaging in which each client performs E epochs of local updates. Then, the number of local iterations is $E_i = \lfloor n_i E/B \rfloor$ at client i, where B is the mini-batch size, and n_i is the number of samples at the ith client. In this case, the number of local iterations can vary widely across clients. In short, clients update local parameters in a given time interval using datasets arrived in this period of time, which makes E_i proportional to n_i. Wang et al.[20] first analyzed FedAvg on non-IID data when the number of local SGD iterations is nonidentical across clients. They have shown that the heterogeneity in the number of local iterations will result in objective inconsistency. That is, the FedAvg estimate does not converge to θ^*. Some methods that are proposed to solve non-IID data, such as FedProx[21], SCAFFOLD[22], and VRLSGD[23], can be used to reduce the inconsistency to some extent. However, these methods require additional memory and slow down the convergence. They proposed the federated normalized averaging (FedNova) algorithm which eliminates the inconsistency and preserves fast convergence.

In the big data era, many classical optimization methods for statistical problems, such as gradient descent, need great memory storage and computation power. Hence, online optimization methods, such as SGD, for statistical problems are of interest. From a statistical viewpoint, it is essential to perform confidence interval estimation in federated learning and online learning. However, only a few papers have considered confidence interval estimation problems in online learning, and statistical estimation problems have been even rarely studied in federated learning. In the online fashion, Ruppert[24] and Polyak et al.[25] have proven that the averaged SGD path is asymptotically normal with unknown asymptotic covariance. To conduct online confidence interval estimation, there are many studies trying to estimate the unknown asymptotic covariance. Zhu et al.[26] introduced a fully online overlapping estimate of the asymptotic covariance using only the iterates from SGD and its nonoverlapping variant. Fang et al.[27] introduced an online bootstrap procedure for estimating confidence intervals. In federated learning, Li et al.[28] showed how to perform statistical inference via local SGD[16]. They proposed two online confidence interval estimation methods under federated settings. However, they assumed that the number of local iterations is identical across clients. The identical number of local iterations violates the imbalanced nature of federated learning. More importantly, local SGD requires full client participation. The server needs to wait for the slow clients to upload the parameters if all N clients participate in aggregation, which is time expensive. These slow clients are regarded as stragglers. Li et al.[15] explained that the full client participation requirement in local SGD suffers from a serious "straggler effect". Meanwhile, Wang et al.[20] pointed out that the local SGD or FedAvg estimate does not converge to θ^* if the number of local iterations is nonidentical. Hence, it is inappropriate to estimate the confidence interval of θ^* via local SGD or FedAvg when the number of local iterations is nonidentical on non-IID data in federated learning.

Our research aims to give a confidence interval of θ^* in an online fashion when the number of local iterations is nonidentical. We denote the global parameter at the tth round by θ_t and the averaged path by $\bar{\theta}_T = \sum_{t=1}^{T} \theta_t$, where T is the max number of rounds. Since it is impossible to use FedAvg, we perform confidence interval estimation via rescaled FedAvg which is a special case of FedNova. Wang et al.[20] gave the convergence guarantee for FedNova with a constant learning rate. However, they did not analyze the statistical properties. In our research, we give a nonasymptotic convergence rate of the rescaled FedAvg estimate and prove that θ_T converges θ^* in L_2. Moreover, we give the asymptotic distribution of the averaged estimate $\bar{\theta}_T$. Furthermore, we propose the rescaled plug-in method to estimate the confidence interval of θ^* in an online fashion. In summary, this work makes the following contributions:

First, under certain regularity conditions, we prove that the rescaled FedAvg estimate θ_T is a consistent estimate of θ^*, and give a nonasymptotic convergence rate for the estimate.

Second, we prove that $\bar{\theta}_T = \frac{1}{T}\sum_{t=1}^{T} \theta_t$ is asymptotically normal under some regularity conditions. Our research shows that the asymptotic covariance of $\bar{\theta}_T$ is inversely proportional to the client participation rate ν.

Third, we propose the separated plug-in method to construct a confidence interval of θ^* on non-IID data when the number of local iterations is nonidentical. Additionally, we experimentally prove the effectiveness of the method.

The rest of this paper is organized as follows. Section 2 begins with some general definitions and notations used throughout the paper and introduces the rescaled FedAvg algorithm. In Section 3, we start by stating some assumptions essential to our theoretical proofs. Then, we analyze the statistical properties of the rescaled FedAvg estimator and propose an online confidence interval estimation method. In Section 4, we investigate the empirical performance of the proposed method by numerical simulation. Section 5 gives the conclusions of this research.

2 Problem formulation

Throughout this paper, we use the following notations. For a vector $a \in \mathbf{R}^p$, $\|a\|$ is defined to be the vector L_2 norm

DOI: 10.52396/JUSTC-2022-0179
JUSTC, 2023, 53(11): 1103

$\|a\| = \left(\sum_{i=1}^{p} a_i^2\right)^{1/2}$. For vectors $x, y \in \mathbf{R}^p$, the inner product is defined as $\langle x, y\rangle = \sum_{i=1}^{p} x_i y_i$. For matrix $\boldsymbol{A} \in \mathbf{R}^{p\times p}$, $\|\boldsymbol{A}\|$ is the operator norm of $\boldsymbol{A}$, i.e., $\|\boldsymbol{A}\| = \sup_{\|x\|\leqslant 1} \|\boldsymbol{A}x\|$ where $x \in \mathbf{R}^p$. For positive sequences $\{x_n\}$ and $\{y_n\}$, $x_n = O(y_n)$ indicates that there exists a constant C such that $x_n \leqslant Cy_n$ for large n. In addition, $x_n = o(y_n)$ represents $\lim_n x_n/y_n = 0$. For stochastic sequences $\{\alpha_n\}$ and $\{\beta_n\}$, $\beta_n = o_P(\alpha_n)$ means β_n/α_n converges to 0 in probability. Moreover, we use $\xrightarrow{\mathrm{d}}$ to denote convergence in distribution, use $\xrightarrow{\mathrm{p}}$ to denote convergence in probability, and use $\xrightarrow{\mathrm{a.s.}}$ to denote almost sure convergence.

We consider the problem of performing confidence interval estimation for the federated heterogeneous optimization problem. The "heterogeneous" here means the number of local iterations is different across clients. The federated heterogeneous optimization aims to solve the following problem:

$$\theta^* = \arg\min_{\theta\in\Theta} R(\theta),$$

on non-IID data when the number of local iterations is non-identical. In this research, we suppose that the samples arrive one by one in an online fashion at each client, which is the same as Refs. [27, 29].

