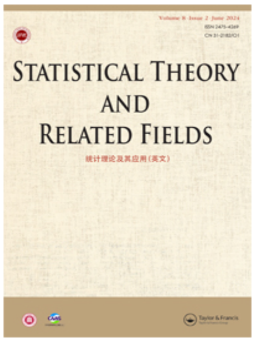

Statistical Theory and Related Fields

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/tstf20

Communication-efficient distributed statistical inference on zero-inflated Poisson models

Ran Wan & Yang Bai

To cite this article: Ran Wan & Yang Bai (2024) Communication-efficient distributed statistical inference on zero-inflated Poisson models, Statistical Theory and Related Fields, 8:2, 81-106, DOI: 10.1080/24754269.2023.2263721

To link to this article: https://doi.org/10.1080/24754269.2023.2263721

Published online: 30 Oct 2023.

STATISTICAL THEORY AND RELATED FIELDS
2024, VOL. 8, NO. 2, 81–106
https://doi.org/10.1080/24754269.2023.2263721

Taylor & Francis
Taylor & Francis Group

OPEN ACCESS Check for updates

Communication-efficient distributed statistical inference on zero-inflated Poisson models

Ran Wan and Yang Bai

School of Statistics and Management, Shanghai University of Finance and Economics, Shanghai, People's Republic of China

ABSTRACT

Zero-inflated count outcomes are common in many studies, such as counting claim frequency in the insurance industry in which identifying and understanding excessive zeros are of interest. Moreover, with the progress of data collecting and storage techniques, the amount of data is too massive to be stored or processed by a single node or branch. Hence, to develop distributed data analysis is blossoming. In this paper, several communication-efficient distributed zero-inflated Poisson regression algorithms are developed to analyse such kind of large-scale zero-inflated data. Both asymptotic properties of the proposed estimators and algorithm complexities are well studied and conducted. Various simulation studies demonstrate that our proposed method and algorithm work well and efficiently. Finally, in the case study, we apply our proposed algorithms to car insurance data from Kaggle.

ARTICLE HISTORY
Received 12 February 2023
Revised 8 September 2023
Accepted 21 September 2023

KEYWORDS
Zero-inflated count;
distributed EM algorithm;
communication-efficient

1. Introduction

For the analysis of count data, many zero-inflated regression models have been developed. These models are designed to deal with situations where there is an 'excessive' number of individuals with a count of 0. For example, in a car insurance study where the dependent variable is 'number in an insurance period a policyholder has claimed', the vast majority of policyholders may have a value of 0 ('zero' means certain policyholder has no insurance claims). The reason for zero-inflation is twofold. First, a large proportion of insurance policies are subject to deductible excess, which means only if the loss exceeds some given amount the insurance company will pay the claims. Second, in the car insurance, there exists 'no claim discount (NCD)', that is, if the applicant does not claim in the current insurance period, he will enjoy a certain discount in the next insurance period, which will make some policyholders give up the claim in order to enjoy the discount in next period. Based on this situation, it is of great practical significance to identify and understand those inflated zeros. Commonly used counted distributions are problematic and require appropriate model assumptions to characterize such zero-inflation structure. Zero-inflated Poisson (ZIP) distribution is a popular method to deal with such problems. This distribution is a special case of finite mixture distributions. Specifically, it takes probability p' at 0 and takes probability $1 - p'$ at Poisson distribution with parameter λ (denoted as Poisson (λ)). When p' is greater than 0, the data will contain more zeros than the ordinary Poisson distribution, showing zero-inflation.

ZIP distribution was first presented by Cohen (1963) and Johnson and Kotz (1970). Then, a ZIP regression model was developed by Lambert (1992) to study manufacturing defect problem. EM algorithm was used to estimate the parameters, avoiding the computational problems caused by simultaneous estimating parameters related to p' and λ. In addition, the algorithm and theoretical properties were discussed according to whether p' and λ were dependent or not, respectively. Hall (2000) proposed ZIP and Zero-inflated Binomial (ZIB) regressions with random effects and derived corresponding estimation algorithms. In order to deal with the zero-inflated outcomes with more complex correlations, Lee et al. (2006) presented a multi-level ZIP model and studied the parameter estimating method. Tang et al. (2014) discussed the application of ZIP regression to risk factor selection in the context of insurance industries. Adaptive lasso-based EM algorithm was developed to process parameter estimation.

Hall (2000), Lee et al. (2006) and Tang et al. (2014) gave the applications of ZIP regression under different scenarios and corresponding algorithms, but above algorithms are based on data stored in a single institution. With the growth of data volume and emergence of distributed data, it is getting more and more urgent to propose ZIP regression algorithm based on distributed data, but the relevant research is still deficient.

CONTACT Yang Bai statbyang@mail.shufe.edu.cn School of Statistics and Management, Shanghai University of Finance and Economics, 777 Guoding Road, Shanghai 200433, People's Republic of China

This article has been corrected with minor changes. These changes do not impact the academic content of the article.

With the progress of techniques, distributed structure is common in the storage of data. There are two main reasons for this structure. First, the amount of data exceeds the storage limit of a single institution, so it has to be stored distributedly. Second, because the process of data collection is completed by different countries and regions, the data naturally forms a distributed structure. In the first case, it is difficult to consolidate data from different institutions. In the second case, sometimes the data of different institutions cannot be directly exchanged or combined for the purpose of privacy protection, etc. In these cases, distributed algorithms are needed to process the data, so that the purpose of data analysis can be achieved without merging data. To evaluate a distributed algorithm, communication cost is an important element. Because of this, how to reduce the communication cost of distributed algorithm is also a hot topic.

In recent years, there have been some studies on communication-efficient distributed algorithms and their theoretical properties. Shamir et al. (2014) presented a distributed Newton-type optimization method, which had linear converging properties when the objective function had a quadratic structure. Mota et al. (2013) presented a distributed ADMM algorithm for separable optimization in node network structures. The algorithm converged when the network is bipartite or the loss function is strongly convex. The theoretical properties of these two algorithms require the strong convexity of the objective function, which is too harsh in some cases. Jordan et al. (2018) presented a distributed algorithm with efficient communication. The algorithm was used in a wide range of scenarios and had relatively weak convergence conditions. In low-dimensional M-estimation scenarios, the objective function only needed to be locally convex, which is our motivation of the proposed Algorithm 2. Zhu et al. (2021) presented a distributed least squares approximation method (DLSA) which could deal with a kind of regression problem. By approximating the local objective function using a local quadratic form, they were able to obtain a combined estimator by taking a weighted average of local estimators. The estimator had the same statistical efficiency as the global estimator. In this paper, this idea is consulted as an extended idea to construct the distributed algorithm with high communication efficiency Algorithm 3.

In this paper, in order to reduce the difficulty of calculation in solving the maximized likelihood function, latent variables are introduced to divide the parameters into two parts. EM algorithm is an effective algorithm to optimize objective function with latent variables. Dempster et al. (1977) applied EM algorithm to calculate MLE with incomplete data. They gave the general form of E step, M step, the theoretical properties of EM algorithm and scenarios in which it could be applied. Wu (1983) pointed out the problem in the proof of Dempster et al. (1977) and studied two more general convergence forms of EM algorithm. In addition to the theoretical properties, there were also plenty of research on EM algorithm applications. Redner and Walker (1984) investigated the implementation of the EM algorithm in the context of the problem of parameter estimation in mixture density, and its theoretical properties, especially when the mixed components come from exponential distribution family. For distributed EM algorithm, sensing network is often used as the research background, for example, Nowak (2003) and Gu (2008). This kind of distributed EM algorithm usually has some special properties because of sensing network, such as decentralization and connectivity of only adjacent nodes, which is not consistent with the research scene in this paper. There are relatively few researches on EM algorithm applied to traditional distributed scenarios.