In the tth communication round, suppose that there are E_i samples arriving sequentially at client i within a given wall-clock time interval. At the ith client, these samples are denoted by $z_{t,0}^i, z_{t,1}^i, \cdots, z_{t,E_i-1}^i$. The sample $z_{t,k}^i$ is an input/output pair $(x_{t,k}^i, y_{t,k}^i)$. As the samples arrive sequentially, the ith client updates as the following formula:

$$\theta_{t,k+1}^i = \theta_{t,k}^i - \eta_t \nabla l_i(\theta_{t,k}^i; z_{t,k}^i), \quad k = 0, 1, \cdots, E_i - 1,$$

where $\theta_{t,k}^i$ denotes the local model parameter of client i after the kth local update in the tth round. By convention, $\theta_{t,0}^i = \theta_t^i$. The learning rate in the tth round is η_t. FedAvg simply aggregates local parameter updates by averaging at the end of each round. The global parameter is updated as $\theta_{t+1} = \theta_t - \frac{1}{\nu}\sum_{i\in\mathcal{S}_t} p_i \Delta_t^i$. Here, $\Delta_t^i = \theta_{t,E_i}^i - \theta_{t,0}^i$ is the local parameter update of client i in the tth round. As mentioned before, only a subset of clients update their local models in a round. The set of these selected clients in the tth round is denoted by $\mathcal{S}_t$. The constant ν is the fraction of updated clients. Therefore, the number of selected clients is $K = \lfloor \nu N \rfloor$. These clients are selected without replacement from $[N]$ with probability $\{p_1, \cdots, p_N\}$.

For instance, the global risk function in the linear model is

$$R(\theta) = \sum_{i=1}^{N} p_i \mathbb{E}_{z^i\sim\rho_i}(\theta^\top x^i - y^i)^2,$$

where $z^i = (x^i, y^i)$ is the input/output pair. Suppose that the true local parameter of client i is θ_i^*, which means $y^i = (\theta_i^*)^\top x^i + \varepsilon^i$, where ε^i is random noise with zero mean. Consequently, the minimizer of $R(\theta)$ is $\theta^* = \frac{1}{N}\sum_{i=1}^{N} \theta_i^*$. Wang et al.[20] has shown that the global estimate θ_T converges to $\theta' = \sum_{i=1}^{N} \frac{p_i E_i}{\sum_{j=1}^{N} p_j E_j} \theta_i^*$. However, $\theta^* = \theta'$ if and only if $E_i = E_j$ for all $i, j \in [N]$. Hence, there is inconsistency in estimating θ^* using FedAvg. It is not reasonable to construct a confidence interval using this estimate because it is not unbiased or consistent.

It is not effective to perform confidence interval estimation via a federated averaging algorithm due to the inconsistency caused by non-IID data and heterogeneity in the number of local iterations. To overcome this ineffectiveness, we have to perform confidence interval estimation via new federated algorithms.

To eliminate the inconsistency, Algorithm 1 rescales the local parameter updates and tries updating the global model parameter by averaging the rescaled local parameter updates:

$$\theta_{t+1} = \theta_t - \frac{1}{\nu}\sum_{i=1}^{N} p_i \frac{\bar{E}}{E_i} \Delta_t^i,$$

where $\bar{E}$ is the average of $\{E_i\}_1^N$. We call this algorithm rescaled federated averaging (rescaled FedAvg). Actually, it is a special case of FedNova[20] which updates local parameters by stochastic gradient descent. Although the convergence of FedNova has been guaranteed, the statistical properties remain unexplored. Li et al.[28] allows E_i's to grow with the communication rounds. From their work, growing E_i's converge faster than fixed E_i's in terms of communication round. If the number of samples is the same, increasing E_i's will reduce the communication round. In addition, it will enlarge the heterogeneity in the number of local iterations and slow down convergence. Thus, we set the number of local steps E_i's to be identical across different rounds. In the next section, we give a nonasymptotic convergence rate of the resclaed FedAvg

Algorithm 1. Rescaled federated averaging

Input: Initial point θ_0, client participation rate ν, learning rates $\{\eta_t\}_0^T$;
for $t = 0$ to $T-1$ **do**
  Sample subset $\mathcal{S}_t$ of $K = \lfloor \nu N \rfloor$ clients from $[N]$;
  Communicate θ_t to all client $i \in \mathcal{S}_t$;
  for client $i \in \mathcal{S}_t$ in parallel **do**
    Initialize local parameters: $\theta_{t,0}^i = \theta_t$;
    for $k = 0$ to $E_i - 1$ **do**
      After the sample $z_{t,k}^i$ arriving, update local parameters:
      $\theta_{t,k+1}^i = \theta_{t,k}^i - \eta_t \nabla l_i(\theta_{t,k}^i; z_{t,k}^i)$;
    end
    Local updates: $\Delta_t^i = \theta_{t,E_i}^i - \theta_t$;
    /* $\bar{E}$ `is the average of` $\{E_i\}_1^N$ */
    Communicate $\frac{\bar{E}}{E_i}\Delta_t^i$ to the central server;
  end
  On the central parameter server, update global parameters:
  $\theta_{t+1} = \theta_t + \frac{1}{\nu}\sum_{i\in\mathcal{S}_t} \frac{p_i \bar{E}}{E_i} \Delta_t^i$;
end
Output: θ_T and averaged rescaled FedAvg estimator $\bar{\theta}_T = \frac{1}{T}\sum_{t=1}^{T} \theta_t$

DOI: 10.52396/JUSTC-2022-0179
JUSTC, **2023**, 53(11): 1103

algorithm. In addition, we prove the asymptotic normality of the averaged rescaled FedAvg estimate $\bar{\theta}_T$.

Therefore, we construct confidence intervals online based on Rescaled FedAvg in this research.

3 Theoretical results

In this section, we first introduce some assumptions that are essential to our theoretical proofs. Second, we will prove that the rescaled FedAvg estimate is consistent. Then, we propose an online confidence interval estimation method called separated plug-in. This method leverages the asymptotic normality of the averaged rescaled FedAvg estimate.

Assumption 1. For each client $i \in [N]$, assume that the loss function $l_i(\cdot)$ is continuously differentiable. At client i, assume that $\mathbb{E}\left[\left\|\nabla l_i(\theta_{t,k}^i; z_{t,k}^i)\right\|^2\right] \leqslant G$, where $0 \leqslant k < E_i$ and $0 \leqslant t \leqslant T$.

Assumption 1 is also assumed in Ref. [15]. Assumption 2 is standard, and is widely assumed in many papers[30, 31].

Assumption 2. Assume that the risk functions $\{R_i(\cdot)\}_{i=1}^N$ are μ-strongly convex. The loss function $l_i(\cdot)$ is assumed to be L-average smooth, i.e., for any vectors $\theta_1, \theta_2 \in \Theta$,

$$\mathbb{E}\left\|\nabla l_i(\theta_2; z^i) - \nabla l_i(\theta_1; z^i)\right\|^2 \leqslant L^2 \|\theta_2 - \theta_1\|^2,$$

where z^i is an independent realization from ρ_i.

By Jensen's inequality, $\{R_i(\cdot)\}_{i=1}^N$ are all L-smooth. The functions $\{R_i(\cdot)\}_{i=1}^N$ are μ-strongly convex and L-smooth by Assumption 2. Global risk $R(\theta)$ is also L-smooth and μ-strongly convex because it is a linear combination of $\{R_i(\cdot)\}_{i=1}^N$.

The next assumption considers the Lipschitz continuity of $\nabla^2 R_i(\cdot)$ in the neighborhood of θ^*.

Assumption 3. Assume that the Hessian matrix of $R_i(\cdot)$ exists and that there exist some $\delta_1 > 0$ and $L' > 0$ such that for all $i \in [N]$

$$\|\nabla^2 R_i(\theta) - \nabla^2 R_i(\theta^*)\| \leqslant L' \|\theta - \theta^*\|,$$

whenever $\|\theta - \theta^*\| < \delta_1$.

Define $\varepsilon_i(\theta) = \nabla l_i(\theta; z^i) - \nabla R_i(\theta)$ to be the gradient noise at the ith client, where z^i is an independent realization from ρ_i. Note that $\mathbb{E}[\varepsilon_i(\theta)] = 0$ for all $\theta \in \Theta$. The covariance of the gradient noise at θ^* is $\mathbb{E}[\varepsilon_i(\theta^*)\varepsilon_i(\theta^*)^\top]$. We denote it by $\boldsymbol{S}_i$. Denote the Hessian matrix $\nabla^2 R(\theta^*)$ by $\boldsymbol{H}$.

Next, we assume that the difference between the covariance of $\varepsilon_i(\theta)$ and $\boldsymbol{S}_i$ is bounded by the quadratic polynomial of $\|\theta - \theta^*\|$. Thus, it ensures the continuity of the covariance of $\varepsilon_i(\cdot)$ at θ^*. The term $\|\theta - \theta^*\|^2$ controls the growth speed of the covariance. The boundedness of the $(2 + \delta_2)$ moment of gradient noise is first assumed by Ref. [28].

Assumption 4. Assume that there exists some constant $L_0 > 0$ such that for every $i \in [N]$,

$$\|\mathbb{E}(\varepsilon_i(\theta)\varepsilon_i(\theta)^\top) - \boldsymbol{S}_i\| \leqslant L_0(\|\theta - \theta^*\| + \|\theta - \theta^*\|^2).$$

Moreover, suppose that there exists a constant $\delta_2 > 0$ such that $\sup_\theta \mathbb{E}\|\varepsilon(\theta)\|^{2+\delta_2}$ is finite.

Assumption 5. There exists some constant C such that for all $i \in [N]$,

$$\mathbb{E}\|\nabla^2 l_i(\theta; z^i) - \nabla^2 l_i(\theta^*; z^i)\| \leqslant C\|\theta - \theta^*\|,$$

where z^i is an independent realization from ρ_i.

3.1 Statistical properties

When the learning rates $\{\eta_t\}_0^\infty$ satisfy some conditions, Li et al.[28] established a nonasymptotic convergence rate $\mathbb{E}\|\theta_t - \theta^*\|^2 \leqslant C_0\eta_t$ for local SGD, where C_0 is a constant. Our results give a convergence rate $O(T^{-\alpha})$ for rescaled FedAvg when the learning rates are $\eta_t = \beta t^{-\alpha}$, where $\alpha \in (0.5, 1)$ and $\beta > 2/\mu$. Furthermore, our results show how the difference in the number of local iterations influences the convergence.

In the SGD and local SGD algorithms and their variants, decreasing learning rates are critical. Li et al.[15] has shown that FedAvg with a fixed learning rate does not converge to θ^*. To reach the minimum, the decay of learning rates is essential in FedAvg. The convergence to the optimal of local SGD is guaranteed with a fixed learning rate when $E_i = 1$ for all $i \in [N]$[32]. However, the FedAvg estimate θ_T does not converge to θ^* if $E_i > 1$. Hence, the learning rates are decreasing and satisfy some regularized conditions in our analysis. We give the following theorems with decreasing learning rates.

Theorem 1. Let the learning rates be $\eta_t = \beta t^{-\alpha}$ with $\alpha \in (0.5, 1)$, where $\beta > 2/\mu$. Under Assumptions 1–4, there exists a constant t_0 such that whenever $t \geqslant t_0$

$$\mathbb{E}\|\theta_t - \theta^*\|^2 \leqslant c_0\eta_t,$$

where $c_0 = \frac{2}{\mu}\max\left\{\frac{2\beta^2}{\mu\beta - 2}B, t_0^\alpha \mathbb{E}\Delta_{t_0}\right\}$, $B = \frac{GL}{\nu} + \frac{CL^2\tau}{2N\nu}$, $\tau = \sum_{i=1}^N E_i^2$, $\mathbb{E}\Delta_{t_0} = \mathbb{E}[R(\theta_{t_0}) - R(\theta^*)]$, and C is a constant.

Theorem 1 in Wang et al.[20] has shown that $\min_{t\in[T]}\mathbb{E}\|\nabla R(\theta_t)\|^2$ converges to 0 with a fixed learning rate determined by T. In addition, it has proven that the convergence rate is $O(T^{-1/2})$. Nevertheless, this cannot ensure the consistency of θ_t. Compared with that, Theorem 1 showed that $\mathbb{E}\|\theta_t - \theta^*\|^2$ converges to 0 with decaying learning rates. The convergence rate is $O(T^{-\alpha})$. Moreover, our theorem is under the assumption of convexity while their work can be applied to nonconvex cases.

Theorem 1 implies that the convergence of the estimate is related to ν and $\{E_i\}_{i=1}^N$. When ν is large, the convergence is fast. Intuitively, a large ν indicates that more clients paticipate in aggregation. Consequently, the estimate converges faster with more information. From Theorem 1, τ is a good measure to quantify the heterogeneity in the number of local iterations when $\bar{E}$ (the average of $\{E_i\}_1^N$) is fixed. With smaller τ, the estimate θ_t converges faster. When there is no heterogeneity in the number of local iterations, θ_t converges the fastest.

In a recent work, Toulis and Airoldi[33] proposed the implicit SGD procedures and analyzed the asymptotic distribution of averaged implicit SGD iterations. Similarly, Li et al.[28] analyzed the asymptotic distribution of the averaged local SGD iterates on non-IID data and performed confidence interval estimation using the asymptotic distribution. We formulate the asymptotic normality of the averaged rescaled FedAvg estimate $\bar{\theta}_T$ in the following theorem.

Theorem 2. If the learning rates are the same as those in

DOI: 10.52396/JUSTC-2022-0179
JUSTC, 2023, 53(11): 1103

Theorem 1 and Assumptions 1–4 hold, the averaged rescaled FedAvg estimate $\bar{\theta}_T$ is asymptotically normal,

$$\sqrt{T}(\bar{\theta}_T-\theta^*)\xrightarrow{\mathrm{d}}\boldsymbol{N}_p\Big(0,\frac{1}{\nu}\boldsymbol{H}^{-1}\boldsymbol{S}^*\boldsymbol{H}^{-1}\Big),$$

where $\boldsymbol{H}=\nabla^2R(\theta^*)$, $\boldsymbol{S}^*=\sum_{i=1}^N\frac{p_i^2}{E_i}\boldsymbol{S}_i$, $\boldsymbol{S}_i=\mathbb{E}[\varepsilon_i(\theta^*)\varepsilon_i(\theta^*)^\top]$, and ν is the client participation rate.

Theorem 2 reveals that $\bar{\theta}_T$ is asymptotically normal with an asymptotic covariance depending on $\{E_i\}_1^N$, local gradient noise variance $\{\boldsymbol{S}_i\}_0^N$, and Hessian matrix $\boldsymbol{H}=\nabla^2R(\theta^*)$. As shown in Ref. [28], the effect of data heterogeneity does not appear in the asymptotic distribution. However, this theorem shows that the heterogeneity in the number of local iterations appears in the asymptotic distribution.