According to the derivation results in this paper, E step of EM algorithm can be completed only with the data of the local institution, so there is no need to propose a distributed algorithm. However, M step of EM algorithm requires the data of all institutions, so a distributed algorithm needs to be proposed. In view of this, we first present the distributed EM Algorithm (Algorithm 1) for M step and analyse the communication cost of this algorithm. Further, motivated by two ideas, the communication cost of Algorithm 1 is decreased, and distributed algorithms with high communication efficiency are given. Specifically, the improvement of Algorithm 2 is to reduce the communication cost of each internal iteration of M step, and the improvement of Algorithm 3 is to reduce the number of internal iterations of M step to one-shot. Besides these, we also present the asymptotic analysis of Algorithm 2. In general, our contributions are twofold. First, we present three distributed algorithms for ZIP regression. And two of them are communication efficient. Second, we give the theoretical properties of Algorithm 2 and compare the time computation complexity and transmission cost of the three algorithms. Third, we fully compare three algorithms in simulations, and then summarize the applicable scenarios of each algorithm, which is of practical significance.

The rest of the paper is as follows. Section 2 gives the main algorithm of this paper: the EM algorithm of ZIP regression, the distributed EM algorithm (Algorithm 1) and the communication-efficient distributed EM algorithm motivated by Jordan et al. (2018) (Algorithm 2). Section 3 mainly introduces the theoretical results of Algorithm 2. Section 4 presents the second communication-efficient distributed algorithm (Algorithm 3) motivated by Zhu et al. (2021). In Section 5, various simulations are provided to verify the proposed Algorithms 1–3. The algorithms are then applied to the car insurance data in Section 6. The article is concluded with a short discussion in Section 7.

2. Model and method

This section introduces the ZIP regression model, EM algorithm to solve it and distributed version of the algorithm. Combining with the characteristics of distributed computing, we then propose an improved algorithm for more efficient communication.

2.1. Model

Assume that independent responses $\{Y_i : i = 1, \dots, n\}$ are from the ZIP distribution as follows:

$$Y_i \sim \begin{cases} 0, & \text{with probability } p_i, \\ \text{Poisson}\left(\lambda_i\right), & \text{with probability } 1 - p_i. \end{cases} \tag{1}$$

Simple calculation yields the distribution function of Y_i as

$$\begin{cases} \mathrm{P}\left(Y_i = 0\right) = p_i + \left(1 - p_i\right) \mathrm{e}^{-\lambda_i}, \\ \mathrm{P}\left(Y_i = k\right) = \left(1 - p_i\right) \mathrm{e}^{-\lambda_i} \lambda_i^k / k!, \quad k = 1, 2, \dots. \end{cases} \tag{2}$$

We see that $\mathrm{P}(Y_i = 0) > \mathrm{e}^{-\lambda_i}$ when $p_i > 0$, which indicates zero-inflation.

Based on some early work (Lambert, 1992), the parameters p_i and λ_i can be modelled by a logistic regression model and log-linear model as follows:

$$\begin{cases} \operatorname{logit}\left(p_i\right) & = \log\left\{p_i / \left(1 - p_i\right)\right\} = \boldsymbol{z}_i^\top \boldsymbol{\gamma}, \\ \log\left(\lambda_i\right) & = \boldsymbol{x}_i^\top \boldsymbol{\beta}, \end{cases} \tag{3}$$

where $\{\boldsymbol{z}_i\}, \{\boldsymbol{x}_i\}, i = 1, \dots, n$ are two vectors of covariates with respect to observation i, with dimensions p and q, respectively. These two vectors can either be the same or different. $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_p)^\top$ and $\boldsymbol{\beta} = (\beta_1, \dots, \beta_q)^\top$ are corresponding coefficient vectors.

According to the assumptions above, we can give the likelihood function as

$$\begin{aligned} L(\boldsymbol{\gamma}, \boldsymbol{\beta}) &= \log\left\{\Pi_{i=1}^n \mathrm{P}\left(Y_i = y_i \mid \boldsymbol{z}_i, \boldsymbol{x}_i\right)\right\} \\ &= \sum_{y_i = 0} \log\left\{\mathrm{e}^{\boldsymbol{z}_i^\top \boldsymbol{\gamma}} + \exp\left(-\mathrm{e}^{\boldsymbol{x}_i^\top \boldsymbol{\beta}}\right)\right\} + \sum_{y_i > 0}\left(y_i \boldsymbol{x}_i^\top \boldsymbol{\beta} - \mathrm{e}^{\boldsymbol{x}_i^\top \boldsymbol{\beta}}\right) \\ &\quad - \sum_{i=1}^n \log\left(1 + \mathrm{e}^{\boldsymbol{z}_i^\top \boldsymbol{\gamma}}\right) - \sum_{y_i > 0} \log\left(y_i!\right) \\ &\doteq \sum_{i=1}^n l\left(y_i, \boldsymbol{x}_i, \boldsymbol{z}_i, \boldsymbol{\gamma}, \boldsymbol{\beta}\right). \end{aligned} \tag{4}$$

Optimizing this function directly will meet great trouble, especially as the first part of the likelihood includes both $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$. First, the responses are from a mixture distribution, which includes the parameters p_i and λ_i. Second, regression models are designed for both parameters. Therefore, the computation is quite challenging. We consider using EM algorithm to optimize it and the next section will give the specific implementation.

2.2. EM algorithm

The EM algorithm for the ZIP regression model was firstly introduced in early literature by Lambert (1992). The EM algorithm is based on a latent variable $U = I$ (Y from zero state), which indicates the response is either from zero state or Poisson state. The distribution of U is

$$U_i = \begin{cases} 1, & \text{with probability } p_i, \\ 0, & \text{with probability } 1 - p_i. \end{cases} \tag{5}$$

According to conditional expection,

$$\begin{aligned} &\mathrm{P}\left(Y_i = y_i, U_i = u_i \mid \boldsymbol{x}_i, \boldsymbol{z}_i, \boldsymbol{\beta}, \boldsymbol{\gamma}\right) \\ &\quad = \mathrm{P}\left(Y_i = y_i \mid U_i = u_i, \boldsymbol{x}_i, \boldsymbol{z}_i, \boldsymbol{\beta}, \boldsymbol{\gamma}\right) \end{aligned}$$

$$\times \mathrm{P}\left(U_i = u_i \,|\, \boldsymbol{x}_i, \boldsymbol{z}_i, \boldsymbol{\beta}, \boldsymbol{\gamma}\right)$$
$$= \frac{\left(\mathrm{e}^{\boldsymbol{z}_i^\top \boldsymbol{\gamma}}\right)^{u_i}}{1 + \mathrm{e}^{\boldsymbol{z}_i^\top \boldsymbol{\gamma}}} \left(\frac{\mathrm{e}^{y_i \boldsymbol{x}_i^\top \boldsymbol{\beta} - \mathrm{e}^{X_i^\top}}}{y_i!}\right)^{1-u_i}. \tag{6}$$

The log-likelihood function based on (Y, U) is

$$\begin{aligned}
L_c(\boldsymbol{\gamma}, \boldsymbol{\beta}) &= \log\left[\prod_{i=1}^{n}\left\{\mathrm{P}\left(Y_i = y_i \,|\, U_i = u_i, \boldsymbol{x}_i, \boldsymbol{z}_i, \boldsymbol{\beta}, \boldsymbol{\gamma}\right) \times \mathrm{P}\left(U_i = u_i \,|\, \boldsymbol{x}_i, \boldsymbol{z}_i, \boldsymbol{\beta}, \boldsymbol{\gamma}\right)\right\}\right] \\
&= \sum_{i=1}^{n} \log\left\{\mathrm{P}\left(Y_i = y_i | U_i = u_i, x_i, z_i, \boldsymbol{\beta}, \boldsymbol{\gamma}\right)\right\} \\
&\quad + \sum_{i=1}^{n} \log\left\{\mathrm{P}\left(U_i = u_i | \boldsymbol{x}_i, \boldsymbol{z}_i, \boldsymbol{\beta}, \boldsymbol{\gamma}\right)\right\} \\
&= \sum_{i=1}^{n}\left\{u_i \boldsymbol{z}_i^\top \boldsymbol{\gamma} - \log\left(1 + \mathrm{e}^{\boldsymbol{z}_i^\top \boldsymbol{\gamma}}\right)\right\} + \sum_{i=1}^{n}(1 - u_i)\left(y_i \boldsymbol{x}_i^\top \boldsymbol{\beta} - \mathrm{e}^{\boldsymbol{x}_i^\top \boldsymbol{\beta}}\right) \\
&\quad - \sum_{i=1}^{n}(1 - u_i)\log\left(y_i!\right).
\end{aligned} \tag{7}$$

The parts of the above formula concerning the parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are defined as

$$L_{c,1}(\boldsymbol{\gamma}) = \sum_{i=1}^{n}\left\{u_i \boldsymbol{z}_i^\top \boldsymbol{\gamma} - \log\left(1 + \mathrm{e}^{\boldsymbol{z}_i^\top \boldsymbol{\gamma}}\right)\right\}, \tag{8}$$

$$L_{c,2}(\boldsymbol{\beta}) = \sum_{i=1}^{n}(1 - u_i)\left(y_i \boldsymbol{x}_i^\top \boldsymbol{\beta} - \mathrm{e}^{\boldsymbol{x}_i^\top \boldsymbol{\beta}}\right). \tag{9}$$

Now, we can optimize the above two functions with respect to $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$, respectively.

Based on the objective function (7), the $(k + 1)$ step of the EM algorithm is as follows.

E step: Based on $\boldsymbol{\gamma}^{(k)}$ and $\boldsymbol{\beta}^{(k)}$, estimate $U_i^{(k)}$ using its posterior mean.

$$\begin{aligned}
U_i^{(k)} &= E\left(U_i \,|\, y_i, \boldsymbol{\gamma}^{(k)}, \boldsymbol{\beta}^{(k)}\right) \\
&= \mathrm{P}\left(U_i = 1 \,|\, y_i, \boldsymbol{\gamma}^{(k)}, \boldsymbol{\beta}^{(k)}\right) \\
&= \mathrm{P}\left(Y_i = y_i \,|\, U_i = 1\right)\mathrm{P}\left(U_i = 1\right) \times \left\{\mathrm{P}\left(Y_i = y_i \,|\, U_i = 1\right)\mathrm{P}\left(U_i = 1\right)\right. \\
&\quad \left. + \mathrm{P}\left(Y_i = y_i \,|\, U_i = 0\right)\mathrm{P}\left(U_i = 0\right)\right\}^{-1} \\
&= \begin{cases} \left[1 + \exp\left(-\mathrm{e}^{\boldsymbol{x}_i^\top \boldsymbol{\beta}^{(k)}} - \boldsymbol{z}_i^\top \boldsymbol{\gamma}^{(k)}\right)\right]^{-1}, & \text{if } y_i = 0, \\ 0, & \text{if } y_i = 1, 2, \ldots. \end{cases}
\end{aligned} \tag{10}$$

M step: Based on the results of E step, $U_i^{(k)}$is substituted into Equations (8) and (9) to calculate parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$, respectively.

$$\boldsymbol{\gamma}^{(k+1)} = \underset{\boldsymbol{\gamma}}{\operatorname{argmin}}\left\{-L_{c,1}(\boldsymbol{\gamma})\right\}, \tag{11}$$

$$\boldsymbol{\beta}^{(k+1)} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\left\{-L_{c,2}(\boldsymbol{\beta})\right\}. \tag{12}$$

Newton's algorithm is used to solve Equations (11) and (12). The algorithm iteration expression is as follows, where t represents the iteration step in Newton's algorithm:

$$\boldsymbol{\gamma}^{(k,t+1)} = \boldsymbol{\gamma}^{(k,t)} - \left[\sum_{i=1}^{n}\frac{\mathrm{e}^{\boldsymbol{z}_i^\top \boldsymbol{\gamma}^{(k,t)}}}{\left(1 + \mathrm{e}^{\boldsymbol{z}_i^\top \boldsymbol{\gamma}^{(k,t)}}\right)^2} \cdot \boldsymbol{z}_i \boldsymbol{z}_i^\top\right]^{-1}\left[\sum_{i=1}^{n}\left(\frac{\mathrm{e}^{\boldsymbol{z}_i^\top \boldsymbol{\gamma}^{(k,t)}}}{1 + \mathrm{e}^{\boldsymbol{z}_i^\top \boldsymbol{\gamma}^{(k,t)}}} - u_i^{(k)}\right) \boldsymbol{z}_i\right], \tag{13}$$

$$\boldsymbol{\beta}^{(k,t+1)} = \boldsymbol{\beta}^{(k,t)} - \left[\sum_{i=1}^{n} \left(1 - u_i^{(k)}\right) \mathrm{e}^{\boldsymbol{x}_i^\top \boldsymbol{\beta}^{(k,t)}} \cdot \boldsymbol{x}_i \boldsymbol{x}_i^\top\right]^{-1} \left[\sum_{i=1}^{n} \left(1 - u_i^{(k)}\right) \left(\mathrm{e}^{\boldsymbol{x}_i^\top \boldsymbol{\beta}^{(k,t)}} - y_i\right) \boldsymbol{x}_i\right]. \tag{14}$$

After the number of Newton iterations meets the requirements (denoted as T times), the final iteration result is denoted as the updated parameter, i.e., $\boldsymbol{\gamma}^{(k,T)} := \boldsymbol{\gamma}^{(k+1)}$, $\boldsymbol{\beta}^{(k,T)} := \boldsymbol{\beta}^{(k+1)}$. Note that $\boldsymbol{\gamma}^{(k+1)}$ and $\boldsymbol{\beta}^{(k+1)}$ are the approximate optimization result of Equations (11) and (12). Here, for simplicity, without introducing a new notation, we use the same notations $\boldsymbol{\gamma}^{(k+1)}$ and $\boldsymbol{\beta}^{(k+1)}$. And we maximize Equation (7) based on $u_i = I(y_i = 0)$ to get the initial values of $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ for EM algorithm.

2.3. Distributed EM algorithm

The distributed structure of data is introduced first. Responses $Y = \{y_1, \ldots, y_n\}$, covariates $Z = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n\}$, $X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ horizontally distributed across J institutions, where institution j's sample size is n_j, $n_j = O(a)$, and $a = n/J$, i.e., $c_1 \le \min_j n_j/n \le \max_j n_j/n \le c_2$, in which c_1, c_2 are two positive constants, $j = 1, \ldots, J$, $\sum_{j=1}^{J} n_j = n$. This section introduces a distributed implementation for EM algorithm (Equations (10), (13) and (14)). Note that for the sake of illustration, $\boldsymbol{z}_i, \boldsymbol{x}_i, \boldsymbol{u}_i, \boldsymbol{y}_i, i = 1, \ldots, n$ are re-denoted as double subscript $\boldsymbol{z}_{ji}, \boldsymbol{x}_{ji}, \boldsymbol{u}_{ji}, \boldsymbol{y}_{ji}, j = 1, \ldots, J, i = 1, \ldots, n_j$.

The realization of Equation (10) does not need to cross different institutions, meaning that it can be completed directly in the local institution. However, the realization of Equations (13) and (14) is cross-institution, and a distributed algorithm needs to be proposed. The specific algorithm is as follows.