In the federated learning system, the central server only has access to the global iterates $\{\theta_t\}_0^T$. To construct a valid confidence interval, we may leverage the asymptotic distribution of averaged estimator $\bar{\theta}_T$ under the federated learning constraints.

3.2 Separated plug-in method

To perform confidence interval estimation, a good and explicit estimate of the asymptotic covariance is necessary. There are several methods[26, 27, 34, 35] to estimate the asymptotic covariance in the SGD statistical inference problem. However, these methods cannot be directly applied in federated settings.

From Theorem 2, the asymptotic covariance is determined by the second order derivative $\boldsymbol{H}$ and the covariance of stochastic gradient error $\boldsymbol{S}_i$. Note that $\boldsymbol{S}^*$ is different from the covariance of $\varepsilon(\theta^*)$. Hence, we separately estimate $\boldsymbol{S}^*$ and $\boldsymbol{H}$ to construct an estimate of the asymptotic covariance instead of estimating the asymptotic covariance directly. More precisely, we can separately estimate each $\boldsymbol{S}_i$ by some estimate $\hat{\boldsymbol{S}}_i$ and use $\sum_{i=1}^N\frac{p_i^2}{E_i}\hat{\boldsymbol{S}}_i$ as an estimate of $\boldsymbol{S}^*$.

We assume that $\{z_{t,k}^i\}_{0\leqslant k<E_i,0\leqslant t\leqslant T}$ are IID from the local data distribution ρ_i. In the tth round, the input/output pairs $z_{t,0}^i,z_{t,1}^i,\cdots,z_{t,E_i-1}^i$ arrive at client i sequentially, which simulates real-world data streams such as mobile phone data and IoT device data. In addition, we assume that z_t^i from distribution ρ_i is independent with z_t^j from distribution ρ_j if $i\neq j$.

Since θ_T converges to θ^* in probability, an intuitive way to estimate $\boldsymbol{H}$ is to use the sample estimate

$$\boldsymbol{H}_T=\frac{1}{\nu T}\sum_{t=1}^T\sum_{i\in\mathcal{S}_t}p_i\nabla^2l_i(\theta_t;z_t^i),\tag{2}$$

where $\mathcal{S}_t$ is the set consisting of clients participating in the tth aggregation step and ν is the fraction of participating clients. Due to partial client participation, information about the second-order derivatives of all clients is not always accessible. For this reason, we only have access to the derivatives of clients that are selected in the current round. In fact, the probability of each client being selected in a round is ν. The expectation of the number that a client has been chosen in the training process is νT.

The estimation of $\boldsymbol{S}^*$ is similar to the estimation of $\boldsymbol{H}$ but there are some additional problems in estimating $\boldsymbol{S}^*$. First, we rewrite $\boldsymbol{S}_i$ as

$$\boldsymbol{S}_i=\mathbb{E}[\nabla l_i(\theta^*;z^i)][\nabla l_i(\theta^*;z^i)]^\top-[\nabla R_i(\theta)][\nabla R_i(\theta)]^\top.\tag{3}$$

Due to the features of federated learning, we cannot directly estimate $\boldsymbol{S}^*$. An estimate of $\boldsymbol{S}^*$ can be obtained by combining estimates of $\boldsymbol{S}_i$. As is known before, partial participation makes direct estimation of $\nabla R(\theta)$ difficult. More importantly, it is infeasible to calculate the expectation of local loss functions. We then estimate $\boldsymbol{S}_i$ by

$$\boldsymbol{S}_{T,i}=\frac{1}{\nu T}\sum_{t=1}^T[\nabla l_i(\theta_t;z_t^i)][\nabla l_i(\theta_t;z_t^i)]^\top\mathbb{I}(i\in\mathcal{S}_t)-[g_{T,i}][g_{T,i}]^\top,\tag{4}$$

where $g_{T,i}=\frac{1}{\nu T}\sum_{t=1}^T\nabla l_i(\theta_t;z_t^i)\mathbb{I}(i\in\mathcal{S}_t)$, and $\{z_t^i\}$ are independent realizations from ρ_i. An advantage of the above estimate (4) is that it can be updated online, which is the main purpose of rewriting $\boldsymbol{S}_i$.

Since $\boldsymbol{S}^*$ is a linear combination of $\{\boldsymbol{S}_i\}_1^N$, it is natural to estimate $\boldsymbol{S}^*$ by

$$\boldsymbol{S}_T=\sum_{i=1}^N\frac{p_i^2}{E_i}\boldsymbol{S}_{T,i},\tag{5}$$

where $\boldsymbol{S}_{T,i}$ is estimated in Eq. (4).

From the above discussion, $\boldsymbol{H}_T$ and $\boldsymbol{S}_T$ are estimates of $\boldsymbol{H}$ and $\boldsymbol{S}^*$, respectively. Furthermore, $\boldsymbol{H}_T$ and $\boldsymbol{S}_T$ can be updated recursively in the spirit of SGD. Next, we prove the consistency of the two estimates under some additional assumptions in addition to Assumptions 1–4. Assumption 5 assumes that the second-order derivatives of local risk functions $\{R_i(\cdot)\}_1^N$ in the neighborhood of θ^* are Lipschitz continuous. This assumption is critical in the following theorem.

Theorem 3. Under Assumptions 1–5, $\boldsymbol{H}_T\xrightarrow{\mathrm{p}}\boldsymbol{H}$ and $\boldsymbol{S}_{T,i}\xrightarrow{\mathrm{p}}\boldsymbol{S}_i$ for all $i\in[N]$. Hence, $\boldsymbol{S}_T$ converges to $\boldsymbol{S}^*$ in probability, and $\boldsymbol{H}_T^{-1}\boldsymbol{S}_T\boldsymbol{H}_T^{-1}$ converges to $\boldsymbol{H}^{-1}\boldsymbol{S}^*\boldsymbol{H}^{-1}$ in probability.

Although we cannot give an exact confidence interval of θ^*, we can form an asymptotic confidence interval by Theorem 3. Denote $\hat{\sigma}_{T,j}^2=(\boldsymbol{H}_T^{-1}\boldsymbol{S}_T\boldsymbol{H}_T^{-1})_{j,j}$ and θ_j^* jth coordinate of θ^*. From Theorem 3, we can directly derive Corollary 1. Based on Corollary 1, we proposed a new online confidence interval estimation method. Since we separately estimate $\boldsymbol{S}_i$, we call this method the separated plug-in. Details about the algoroithm are in Algorithm 2.

Corollary 1. Under the same assumptions as Theorem 3,

$$\mathbb{P}\Big(\bar{\theta}_{T,j}-\frac{z_{\alpha/2}}{\sqrt{\nu T}}\hat{\sigma}_{T,j}\leqslant\theta_j^*\leqslant\bar{\theta}_{T,j}+\frac{z_{\alpha/2}}{\sqrt{\nu T}}\hat{\sigma}_{T,j}\Big)\to1-\alpha,$$

where $z_{\alpha/2}$ is the $1-\alpha/2$ quantile of the standard normal distribution, and $\bar{\theta}_{T,j}$ is the jth coordinate of $\bar{\theta}_T$.

When $E_i=E$ for all $i\in[N]$ and $\nu=1$, there is no need to estimate $\boldsymbol{S}^*$ separately. In this case, the covariance becomes $\frac{1}{E}\boldsymbol{H}^{-1}\boldsymbol{S}\boldsymbol{H}^{-1}$, where $\boldsymbol{S}$ is the covariance of gradient noise. This is the same as that in Ref. [28].