Algorithm 1 Distributed EM Algorithm

```
Initialize: Let $U_{ji}^{(0)} = I(y_{ji} = 0)$, and every institution calculates Equations (11) and (12) by its own data. Take average to get $\boldsymbol{\gamma}^{(0)}$ and $\boldsymbol{\beta}^{(0)}$ on central institution and transmit the results to institution $j = 1, \ldots, J$.
While $\|\boldsymbol{\gamma}^{(k+1)} - \boldsymbol{\gamma}^{(k)}\|_2 \ge \delta$ or $\|\boldsymbol{\beta}^{(k+1)} - \boldsymbol{\beta}^{(k)}\|_2 \ge \delta$, $k$ is the kth EM iteration.
    E step: For institution $j = 1, \ldots, J$, compute $U_{ji}^{(k)}$ by (10) , $i = 1, \ldots, n_j$.
    M step: Internal Newton iteration ($\boldsymbol{\gamma}^{(k,0)} = \boldsymbol{\gamma}^{(k-1)}$,$\boldsymbol{\beta}^{(k,0)} = \boldsymbol{\beta}^{(k-1)}$):
        For institution $j = 1, \ldots, J$, $t = 1, \ldots, T$ compute
            $T_{1j}^{(k,t)} = \sum_{i=1}^{n_j} \frac{\mathrm{e}^{\boldsymbol{z}_{ji}^\top \boldsymbol{\gamma}^{(k,t)}}}{\left(1+\mathrm{e}^{\boldsymbol{z}_{ji}^\top \boldsymbol{\gamma}^{(k,t)}}\right)^2} \cdot \boldsymbol{z}_{ji}\boldsymbol{z}_{ji}^\top$,
            $T_{2j}^{(k,t)} = \sum_{i=1}^{n_j} \left(\frac{\mathrm{e}^{\boldsymbol{z}_{ji}^\top \boldsymbol{\gamma}^{(k,t)}}}{1+\mathrm{e}^{\boldsymbol{z}_{ji}^\top \boldsymbol{\gamma}^{(k,t)}}} - u_{ji}^{(k)}\right) \boldsymbol{z}_{ji}$,
            $T_{3j}^{(k,t)} = \sum_{i=1}^{n_j} \left(1 - u_{ji}^{(k)}\right) \mathrm{e}^{\boldsymbol{x}_{ji}^\top \boldsymbol{\beta}^{(k,t)}} \cdot \boldsymbol{x}_{ji}\boldsymbol{x}_{ji}^\top$,
            $T_{4j}^{(k,t)} = \sum_{i=1}^{n_j} \left(1 - u_{ji}^{(k)}\right) \left(\mathrm{e}^{\boldsymbol{x}_{ji}^\top \boldsymbol{\beta}^{(k,t)}} - y_{ji}\right) \boldsymbol{x}_{ji}$.
        Transmit $T_{1j}^{(k,t)}, \ldots, T_{4j}^{(k,t)}$ to central institution.
        For central institution, compute
            $\boldsymbol{\gamma}^{(k,t+1)} = \boldsymbol{\gamma}^{(k,t)} - \left(\sum_{j=1}^{J} T_{1j}^{(k,t)}\right)^{-1} \left(\sum_{j=1}^{J} T_{2j}^{(k,t)}\right)$,
            $\boldsymbol{\beta}^{(k,t+1)} = \boldsymbol{\beta}^{(k,t)} - \left(\sum_{j=1}^{J} T_{3j}^{(k,t)}\right)^{-1} \left(\sum_{j=1}^{J} T_{4j}^{(k,t)}\right)$.
        Transmit them to institution $j = 1, \ldots, J$ until $t = T$.
    Update $\boldsymbol{\gamma}^{(k+1)} = \boldsymbol{\gamma}^{(k,T)}$, $\boldsymbol{\beta}^{(k+1)} = \boldsymbol{\beta}^{(k,T)}$ and transmit them to institution $j = 1, \ldots, J$.
```

In the algorithm, δ is the parameter about the stopping criterion of the algorithm, which can be adjusted according to the accuracy requirement of the result.

To discuss the computation time complexity of the algorithm, we consider each EM algorithm iteration. For a local institution, the matrix multiplication operation is the main ingredient. The E step time complexity is $O(n_j(p + q))$, and the M step time complexity is $O(Tn_j(p^2 + q^2))$. For the central institution, it mainly completes the matrix inverse and multiplication operation, and the time complexity is $O(T(Jp^2 + Jq^2 + p^3 + q^3))$.

As for the communication cost of the algorithm, every local institution passing Hessian matrix in each internal iteration of M step is the main resource. And the communication cost of each EM algorithm iteration is $O(TJ(q^2 + p^2))$. The central institution gives the updated parameters to the local institutions, and the local institutions pass the calculated statistics based on the updated parameters to the centre instead of the original data

$(\boldsymbol{x}_{ji}, \boldsymbol{z}_{ji})$, so the algorithm has the property of privacy protection. However, when the dimension of the covariable is high (i.e., p and q are large), the communication cost of Algorithm 1 will also increase sharply. For distributed algorithms, communication cost is one of the important criteria to measure the algorithm, so the algorithm with higher communication efficiency is proposed below.

2.4. Communication-efficient distributed EM algorithm

Based on Equations (13) and (14), its equivalent expression can be obtained:

$$\boldsymbol{\gamma}^{(k,t+1)} = \boldsymbol{\gamma}^{(k,t)} - \left[\frac{1}{n}\sum_{j=1}^{J}\sum_{i=1}^{n_j}\frac{\mathrm{e}^{\boldsymbol{z}_{ji}^\top\boldsymbol{\gamma}^{(k,t)}}}{\left(1+\mathrm{e}^{\boldsymbol{z}_{ji}^\top\boldsymbol{\gamma}^{(k,t)}}\right)^2}\cdot\boldsymbol{z}_{ji}\boldsymbol{z}_{ji}^\top\right]^{-1} \times\left[\frac{1}{n}\sum_{j=1}^{J}\sum_{i=1}^{n_j}\left(\frac{\mathrm{e}^{\boldsymbol{z}_{ji}^\top\boldsymbol{\gamma}^{(k,t)}}}{1+\mathrm{e}^{\boldsymbol{z}_{ji}^\top\boldsymbol{\gamma}^{(k,t)}}}-u_{ji}^{(k)}\right)\boldsymbol{z}_{ji}\right], \tag{15}$$

$$\boldsymbol{\beta}^{(k,t+1)} = \boldsymbol{\beta}^{(k,t)} - \left[\frac{1}{n}\sum_{j=1}^{J}\sum_{i=1}^{n_j}\left(1-u_{ji}^{(k)}\right)\mathrm{e}^{\boldsymbol{x}_{ji}^\top\boldsymbol{\beta}^{(k,t)}}\cdot\boldsymbol{x}_{ji}\boldsymbol{x}_{ji}^\top\right]^{-1} \times\left[\frac{1}{n}\sum_{j=1}^{J}\sum_{i=1}^{n_j}\left(1-u_{ji}^{(k)}\right)\left(\mathrm{e}^{\boldsymbol{x}_{ji}^\top\boldsymbol{\beta}^{(k,t)}}-y_{ji}\right)\boldsymbol{x}_{ji}\right], \tag{16}$$

and the inverse term of Hessian matrix for all data in Equations (15) and (16) is replaced by an operation based on the data of 'Institution 1' (just choose one institution), i.e.,

$$\tilde{\boldsymbol{\gamma}}^{(k,t+1)} = \tilde{\boldsymbol{\gamma}}^{(k,t)} - \left[\frac{1}{n_1}\sum_{i=1}^{n_1}\frac{\mathrm{e}^{\boldsymbol{z}_{1i}^\top\tilde{\boldsymbol{\gamma}}^{(k,t)}}}{\left(1+\mathrm{e}^{\boldsymbol{z}_{1i}^\top\tilde{\boldsymbol{\gamma}}^{(k,t)}}\right)^2}\cdot\boldsymbol{z}_{1i}\boldsymbol{z}_{1i}^\top\right]^{-1} \times\left[\frac{1}{n}\sum_{j=1}^{J}\sum_{i=1}^{n_j}\left(\frac{\mathrm{e}^{\boldsymbol{z}_{ji}^\top\tilde{\boldsymbol{\gamma}}^{(k,t)}}}{1+\mathrm{e}^{\boldsymbol{z}_{ji}^\top\tilde{\boldsymbol{\gamma}}^{(k,t)}}}-u_{ji}^{(k)}\right)\boldsymbol{z}_{ji}\right], \tag{17}$$

$$\tilde{\boldsymbol{\beta}}^{(k,t+1)} = \tilde{\boldsymbol{\beta}}^{(k,t)} - \left[\frac{1}{n_1}\sum_{i=1}^{n_1}\left(1-u_{1i}^{(k)}\right)\mathrm{e}^{\boldsymbol{x}_{1i}^\top\tilde{\boldsymbol{\beta}}^{(k,t)}}\cdot\boldsymbol{x}_{1i}\boldsymbol{x}_{1i}^\top\right]^{-1} \times\left[\frac{1}{n}\sum_{j=1}^{J}\sum_{i=1}^{n_j}\left(1-u_{ji}^{(k)}\right)\left(\mathrm{e}^{\boldsymbol{x}_{ji}^\top\tilde{\boldsymbol{\beta}}^{(k,t)}}-y_{ji}\right)\boldsymbol{x}_{ji}\right]. \tag{18}$$

At this point, parameter update only needs one institution to pass its Hessian matrix at most. If the institution that calculates Hessian matrix can be selected as the central institution, then the whole algorithm does not need to pass Hessian matrix. After the iteration of Equations (17) and (18) reaches the requirements (for example, the iteration reaches T times), the final iteration result is recorded as the updated parameter $\tilde{\boldsymbol{\gamma}}^{(k,T)} := \tilde{\boldsymbol{\gamma}}^{(k+1)}$, $\tilde{\boldsymbol{\beta}}^{(k,T)} := \tilde{\boldsymbol{\beta}}^{(k+1)}$. We have Algorithm 2.

For the calculation time complexity of the algorithm, we consider each EM algorithm iteration. For a local institution, the matrix multiplication operation is mainly completed, the time complexity of E step is $O(n_j(p+q))$, and the time complexity of M step is $O(T(p^2+q^2+n_j(p+q)))$. For the central institution, it mainly completes the matrix inverse and multiplication operation, and the time complexity is $O(T(p^3+q^3+J(p+q)))$. The computational time complexity of a local institution or central institution is lower than Algorithm 1.

As for the communication, in each iteration of the algorithm, the communication cost is at most $O(T(q^2+p^2+Jq+Jp))$. If the institution participating in calculating Hessian matrix can be selected as the central institution, the communication cost of each iteration is reduced to $O(JT(q+p))$. It is lower than Algorithm 1. The central institution passes the updated parameters to the local institutions, and the local institutions transmit the calculated statistics based on the updated parameters to the central institution instead of the original data $(\boldsymbol{x}_{ji}, \boldsymbol{z}_{ji})$. Therefore, the algorithm has the property of privacy protection.

Algorithm 2 Communication-Efficient Distributed EM Algorithm

```
**Initialize**: Let $U_{ji}^{(0)} = I(y_{ji} = 0)$, and every institution calculates Equations (11) and (12) by its own data. Take average to get $\tilde{\boldsymbol{\gamma}}^{(0)}$ and $\tilde{\boldsymbol{\beta}}^{(0)}$ on central institution and transmit the results to institution $j = 1, \ldots, J$.
While $\|\tilde{\boldsymbol{\gamma}}^{(k+1)} - \tilde{\boldsymbol{\gamma}}^{(k)}\|_2 \geq \delta$ or $\|\tilde{\boldsymbol{\beta}}^{(k+1)} - \tilde{\boldsymbol{\beta}}^{(k)}\|_2 \geq \delta$, $k$ is the $k$th iteration.
    **E step**: For institution $j = 1, \ldots, J$, compute $U_{ji}^{(k)}$ by Equation (10), $i = 1, \ldots, n_j$.
    **M step**: Internal Newton iteration ($\tilde{\boldsymbol{\gamma}}^{(k,0)} = \tilde{\boldsymbol{\gamma}}^{(k-1)}$,$\tilde{\boldsymbol{\beta}}^{(k,0)} = \tilde{\boldsymbol{\beta}}^{(k-1)}$):
        For **institution** $j = 1, \ldots, J$, $t = 1, \ldots, T$ compute
            $T_{11}^{(k,t)} = \frac{1}{n_1}\sum_{i=1}^{n_1} \frac{e^{\boldsymbol{z}_{1i}^\top \tilde{\boldsymbol{\gamma}}^{(k,t)}}}{\left(1+e^{\boldsymbol{z}_{1i}^\top \tilde{\boldsymbol{\gamma}}^{(k,t)}}\right)^2} \cdot \boldsymbol{z}_{1i}\boldsymbol{z}_{1i}^\top$,
            $T_{2j}^{(k,t)} = \sum_{i=1}^{n_j} \left( \frac{e^{\boldsymbol{z}_{ji}^\top \tilde{\boldsymbol{\gamma}}^{(k,t)}}}{1+e^{\boldsymbol{z}_{ji}^\top \tilde{\boldsymbol{\gamma}}^{(k,t)}}} - u_{ji}^{(k)} \right) \boldsymbol{z}_{ji}$,
            $T_{31}^{(k,t)} = \frac{1}{n_1}\sum_{i=1}^{n_1} \left(1 - u_{1i}^{(k)}\right) e^{\boldsymbol{x}_{1i}^\top \tilde{\boldsymbol{\beta}}^{(k,t)}} \cdot \boldsymbol{x}_{1i}\boldsymbol{x}_{1i}^\top$,
            $T_{4j}^{(k,t)} = \sum_{i=1}^{n_j} \left(1 - u_{ji}^{(k)}\right) \left( e^{\boldsymbol{x}_{ji}^\top \tilde{\boldsymbol{\beta}}^{(k,t)}} - y_{ji} \right) \boldsymbol{x}_{ji}$.
        Transmit $T_{11}^{(k,t)}, \ldots, T_{4j}^{(k,t)}$ to central institution.
        For **central institution**, compute
            $\tilde{\boldsymbol{\gamma}}^{(k,t+1)} = \tilde{\boldsymbol{\gamma}}^{(k,t)} - \left(T_{11}^{(k,t)}\right)^{-1} \left(\frac{1}{n}\sum_{j=1}^{J} T_{2j}^{(k,t)}\right)$,
            $\tilde{\boldsymbol{\beta}}^{(k,t+1)} = \tilde{\boldsymbol{\beta}}^{(k,t)} - \left(T_{31}^{(k,t)}\right)^{-1} \left(\frac{1}{n}\sum_{j=1}^{J} T_{4j}^{(k,t)}\right)$.
        Transmit them to institution $j = 1, \ldots, J$ until $t = T$.
    Update $\tilde{\boldsymbol{\gamma}}^{(k+1)} = \tilde{\boldsymbol{\gamma}}^{(k,T)}$, $\tilde{\boldsymbol{\beta}}^{(k+1)} = \tilde{\boldsymbol{\beta}}^{(k,T)}$ and transmit them to institution $j = 1, \ldots, J$.
```

Motivated by Jordan et al. (2018), we can explain Equations (17) and (18) in terms of likelihood. For convenience, rewrite Equations (8) and (9) into the following distributed form, and add the factor $1/n$ (this operation is equivalent from the perspective of optimal solution, and the reason for adding this item will be explained later):

$$\begin{aligned} \frac{1}{n} L_{c,1}(\boldsymbol{\gamma}) &= \frac{1}{n}\sum_{j=1}^{J}\sum_{i=1}^{n_j}\left\{ u_{ji}\boldsymbol{z}_{ji}^\top\boldsymbol{\gamma} - \log\left(1 + e^{\boldsymbol{z}_{ji}^\top\boldsymbol{\gamma}}\right)\right\} \\ &:= \frac{1}{n}\sum_{j=1}^{J}\sum_{i=1}^{n_j} L_{c,1,ji}(\boldsymbol{\gamma}) := \frac{1}{n}\sum_{j=1}^{J} L_{c,1,j}(\boldsymbol{\gamma}), \end{aligned} \tag{19}$$

$$\begin{aligned} \frac{1}{n} L_{c,2}(\boldsymbol{\beta}) &= \frac{1}{n}\sum_{j=1}^{J}\sum_{i=1}^{n_j}\left(1 - u_{ji}\right)\left(y_{ji}\boldsymbol{x}_{ji}^\top\boldsymbol{\beta} - e^{\boldsymbol{x}_{ji}^\top\boldsymbol{\beta}}\right) \\ &:= \frac{1}{n}\sum_{j=1}^{J}\sum_{i=1}^{n_j} L_{c,2,ji}(\boldsymbol{\beta}) := \frac{1}{n}\sum_{j=1}^{J} L_{c,2,j}(\boldsymbol{\beta}). \end{aligned} \tag{20}$$

Based on E step, take Taylor's expansion in Equations (19) and (20) at $\tilde{\boldsymbol{\gamma}}^{(k)}$ or $\tilde{\boldsymbol{\beta}}^{(k)}$. The following discussion is for $L_{c,1}(\boldsymbol{\gamma})$ as an example, and $L_{c,2}(\boldsymbol{\beta})$ is similar.

$$\begin{aligned} \frac{1}{n} L_{c,1}(\boldsymbol{\gamma}) = \frac{1}{n} L_{c,1}\left(\tilde{\boldsymbol{\gamma}}^{(k)}\right) + \frac{1}{n}\left\langle \nabla L_{c,1}\left(\tilde{\boldsymbol{\gamma}}^{(k)}\right), \boldsymbol{\gamma} - \tilde{\boldsymbol{\gamma}}^{(k)}\right\rangle \\ + \frac{1}{n}\sum_{m=2}^{\infty}\frac{1}{m!}\nabla^m L_{c,1}\left(\tilde{\boldsymbol{\gamma}}^{(k)}\right)\left(\boldsymbol{\gamma} - \tilde{\boldsymbol{\gamma}}^{(k)}\right)^{\otimes m}. \end{aligned} \tag{21}$$