3.3 Other methods

In addition to the separated plug-in, we extend the random scaling method proposed by Lee et al.[36] and extended to the federated setting proposed by Li et al[28]. We theoretically prove that the random scaling method is also effective in our setting. In the previous subsection, we propose the separated

DOI: 10.52396/JUSTC-2022-0179
JUSTC, 2023, 53(11): 1103

Algorithm 2. Separated plug-in method

Input: Participation rate ν, number of clients N, learning rates $\{\eta_t\}_1^T$.

for $t=1$ **to** T **do**

Sample a subset of clients from $[N]$ denoted by $\mathcal{S}_t$ with $|\mathcal{S}_t|=\lfloor \nu T\rfloor$;

for each client $i\in\mathcal{S}_t$ in parallel **do**

Initialize local parameters $\theta^i_{t,0}=\theta_t$;

Update estimators:

$$\boldsymbol{S}^i_t=\boldsymbol{S}^i_{t-1}+[\nabla l_i(\theta_t;z^i)][\nabla l_i(\theta_t;z^i)]^\top;$$
$$\boldsymbol{H}_{t,i}=\boldsymbol{H}_{t-1,i}+\nabla^2 l_i(\theta_t;z^i);$$
$$g_{t,i}=\frac{t-1}{t}g_{t-1,i}+\frac{1}{t}\nabla l_i(\theta_t;z^i);$$

for $k=0$ **to** E_i-1 **do**

Update $\theta^i_{t,k+1}=\theta^i_{t,k}-\eta_t\nabla l_i(\theta^i_{t,k};z^i_{t,k})$;

end

Send parameter update $\Delta^i_t=\frac{E}{E_i}(\theta^i_{t,E_i}-\theta_t)$ to server;

end

Server update global parameter by $\theta_{t+1}=\theta_t+\sum_{i\in\mathcal{S}_t}\Delta^i_t$;

Update $\bar\theta_t=\frac{t-1}{t}\bar\theta_{t-1}+\frac{1}{t}\theta_t$;

end

Send $\bar\theta_T$, $\boldsymbol{S}^i_T$, $\boldsymbol{H}_{T,i}$, and $g_{T,i}$ to the server. Compute $\boldsymbol{H}_T$ and $\boldsymbol{S}_T$ following Eqs. (2) and (5);

Output: $\bar\theta_T=\frac{1}{T}\sum_{t=1}^T\theta_t$, $\boldsymbol{H}_T$, and $\boldsymbol{S}_T$;

plug-in method and give the statistical properties of the proposed estimate θ_T and $\bar\theta_T$. In fact, the asymptotic normality of $\bar\theta_T$ can be extended to a more general form. The following theorem is a general case of Theorem 2.

Theorem 4. Under the same assumptions as Theorem 1, the following random function weakly converges to a scaled Brownian motion as $T\to\infty$, i.e.,

$$\frac{1}{\sqrt{T}}\sum_{t=1}^{\lfloor rT\rfloor}(\theta_t-\theta^*)\xrightarrow{\mathrm{d}}\frac{1}{\sqrt{\nu}}\boldsymbol{H}^{-1}\boldsymbol{S}^{*1/2}\boldsymbol{B}_p(r),$$

where $\boldsymbol{B}_p(\cdot)$ denotes p-dimensional standard Brownian motion, $\boldsymbol{S}^*$ is the same as that in Theorem 2, $\boldsymbol{H}=\nabla^2 R(\theta^*)$ and ν is the client participation rate.

Based on Theorem 4, we can then construct a confidence interval using an online procedure the same as Algorithm 2 in Ref. [28].

4 Numerical studies

The numerical studies are divided into three parts. In the first part, we will research how the heterogeneity in the number of local iterations influences the convergence on simulated data. In the second part, we compare the separated plug-in method with the random scaling method to show the effectiveness of the proposed method on simulated data. In the last part, we show how to use the proposed method on two real datasets.

4.1 Effect of the heterogeneity in the number of local iterations

The first simulation experiment shows how the heterogeneity in the number of local iterations influences the convergence in linear regression. The feature space here is $\mathcal{X}=\mathbf{R}^p$, and the label space is $\mathcal{Y}=\mathbf{R}$, where $p=5$. For simplicity, we do not consider intercepts. Hence, the parameter space is $\Theta=\mathbf{R}^p$. We set the learning rates as $\eta_t=0.2t^{-0.505}$ in this experiment.

At client i, the true model parameter θ^*_i is generated from $\boldsymbol{N}_p(0,I_p)$. The input/output pair $z^i_{t,k}=(x^i_{t,k},y^i_{t,k})$ is a realization from local data distribution ρ_i. Here, $x^i_{t,k}$ is generated from multivariate normal distribution $\boldsymbol{N}_p(0,I_p)$, and the response $y^i_{t,k}$ is generated according to the linear model $y^i_{t,k}=(\theta^*_i)^\top x^i_{t,k}+\varepsilon$ where $\varepsilon\sim\boldsymbol{N}(0,1)$. The average number of local iterations is fixed at 4. Instead, we set different degrees of heterogeneity in the number of local iterations. We set three different degrees of heterogeneity ("Balance", "Small", "Large"). Note that we have a measure $\tau=\sum_{i=1}^N E_i^2$ to quantify the heterogeneity in the number of local iterations. In the "Balance" case, the number of local steps is $E_i=4$ for all $i\in[N]$. The measure is 160 in the "Balance" case. In the "Small" case, the number of local iterations $\{E_i\}$ is IID from a discrete uniform distribution on $\{3,4,5\}$. The expected measure is $\mathbb{E}\tau=500/3$ in this case. In the "Large" case, the number of local steps $\{E_i\}$ is IID from a discrete uniform distribution on $\{1,2,3,4,5,6,7\}$, and the measure is $\mathbb{E}\tau=200$ in this case. The results are shown in Fig. 1.

From Fig. 1, $\bar\theta_T$ converges the fastest in the "Balance" case. In the "Large" case, $\mathbb{E}\tau$ is the largest in the three cases and the estimate converges the most slowly. This indicates that the heterogeneity in the number of local iterations slows down convergence. The empirical results coincide with the theoretical results in Theorem 1.

4.2 Separated plug-in and random scaling

In the second simulation experiment, we show the effectiveness of the proposed separated plug-in method. In the experiment, the nominal coverage probability of the confidence intervals is 95% in both linear regression and logistic regression models. The learning rates are set to $\eta_t=0.2t^{-0.505}$ in linear regression and $\eta_t=0.5t^{-0.505}$ in logistic regression, which is fine tuned in advance. Here, $z^i_{t,k}=(x^i_{t,k},y^i_{t,k})$ is an input/output pair. The predictors $x^i_{t,k}$ is generated from

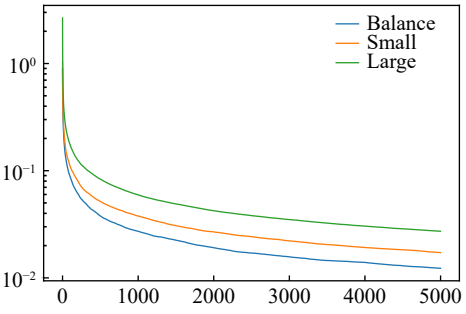


Fig. 1. Impacts of τ based on 1000 replications. The x-axis and y-axis are the number of rounds and $\|\bar\theta_T-\theta^*\|$, respectively. "Balance" means identical number of local iterations, where $E_i=4$ for all $i\in[N]$ and $\tau=160$. "Small" means a small degree of heterogeneity in the number of local iterations, where E_i is IID from a discrete uniform distribution on $\{3,4,5\}$, and $\mathbb{E}\tau=500/3$. "Large" represents a large degree of heterogeneity in the number of local iterations, where E_i is IID from a discrete uniform distribution on $\{1,2,3,4,5,6,7\}$, and $\mathbb{E}\tau=200$.

DOI: 10.52396/JUSTC-2022-0179
JUSTC, **2023**, 53(11): 1103

Table 1. Separated plug-in method in linear regression based on 1000 replications. In brackets are the standard deviations.

Items	ν	$T = 1000$	$T = 2000$	$T = 3000$	$T = 4000$	$T = 5000$
Coverage rate	0.2	0.934	0.942	0.950	0.949	0.952
	0.3	0.958	0.944	0.953	0.955	0.952
	0.4	0.948	0.934	0.943	0.957	0.949
	0.5	0.953	0.956	0.951	0.949	0.951
Average radius ($\times 10^{-2}$)	0.2	10.830 (0.616)	7.587 (0.297)	6.116 (0.196)	5.325 (0.149)	4.757 (0.120)
	0.3	8.725 (0.377)	6.140 (0.193)	5.004 (0.131)	4.328 (0.097)	3.867 (0.077)
	0.4	7.532 (0.298)	5.299 (0.148)	4.319 (0.098)	3.736 (0.072)	3.340 (0.058)
	0.5	6.712 (0.223)	4.733 (0.114)	3.860 (0.075)	3.339 (0.057)	2.985 (0.045)

Table 2. Random scaling method in linear regression based on 1000 replications. In brackets are the standard deviations.

Items	ν	$T = 1000$	$T = 2000$	$T = 3000$	$T = 4000$	$T = 5000$
Coverage rate	0.2	0.955	0.937	0.934	0.942	0.952
	0.3	0.937	0.948	0.944	0.944	0.944
	0.4	0.938	0.948	0.948	0.951	0.944
	0.5	0.940	0.935	0.946	0.952	0.955
Average radius ($\times 10^{-2}$)	0.2	16.430 (8.498)	10.960 (4.857)	8.749 (3.605)	7.507 (3.157)	6.536 (2.734)
	0.3	11.790 (5.347)	8.297 (3.604)	6.760 (2.816)	5.754 (2.384)	5.145 (2.193)
	0.4	9.865 (4.311)	6.866 (2.904)	5.678 (2.228)	4.874 (2.051)	4.318 (1.797)
	0.5	8.665 (3.556)	6.056 (2.538)	4.980 (1.982)	4.313 (1.739)	3.851 (1.503)

$\boldsymbol{N}_p(0, I_p)$ and the response $y_{t,k}^i$ is generated according to the local model, where $p = 5$. Assume that samples from different clients are independent, and that samples from the same client are IID. Details about the data generating and parameters generating approach are as follows:

(Ⅰ) In the linear regression, the response at each client is generated according to $y_{t,k}^i = (\theta_i^*)^\top x_{t,k}^i + \varepsilon_{t,k}^i$, where $\varepsilon_{t,k}^i$ is IID from $\boldsymbol{N}(0,1)$. The true local parameters θ_i^* and $x_{t,k}^i$ are both generated from $\boldsymbol{N}(0, I_p)$. In this case, the minimizer θ^* is the average of $\{\theta_i^*\}_{i=1}^N$.

(Ⅱ) In the logistic regression, the response $y_{t,k}^i$ is generated to be 1 with probability $\sigma((\theta_i^*)^\top x_{t,k}^i)$ and 0 with probability $1 - \sigma((\theta_i^*)^\top x_{t,k}^i)$. The true local model parameter θ_i^* is also generated from $\boldsymbol{N}(0, I_p)$. The first 5 clients have parameter $\theta^{(1)}$, and the rest have the same parameter $\theta^{(2)}$. To calculate the empirical coverage rate, we have to precisely compute the minimizer θ^*. We use the stochastic gradient descent method to iteratively estimate θ^* in a centralized setting on a dataset that is a mixture of data from client 1 and client 6. Half of the dataset is from client 1, and the rest is from client 6.

In both linear and logistic regression, we generate $\{E_i\}_1^N$ from a discrete uniform distribution on $\{1, \cdots, 5\}$. Naturally, the parameter space is $\Theta = \mathbf{R}^p$ since we do not have intercepts in linear regression and omit the constant term in logistic regression here.

In both cases, the coverage rate and the average radius of the confidence intervals are two main aspects used to evaluate the effectiveness of the methods. The coverage rate and the average radius is computed by averaging based on 1000 replications.

Linear regression: Tables 1 and 2 show the empirical performance of the two methods with different participation rates and different maximum numbers of rounds T. Both random scaling and separated plug-in have good performance in the linear regression model with different participation rates.

From the two tables, the average radius of the separated plug-in method is smaller than that of random scaling. Furthermore, the standard deviation of the plug-in confidence interval radius is much smaller than that of random scaling. This is because the plug-in method takes advantage of the first and second derivative information, while random scaling only uses first-order derivatives.

From Table 1, the average radius decreases as T increases. Similarly, the average radius is smaller with more clients participating. For instance, the average length of $\nu = 0.2$ is approximately 1.4 times the average radius of $\nu = 0.4$. This is consistent with the theoretical results. The two methods have similar average coverage rates under linear regression.

Logistic regression: In Tables 3 and 4, the plug-in method also has a smaller average radius and standard deviation than those of random scaling. The random scaling and separated plug-in methods also have similar coverage rates. The coverage rates of both separated plug-in and random scaling are close to the nominal coverage rate of 95% under the logistic regression model. In the logistic regression model, the average radius of the confidence interval also decreases as T and ν increase.

The above experiments showed that the two methods are efficient and applicable in this problem. The separated plug-in method constructs a smaller confidence interval, and the standard deviation of the radius is smaller than that of random scaling. In addition, the two methods have approximate

DOI: 10.52396/JUSTC-2022-0179
JUSTC, **2023**, 53(11): 1103

Table 3. Plug-in method in logistic regression based on 1000 replications. In brackets are the standard deviations.

Items	ν	$T = 1000$	$T = 2000$	$T = 3000$	$T = 4000$	$T = 5000$
Coverage rate	0.2	0.944	0.949	0.939	0.955	0.946
	0.3	0.938	0.942	0.945	0.945	0.958
	0.4	0.941	0.942	0.952	0.955	0.953
	0.5	0.956	0.944	0.954	0.944	0.956
Average radius ($\times 10^{-2}$)	0.2	8.248 (0.287)	5.773 (0.135)	4.690 (0.088)	4.051 (0.064)	3.616 (0.051)
	0.3	6.651 (0.179)	4.668 (0.088)	3.800 (0.059)	3.286 (0.043)	2.935 (0.035)
	0.4	5.718 (0.135)	4.027 (0.064)	3.280 (0.044)	2.837 (0.033)	2.536 (0.026)
	0.5	5.087 (0.109)	3.587 (0.051)	2.924 (0.034)	2.531 (0.026)	2.009 (0.020)

Table 4. Random scaling method in logistic regression based on 1000 replications. In brackets are the standard deviations.