Replace the high-order derivative part (second order and above) of all the data with the data from a local institution (if the factor $1/n$ is not added in Equations (19) and (20), the approximate substitution here will become unreasonable).

$$\frac{1}{n} L_{c,1}(\boldsymbol{\gamma}) \approx \frac{1}{n} L_{c,1}\left(\tilde{\boldsymbol{\gamma}}^{(k)}\right) + \frac{1}{n}\left\langle \nabla L_{c,1}\left(\tilde{\boldsymbol{\gamma}}^{(k)}\right), \boldsymbol{\gamma} - \tilde{\boldsymbol{\gamma}}^{(k)}\right\rangle$$

$$+ \frac{1}{n_1} \sum_{m=2}^{\infty} \frac{1}{m!} \nabla^m L_{c,1,1}\left(\tilde{\boldsymbol{\gamma}}^{(k)}\right)\left(\boldsymbol{\gamma} - \tilde{\boldsymbol{\gamma}}^{(k)}\right)^{\otimes m}. \tag{22}$$

Take Taylor's expansion in $\frac{1}{n_1} L_{c,1,1}(\boldsymbol{\gamma})$ at $\tilde{\boldsymbol{\gamma}}^{(k)}$:

$$\begin{aligned} \frac{1}{n_1} L_{c,1,1}(\boldsymbol{\gamma}) = {} & \frac{1}{n_1} L_{c,1,1}\left(\tilde{\boldsymbol{\gamma}}^{(k)}\right) + \frac{1}{n_1}\left\langle \nabla L_{c,1,1}\left(\tilde{\boldsymbol{\gamma}}^{(k)}\right), \boldsymbol{\gamma} - \tilde{\boldsymbol{\gamma}}^{(k)}\right\rangle \\ & + \frac{1}{n_1} \sum_{m=2}^{\infty} \frac{1}{m!} \nabla^m L_{c,1,1}\left(\tilde{\boldsymbol{\gamma}}^{(k)}\right)\left(\boldsymbol{\gamma} - \tilde{\boldsymbol{\gamma}}^{(k)}\right)^{\otimes m}. \end{aligned} \tag{23}$$

Substitute Equation (23) into Equation (22) and then ignore the constant term to get

$$\frac{1}{n} L_{c,1}(\tilde{\boldsymbol{\gamma}}) := \frac{1}{n_1} L_{c,1,1}(\boldsymbol{\gamma}) - \left\langle \boldsymbol{\gamma}, \frac{1}{n_1} \nabla L_{c,1,1}\left(\tilde{\boldsymbol{\gamma}}^{(k)}\right) - \frac{1}{n} \nabla L_{c,1}\left(\tilde{\boldsymbol{\gamma}}^{(k)}\right)\right\rangle. \tag{24}$$

Equation (24) is solved by Newton's algorithm, and t is the index of Newton iteration. The iteration expression is

$$\tilde{\boldsymbol{\gamma}}^{(k,t+1)} = \tilde{\boldsymbol{\gamma}}^{(k,t)} - \frac{n_1}{n} \nabla^2 L_{c,1,1}\left(\tilde{\boldsymbol{\gamma}}^{(k,t)}\right)^{-1} \nabla L_{c,1}\left(\tilde{\boldsymbol{\gamma}}^{(k,t)}\right). \tag{25}$$

When substituted into the specific expression, this expression is the same as Equations (17) and (18).

3. Main theoretic results

This section introduces the theoretical results of Algorithm 2. We focus on low-dimensional situation. In order to fully reduce the communication cost, $T = 1$ is taken. This part of the theory is obtained under this situation. The conclusions include the asymptotic normality of parameter estimator obtained by Algorithm 2 and the consistent estimator of asymptotic variance.

Assumptions:

(1) Parameter space Ω: let $\boldsymbol{\theta} = [\boldsymbol{\gamma}^\top, \boldsymbol{\beta}^\top]^\top, p + q = r, \boldsymbol{\theta} \in \Omega$, and Ω is a compact and convex subset of $\mathbb{R}^r$.

(2) $\Omega_{\tilde{\boldsymbol{\theta}}^{(0)}} := \left\{\boldsymbol{\theta} \in \Omega : L(\boldsymbol{\theta}) \geq L\left(\tilde{\boldsymbol{\theta}}^{(0)}\right)\right\}$ is compact for any $\boldsymbol{\theta}$ satisfying $L\left(\tilde{\boldsymbol{\theta}}^{(0)}\right) > -\infty$ and $\Omega_{\tilde{\boldsymbol{\theta}}^{(0)}}$ is in the interior of Ω, where $\tilde{\boldsymbol{\theta}}^{(0)} \in \Omega$ is the initial value of Algorithm 2.

(3) $F_n = \frac{1}{n} \sum_{j=1}^{J} \sum_{i=1}^{n_j} F(\boldsymbol{\theta}; w_{ji})$, where

$$\begin{aligned} F(\boldsymbol{\theta}; w_{ji}) = {} & u I_{\{y_{ji}=0\}} \boldsymbol{z}_{ji}^\top \boldsymbol{\gamma} - \log\left(1 + \mathrm{e}^{\boldsymbol{z}_{ji}^\top \boldsymbol{\gamma}}\right) + \left(y_{ji} \boldsymbol{x}_{ji}^\top \boldsymbol{\beta} - \mathrm{e}^{\boldsymbol{x}_{ji}^\top \boldsymbol{\beta}}\right) I_{\{y_{ji}>0\}} \\ & - (1-u) \mathrm{e}^{\boldsymbol{x}_{ji}^\top \boldsymbol{\beta}} I_{\{y_{ji}=0\}} - \log(y_{ji}!) I_{\{y_{ji}>0\}} - (1-u) I_{\{y_{ji}=0\}}, w_{ji} = \boldsymbol{x}_{ji}, \boldsymbol{z}_{ji}, y_{ji}, \end{aligned}$$

and $\boldsymbol{\theta}^{*'} := \arg\max_{\boldsymbol{\theta} \in \Omega} E_y[F(\boldsymbol{\theta}; w)], \forall u \in [0,1]$, which satisfy the following conditions.

(a) $I(\boldsymbol{\theta}^{*'}) := -\nabla^2 E[F(\boldsymbol{\theta}^{*'}, w)]$, and $\mu_- I_r \preccurlyeq I(\boldsymbol{\theta}^{*'}) \preccurlyeq \mu_+ I_r$.

(b) $\forall \delta > 0, \exists \epsilon > 0, \liminf_{n \to \infty} \mathrm{P}\left\{\inf_{\|\boldsymbol{\theta} - \boldsymbol{\theta}^{*'}\|_2 \geq \delta} (F(\boldsymbol{\theta}^{*'}) - F(\boldsymbol{\theta})) \geq \epsilon\right\} = 1$.

(c) Let $U(\rho) = \{\boldsymbol{\theta} \in \mathbb{R}^r | \parallel \boldsymbol{\theta} - \boldsymbol{\theta}^{*'} \parallel_2 \leq \rho\} \subset \Omega$, and there exist constants (G, L) and function $N(w)$ such that $E[\|\nabla F(\boldsymbol{\theta}; W)\|_2^{16}] \leq G^{16}, E[\|\nabla^2 F(\boldsymbol{\theta}; W) - I(\boldsymbol{\theta})\|_2^{16}] \leq L^{16}, \forall \boldsymbol{\theta} \in U(\rho)$; $\|\nabla^2 F(\boldsymbol{\theta}; w) - \nabla^2 F(\boldsymbol{\theta}'; w)\|_2 \leq N(w) \times \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_2, \forall \boldsymbol{\theta}, \boldsymbol{\theta}' \in U(\rho)$. Moreover, $N(w)$ satisfies $E[N^{16}(W)] \leq N^{16}$ for some constant $N > 0$.

(4) $\tilde{\boldsymbol{\theta}}^{(0)} = [\tilde{\boldsymbol{\gamma}}^{(0)^\top}, \tilde{\boldsymbol{\beta}}^{(0)^\top}]^\top$ satisfies $\parallel \tilde{\boldsymbol{\theta}}^{(0)} - \boldsymbol{\theta}^{*(1)} \parallel_2 \leq \min\left\{\rho, \frac{(1-\rho)\mu_-}{16N}\right\}$, and $\boldsymbol{\theta}^{*(1)}$ is $\boldsymbol{\theta}^{*'}$ when $u = u_{ji}^{(0)}$.