Items	ν	$T = 1000$	$T = 2000$	$T = 3000$	$T = 4000$	$T = 5000$
Coverage rate	0.2	0.941	0.944	0.938	0.953	0.951
	0.3	0.942	0.948	0.947	0.945	0.951
	0.4	0.953	0.958	0.949	0.952	0.952
	0.5	0.945	0.944	0.944	0.947	0.948
Average radius ($\times 10^{-2}$)	0.2	9.904 (4.834)	7.090 (3.315)	5.876 (2.759)	5.099 (2.364)	4.563 (2.075)
	0.3	7.819 (3.713)	5.868 (2.770)	4.836 (2.220)	4.170 (1.875)	3.691 (1.619)
	0.4	6.924 (3.392)	5.003 (2.282)	4.104 (1.805)	3.551 (1.502)	3.232 (1.406)
	0.5	5.993 (2.809)	4.213 (1.889)	3.506 (1.556)	3.139 (1.405)	2.852 (1.269)

empirical coverage rates. Hence, the separated plug-in is better than random scaling considering the empirical coverage rate and averaged radius.

4.3 Real data applications

In this section, we apply our proposed methods to conduct confidence interval estimation in linear regression for the power consumption of the Tetuan city dataset①. In logistic regression, we conduct confidence interval estimation for the Skin Segmentation dataset②.

The power consumption of the Tetuan city dataset[37] is related to the power consumption of three different distribution networks of Tetuan city, which is located in northern Morocco. This dataset consists of 52416 samples. We fit a federated linear model to investigate how the variables "temperature", "humidity", "wind speed", "general diffuse flows", and "diffuse flows" influence the response variable "Zone 1 Power Consumption". To simulate the non-IIDness and heterogeneity in the number of local iterations, we allocate the dataset into 10 clients according to the variable "DateTime". The time of a day is divided into 10 parts: Hour 0–1, Hour 2–4, Hour 5–5, Hour 6–9, Hour 10–11, Hour 12–13, Hour 14–16, Hour 17–17, Hour 18–19, and Hour 20–23. Each client only possesses samples in spcific hours. For example, the 1st client only have samples in Hour 0–1.

The Skin Segmentation dataset is constructed over B, G, R color space. Skin and Nonskin dataset is generated using skin textures from face images of diversity of age, gender, and race people. This dataset has 245057 samples and each sample labels with "skin" or "nonskin", out of which 50859 is the "skin" samples and 194198 is "nonskin" samples. We fit a logistic model to observe the relationship between the indicator of skin and the three predictors: B, G, and R. The dataset is also partitioned into 10 parts, each part is related to a client. The datasize at each client is 20000, 30000, 10000, 40000, 4000, 20000, 30000, 10000, 20000, 45057.

Corresponding to the related hours, the number of local iterations is 2, 3, 1, 4, 2, 2, 2, 3, 1, 2, and 4 in linear regression. So the total number of rounds is $T = 2184$. The learning rates are still $\eta_t = 0.2t^{-0.505}$. For logistic regression, we follow the same setting in the previous simulated data experiment. The learning rates are set to be $\eta_t = 0.5t^{-0.505}$. The total number of rounds is $T = 5000$ in this case, and the number of local iterations is 4, 6, 2, 8, 8, 4, 6, 2, 4, and 9, in accordance. The participation rate is $\nu = 0.5$ for both cases to get better performance according to the previous simulated data experiment. The results of our real data analysis are shown in Tables 5 and 6

From Table 5, we see that the power consumption in Zone 1 is greatly influenced by the "temperature". From Table 6, we conclude that the variable B is positively related to the response and the other two variables G and R are negatively related to the response.

5 Conclusions

This study shows how to perform online confidence interval estimation for federated heterogeneous optimization

① The dataset is at http://archive.ics.uci.edu/ml/datasets/Power+consumption+of+Tetouan+city.

② The Skin Segmentation data is at https://archive.ics.uci.edu/ml/datasets/skin+segmentation.

DOI: 10.52396/JUSTC-2022-0179
JUSTC, **2023**, 53(11): 1103

Table 5. Point estimates and 95% confidence intervals (CI) for the power consumption of the Tetuan city dataset for separated plug-in (Sepa.) and random scaling (Rand.) methods in linear regression.

Variable	Estimate	Sepa. CI	Rand. CI
Temperature	0.4807	(0.4657, 0.4957)	(0.3283, 0.6331)
Humidity	0.0440	(0.0312, 0.0568)	(0.0307, 0.0573)
Wind speed	0.0547	(0.0410, 0.0684)	(0.0264, 0.0830)
General diffuse flows	−0.0360	(−0.0493, −0.0227)	(−0.1320, 0.0600)
Diffuse flows	−0.1005	(−0.1125, −0.0885)	(−0.1867, −0.0143)

Table 6. Point estimates and 95% confidence intervals (CI) for the Skin Segmentation dataset for separated plug-in (Sepa.) and random scaling (Rand.) methods in logistic regression.

Variable	Estimate	Sepa. CI	Rand. CI
B	0.8758	(0.7087, 1.0429)	(0.2439, 1.5077)
G	−0.3485	(−0.5160, −0.1810)	(−1.1258, 0.4288)
R	−0.3864	(−0.4922, −0.2806)	(−1.1616, 0.3888)

problems. We first proposed the rescaled FedAvg to estimate θ^*. The research gives a nonasymptotic convergence rate of the estimate. This result also revealed that the heterogeneity in the number of local iterations slows down the convergence. Furthermore, we proved that the averaged rescaled FedAvg estimate is asymptotically normal with unknown covariance. Based on its normality, we proposed the separated plug-in method to estimate the asymptotic covariance. The separated plug-in method estimates the covariances of local gradients separately and constructs an estimate of the asymptotic covariance matrix by these estimates. Additionally, we have proven a functional CLT and applied it to extend the random scaling method to the federated heterogeneous setting. Finally, the simulation showed that the heterogeneity in the number of local iterations slows down the convergence and investigated the empirical performance of the two methods via the Monte-Carlo experiment. The simulation results have shown that the plug-in interval has a smaller radius than the random scaling interval, and is more stable. From the experiment, the average length of the confidence interval decreases when performing more aggregations. In addition, the average length and its variance will decay if there are more clients participating in the aggregation step each round.

Supporting information

The supporting information for this article can be found online at https://doi.org/10.52396/JUSTC-2022-0179. It includes proofs of all theorems.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (12171451, 71873128).

Conflict of interest

The authors declare that they have no conflict of interest.

Biographies

Yu Wang is currently a graduate student at the University of Science and Technology of China. His research mainly focuses on federated learning statistical machine learning.

Wenquan Cui is currently an Associate Professor at the University of Science and Technology of China (USTC). He received his Ph.D. degree in Statistics from USTC in 2004. His research mainly focuses on survival analysis, high-dimensional statistical inference, and statistical machine learning.