The above assumptions are related to the Jordan approximation of the M step of the EM algorithm and the convergence of the EM algorithm. Specifically, Assumption (1) is the condition of the parameter space. Assumption 2 is related to the boundedness of the EM sequences. Assumption 3 is about the objective function of M step. (3)(a)–(3)(c) are used in the proof of Proposition A.4, A.5 in the appendix. Specifically, (3)(a) is related to the local convexity, which is used to deduce the upper bound error of M step objective function after the approximation. (3)(b) is global identifiability condition, which is a basic condition to ensure the consistency of estimators. (3)(c) is related to the smoothness of the objective function of M step, and is used to deduce the error reduction order of the estimator after one iteration of M step. Assumption (4) is a requirement for initial values of the algorithm. Then, we have the following two theorems.

Theorem 3.1: *Denote the final result of Algorithm 2 as $\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\gamma}}$, and the result of Equation (4) maximization is marked as $\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}$, which is the MLE of the original problem. Under the above conditions, if $n \to \infty$, we have*

$$\tilde{\boldsymbol{\beta}} \xrightarrow{d} \hat{\boldsymbol{\beta}}, \quad \tilde{\boldsymbol{\gamma}} \xrightarrow{d} \hat{\boldsymbol{\gamma}},$$

and

$$\sqrt{n}\left(\tilde{\boldsymbol{\gamma}} - \boldsymbol{\gamma}^*\right) \xrightarrow{d} N(0, \Sigma_{11}), \sqrt{n}\left(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\right) \xrightarrow{d} N(0, \Sigma_{22}),$$

where Σ_{11}, Σ_{22} are the upper left $p \times p$ matrix and lower right $q \times q$ matrix of Σ, respectively. $\Sigma := I(\boldsymbol{\theta}^*)^{-1} = E(\nabla^2 l(\boldsymbol{\theta}^*, y, \boldsymbol{z}, \boldsymbol{x}))^{-1}$, where $\boldsymbol{\theta}^* = [\boldsymbol{\beta}^{*\top}, \boldsymbol{\gamma}^{*\top}]^\top$ is the true parameter, which is the maximizer of the expection of Equation (4).

Theorem 3.2: $\widehat{\Sigma} := \left(-\frac{1}{n_1}\sum_{i=1}^{n_1} \nabla^2 l_1\right)^{-1} \left(\frac{1}{n_1}\sum_{i=1}^{n_1} \nabla l_1 \nabla l_1^\top\right) \left(-\frac{1}{n_1}\sum_{i=1}^{n_1} \nabla^2 l_1\right)^{-1}$*, where $l_1 := l(\hat{\boldsymbol{\theta}}; y_{1i}, \boldsymbol{z}_{1i}, \boldsymbol{x}_{1i}), n_1 = O(a)$. We have*

$$\widehat{\Sigma} \xrightarrow{p} \Sigma, \quad \text{when } a \to \infty.$$

4. One shot communication-efficient distributed EM algorithm

In order to reduce the communication cost of distributed algorithm, in addition to reducing the communication cost of each transmission (Algorithm 2), the same purpose can be achieved by reducing the transmission times in the M step. In order to make a full comparison with Algorithm 2, this section proposes another algorithm, in which only one transmission is carried out between different institutions. The main idea is motivated by Zhu et al. (2021).

The distributed data structure is consistent with that described in Section 2.3, and the difference with the previous algorithm is mainly in the M step of the EM algorithm. Specifically, the illustration starts with the M step on $\boldsymbol{\gamma}$. According to the result of Equations (19) and (20), there is $-L_{c,1}(\boldsymbol{\gamma}) = -\sum_{j=1}^{J} L_{c,1,j}(\boldsymbol{\gamma}) = -\sum_{j=1}^{J}\sum_{i=1}^{n_j} L_{c,1,ji}(\boldsymbol{\gamma})$. Take Taylor's expansion in $\sum_{i=1}^{n_j} L_{c,1,ji}(\boldsymbol{\gamma})$ at $\hat{\boldsymbol{\gamma}}_j$, where $\hat{\boldsymbol{\gamma}}_j = \underset{\boldsymbol{\gamma}}{\operatorname{argmin}}(-L_{c,1,j}(\boldsymbol{\gamma}))$ is the optimization result for one local institution. Ignore the higher order terms of Taylor expansion (so this algorithm requires a sufficient sample size of every local institution) and the constant term independent of $\boldsymbol{\gamma}$, we can get (with $\nabla \sum_{i=1}^{n_j} L_{c,1,ji}(\hat{\boldsymbol{\gamma}}_j) = 0$)

$$\begin{aligned} -L_{c,1}(\boldsymbol{\gamma}) &\approx -\sum_{j=1}^{J}\sum_{i=1}^{n_j}(\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}_j)^\top \nabla^2 L_{c,1,ji}(\hat{\boldsymbol{\gamma}}_j)(\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}_j) \\ &= \sum_{j=1}^{J}(\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}_j)^\top \left[\sum_{i=1}^{n_j} \frac{\mathrm{e}^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\gamma}}_j}}{\left(1 + \mathrm{e}^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\gamma}}_j}\right)^2} \cdot \boldsymbol{z}_{ji}\boldsymbol{z}_{ji}^\top\right](\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}_j) \\ &:= -\overline{L_{c,1}}(\boldsymbol{\gamma}). \end{aligned} \tag{26}$$

The above equation is regarded as the optimization objective, and according to the weighted least squares algorithm, it can be obtained

$$\begin{aligned} \hat{\boldsymbol{\gamma}}_{\text{os}} = \underset{\boldsymbol{\gamma}}{\operatorname{argmin}}(-\overline{L_{c,1}}(\boldsymbol{\gamma})) &= \left[\sum_{j=1}^{J}\sum_{i=1}^{n_j} \nabla^2(-L_{c,1,ji}(\hat{\boldsymbol{\gamma}}_j))\right]^{-1} \\ &\times \left[\sum_{j=1}^{J}\sum_{i=1}^{n_j} \nabla^2(-L_{c,1,ji}(\hat{\boldsymbol{\gamma}}_j))\hat{\boldsymbol{\gamma}}_j\right]. \end{aligned} \tag{27}$$

Similar to the derivation of $\boldsymbol{\gamma}$, the M step for $\boldsymbol{\beta}$ is similar, resulting in

$$-\overline{L_{c,2}}(\boldsymbol{\beta}) := \sum_{j=1}^{J}(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}_j)^\top \left[\sum_{i=1}^{n_j}\left(1 - U_{ji}\right)\mathrm{e}^{\boldsymbol{x}_{ji}^\top \hat{\boldsymbol{\beta}}_j} \cdot \boldsymbol{x}_{ji}\boldsymbol{x}_{ji}^\top\right](\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}_j)$$

$$= -\sum_{j=1}^{J}\sum_{i=1}^{n_j}(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}_j)^\top \nabla^2 L_{c,2,ji}(\hat{\boldsymbol{\gamma}}_j)(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}_j), \tag{28}$$

$$\hat{\boldsymbol{\beta}}_{\text{os}} = \underset{\boldsymbol{\beta}}{\text{argmin}}(-\overline{L_{c,2}}(\boldsymbol{\beta})) = \left[\sum_{j=1}^{J}\sum_{i=1}^{n_j}\nabla^2(-L_{c,2,ji}(\hat{\boldsymbol{\beta}}_j))\right]^{-1} \times \left[\sum_{j=1}^{J}\sum_{i=1}^{n_j}\nabla^2(-L_{c,2,ji}(\hat{\boldsymbol{\beta}}_j))\hat{\boldsymbol{\beta}}_j\right], \tag{29}$$

where $\hat{\boldsymbol{\beta}}_j = \underset{\boldsymbol{\beta}}{\text{argmin}}(-L_{c,2,j}(\boldsymbol{\beta}))$is the optimization results of one local institution. We have Algorithm 3.