Jianjun Xu received his Ph.D. degree in Statistics from the University of Science and Technology of China in 2022. His research mainly focuses on functional data analysis.

References

[1] Hastie T, Friedman J, Tibshirani R. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. New York: Springer, **2001**.
[2] Berk R A. Statistical Learning from a Regression Perspective. New York: Springer, **2008**.
[3] James G, Witten D, Hastie T, et al. An Introduction to Statistical Learning: With Applications in R. New York: Springer, **2013**.
[4] Li T, Sahu A K, Talwalkar A, et al. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine,* **2020**, *37* (3): 50–60.
[5] McMahan B, Moore E, Ramage D, et al. Communication-efficient learning of deep networks from decentralized data. In: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS) 2017. Fort Lauderdale, FL: PMLR, **2017**: 1273–1282.
[6] Preuveneers D, Rimmer V, Tsingenopoulos I, et al. Chained anomaly detection models for federated learning: An intrusion detection case study. *Applied Sciences,* **2018**, *8* (12): 2663.
[7] Mothukuri V, Khare P, Parizi R M, et al. Federated-learning-based anomaly detection for IoT security attacks. *IEEE Internet of Things Journal,* **2021**, *9* (4): 2545–2554.
[8] Amiri M M, Gündüz D. Federated learning over wireless fading channels. *IEEE Transactions on Wireless Communications,* **2020**, *19*: 3546–3557.
[9] Bharadhwaj H. Meta-learning for user cold-start recommendation. In: 2019 International Joint Conference on Neural Networks (IJCNN). Budapest, Hungary: IEEE, **2019**: 1–8.
[10] Chen S, Xue D, Chuai G, et al. FL-QSAR: A federated learning-based QSAR prototype for collaborative drug discovery. *Bioinformatics,* **2021**, *36*: 5492–5498.
[11] Tarcar A K. Advancing healthcare solutions with federated learning. In: Federated Learning. Cham, Switzerland: Springer, **2022**: 499–508.
[12] Zhao Y, Li M, Lai L, et al. Federated learning with non-IID data. arXiv: 1806.00582, **2018**.
[13] Zhang W, Wang X, Zhou P, et al. Client selection for federated learning with non-IID data in mobile edge computing. *IEEE Access,* **2021**, *9*: 24462–24474.
[14] Briggs C, Fan Z, Andras P. Federated learning with hierarchical clustering of local updates to improve training on non-IID data. In: 2020 International Joint Conference on Neural Networks (IJCNN). Glasgow, UK: IEEE, **2020**: 1–9.
[15] Li X, Huang K, Yang W, et al. On the convergence of FedAvg on non-IID data. In: 2020 International Conference on Learning Representations. Appleton, WI: ICLR, **2020**.
[16] Stich S U. Local SGD converges fast and communicates little. arXiv: 1805.09767, **2018**.

DOI: 10.52396/JUSTC-2022-0179
JUSTC, **2023**, 53(11): 1103

[17] Zhou F, Cong G. On the convergence properties of a *K*-step averaging stochastic gradient descent algorithm for nonconvex optimization. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence. Palo Alto, CA: AAAI Press, **2018**: 3219–3227.

[18] Wang S, Tuor T, Salonidis T, et al. Adaptive federated learning in resource constrained edge computing systems. *IEEE Journal on Selected Areas in Communications,* **2019**, *37* (6): 1205–1221.

[19] Yu H, Jin R, Yang S. On the linear speedup analysis of communication efficient momentum SGD for distributed non-convex optimization. In: Proceedings of the 36th International Conference on Machine Learning. Long Beach, CA: PMLR, **2019**: 7184–7193.

[20] Wang J, Liu Q, Liang H, et al. Tackling the objective inconsistency problem in heterogeneous federated optimization. In: Advances in Neural Information Processing Systems 33 (NeurIPS 2020). Red Hook, NY: Curran Associates, Inc., **2020**, 33: 7611–7623.

[21] Li T, Sahu A K, Zaheer M, et al. Federated optimization in heterogeneous networks. In: Proceedings of Machine Learning and Systems 2020 (MLSys 2020). Austin, TX: mlsys.org, **2020**, 2: 429–450.

[22] Karimireddy S P, Kale S, Mohri M, et al. SCAFFOLD: Stochastic controlled averaging for federated learning. In: Proceedings of the 37th International Conference on Machine Learning. Online: PMLR, **2020**: 5132–5143.

[23] Liang X, Shen S, Liu J, et al. Variance reduced local SGD with lower communication complexity. arXiv: 1912.12844, **2019**.

[24] Ruppert D. Efficient estimations from a slowly convergent Robbins–Monro process. Ithaca, New York: Cornell University, **1988**.

[25] Polyak B T, Juditsky A B. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization,* **1992**, *30* (4): 838–855.

[26] Zhu W, Chen X, Wu W. Online covariance matrix estimation in stochastic gradient descent. *Journal of the American Statistical Association,* **2021**, *118*: 393–404.

[27] Fang Y, Xu J, Yang L. Online bootstrap confidence intervals for the stochastic gradient descent estimator. *The Journal of Machine Learning Research,* **2018**, *19* (1): 3053–3073.

[28] Li X, Liang J, Chang X, et al. Statistical estimation and online inference via local SGD. In: Proceedings of Thirty Fifth Conference on Learning Theory. London: PMLR, **2022**: 1613–1661.

[29] Su W J, Zhu Y. Uncertainty quantification for online learning and stochastic approximation via hierarchical incremental gradient descent. arXiv: 1802.04876, **2018**.

[30] Reisizadeh A, Mokhtari A, Hassani H, et al. FedPAQ: A communication-efficient federated learning method with periodic averaging and quantization. In: Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS) 2020. Palermo, Italy: PMLR, **2020**, 108: 2021–2031.

[31] Khaled A, Mishchenko K, Richtárik P. First analysis of local GD on heterogeneous data. arXiv: 1909.04715, **2019**.

[32] Nesterov Y. Introductory Lectures on Convex Optimization: A Basic Course. New York: Springer, **2003**.

[33] Toulis P, Airoldi E M. Asymptotic and finite-sample properties of estimators based on stochastic gradients. *The Annals of Statistics,* **2017**, *45* (4): 1694–1727.

[34] Chen X, Lee J D, Tong X T, et al. Statistical inference for model parameters in stochastic gradient descent. *The Annals of Statistics,* **2020**, *48* (1): 251–273.

[35] Liu R, Yuan M, Shang Z. Online statistical inference for parameters estimation with linear-equality constraints. *Journal of Multivariate Analysis,* **2022**, *191*: 105017.

[36] Lee S, Liao Y, Seo M H, et al. Fast and robust online inference with stochastic gradient descent via random scaling. *Proceedings of the AAAI Conference on Artificial Intelligence,* **2022**, *36* (7): 7381–7389.

[37] Salam A, El Hibaoui A. Comparison of machine learning algorithms for the power consumption prediction: Case study of Tetouan city. In: 2018 6th International Renewable and Sustainable Energy Conference (IRSEC). Rabat, Morocco: IEEE, **2018**.

[38] Hall P, Heyde C C. Martingale Limit Theory and Its Application. New York: Academic Press, **2014**.

DOI: 10.52396/JUSTC-2022-0179
JUSTC, **2023**, 53(11): 1103