Algorithm 3 One Shot Communication-Efficient Distributed EM Algorithm

```
Initialize: Let $U_{ji}^{(0)} = I(y_{ji} = 0)$, and every institution calculates Equations (11) and (12) by its own data. Take average to get $\hat{\boldsymbol{\gamma}}_{\text{os}}^{(0)}, \hat{\boldsymbol{\beta}}_{\text{os}}^{(0)}$ on central institution and transmit the results to institution $j = 1, \ldots, J$.
While $\|\hat{\boldsymbol{\gamma}}_{\text{os}}^{(k+1)} - \hat{\boldsymbol{\gamma}}_{\text{os}}^{(k)}\|_2 \geq \delta$ or $\|\hat{\boldsymbol{\beta}}_{\text{os}}^{(k+1)} - \hat{\boldsymbol{\beta}}_{\text{os}}^{(k)}\|_2 \geq \delta$, $k$ is the $k$th iteration.
    E step: For institution $j = 1, \ldots, J$, compute $U_{ji}^{(k)}$ by Equation (10), $i = 1, \ldots, n_j$.
    M step:
        For institution $j = 1, \ldots, J$, compute
        $\hat{\boldsymbol{\gamma}}_j^{(k)} = \underset{\boldsymbol{\gamma}}{\text{argmin}}(-L_{c,1,j}(\boldsymbol{\gamma}; U_{ji}^{(k)}))$,
        $\hat{\boldsymbol{\beta}}_j^{(k)} = \underset{\boldsymbol{\beta}}{\text{argmin}}(-L_{c,2,j}(\boldsymbol{\beta}; U_{ji}^{(k)}))$ , $i = 1, \ldots, n_j$,
        $T_{1j}^{(k)} = \sum_{i=1}^{n_j} \frac{e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\gamma}}_j^{(k)}}}{\left(1+e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\gamma}}_j^{(k)}}\right)^2} \cdot \boldsymbol{z}_{ji}\boldsymbol{z}_{ji}^\top$,   $T_{2j}^{(k)} = \sum_{i=1}^{n_j} \frac{e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\gamma}}_j^{(k)}}}{\left(1+e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\gamma}}_j^{(k)}}\right)^2} \cdot \boldsymbol{z}_{ji}\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\gamma}}_j^{(k)}$,
        $T_{3j}^{(k)} = \sum_{i=1}^{n_j} \frac{e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\beta}}_j^{(k)}}}{\left(1+e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\beta}}_j^{(k)}}\right)^2} \cdot \boldsymbol{z}_{ji}\boldsymbol{z}_{ji}^\top$,   $T_{4j}^{(k)} = \sum_{i=1}^{n_j} \frac{e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\beta}}_j^{(k)}}}{\left(1+e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\beta}}_j^{(k)}}\right)^2} \cdot \boldsymbol{z}_{ji}\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\beta}}_j^{(k)}$.
        Transmit $T_{1j}^{(k)}, \ldots, T_{4j}^{(k)}$ to central institution.
        Central institution update $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ by Equations (26) and (27):
        $\hat{\boldsymbol{\gamma}}_{\text{os}}^{(k+1)} = \left[\sum_{j=1}^{J} T_{1j}^{(k)}\right]^{-1}\left[\sum_{j=1}^{J} T_{2j}^{(k)}\right]$,
        $\hat{\boldsymbol{\beta}}_{\text{os}}^{(k+1)} = \left[\sum_{j=1}^{J} T_{3j}^{(k)}\right]^{-1}\left[\sum_{j=1}^{J} T_{4j}^{(k)}\right]$,
        and then transmit them to institution $j = 1, \ldots, J$.
```

If we solve the $\hat{\boldsymbol{\gamma}}_j^{(k)}, \hat{\boldsymbol{\beta}}_j^{(k)}$ using Newton's method, C is the number of the iterations. For the time complexity of the algorithm, we consider each EM algorithm iteration. For a local institution, matrix multiplication and inverse operation are mainly completed. The E step time complexity is $O(N_j(p + q))$, and the M step time complexity is $O(C(N_j(p^2 + q^2) + p^3 + q^3))$. The central institution mainly completes matrix inversion operation, and the computation time complexity is $O(Jp^2 + Jq^2 + q^3 + q^3)$.

In order to reduce the computing cost of a single local institution, the optimization problem of $\hat{\boldsymbol{\gamma}}_j^{(k)}, \hat{\boldsymbol{\beta}}_j^{(k)}$ obtained in Algorithm 3 can be replaced by a one-step approximation:

$$\hat{\boldsymbol{\gamma}}_j^{(k)} = \hat{\boldsymbol{\gamma}}_{\text{os}}^{(k)} - \left[\sum_{i=1}^{n_j} \frac{e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\gamma}}_{\text{os}}^{(k)}}}{\left(1 + e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\gamma}}_{\text{os}}^{(k)}}\right)^2} \cdot \boldsymbol{z}_{ji}\boldsymbol{z}_{ji}^\top\right]^{-1} \times \left[\sum_{i=1}^{n_j}\left(\frac{e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\gamma}}_{\text{os}}^{(k)}}}{1 + e^{\boldsymbol{z}_{ji}^\top \hat{\boldsymbol{\gamma}}_{\text{os}}^{(k)}}} - u_{ji}^{(k)}\right)\boldsymbol{z}_{ji}\right], \tag{30}$$

$$\hat{\boldsymbol{\beta}}_j^{(k)} = \hat{\boldsymbol{\beta}}_{\mathrm{os}}^{(k)} - \left[\sum_{i=1}^{n_j} \left(1 - u_{ji}^{(k)}\right) \mathrm{e}^{\boldsymbol{x}_{ji}^{\top} \hat{\boldsymbol{\beta}}_{\mathrm{os}}^{(k)}} \cdot \boldsymbol{x}_{ji} \boldsymbol{x}_{ji}^{\top} \right]^{-1}$$
$$\times \left[\sum_{i=1}^{n_j} \left(1 - u_{ji}^{(k)}\right) \left(\mathrm{e}^{\boldsymbol{x}_{ji}^{\top} \hat{\boldsymbol{\beta}}_{\mathrm{os}}^{(k)}} - y_{ji} \right) \boldsymbol{x}_{ji} \right]. \tag{31}$$

After the above approximation, the effect on the computational time complexity is reflected in the M step of a single local institution, which is now $O(n_j(p^2 + q^2) + p^3 + q^3))$.

Table 1. Computing complexity and communication cost of each EM iteration for Algorithms 1–3.

	Algorithm 1	Algorithm 2	Algorithm 3
Single	$O(Tn_j(p^2 + q^2))$	$O(T[p^2 + q^2 + n_j(p + q)])$	$O(C(n_j(p^2 + q^2) + p^3 + q^3))$
Central	$O(T[J(p^2 + q^2) + q^3 + p^3])$	$O(T[J(p + q) + p^3 + q^3])$	$O(J(p^2 + q^2) + p^3 + q^3)$
Communication	$O(TJ(q^2 + p^2))$	$O(T(q^2 + p^2 + J(q + p)))$	$O(J(p^2 + q^2))$

Note: 'Single' stands for computing complexity of a single local institution; 'central' stands for the calculation time complexity of central institution; 'communication' stands for communication cost.

Table 2. Computing complexity and communication cost of each EM iteration for Algorithms 1–3 with $T = C = 1$.

	Algorithm 1	Algorithm 2	Algorithm 3
Single	$O(n_j(p^2 + q^2))$	$O(p^2 + q^2 + n_j(p + q))$	$O(n_j(p^2 + q^2) + p^3 + q^3)$
Central	$O(J(p^2 + q^2) + q^3 + p^3)$	$O(J(p + q) + p^3 + q^3)$	$O(J(p^2 + q^2) + p^3 + q^3)$
Communication	$O(J(q^2 + p^2))$	$O(q^2 + p^2 + J(q + p))$	$O(J(p^2 + q^2))$

Note: 'Single' stands for computing complexity of a single local institution; 'central' stands for the calculation time complexity of central institution; 'communication' stands for communication cost.

The communication cost of the EM algorithm for one iteration is $O(J(p^2 + q^2))$. The central institution passes the updated parameters to the single machine, and the local institution transmits calculated statistics based on updated parameters to the centre instead of the original data $(\boldsymbol{x}_{ji}, \boldsymbol{z}_{ji})$. Therefore, this algorithm has the property of privacy protection. Compared with the previous two algorithms, the iteration of the optimization algorithm within the M step (that is, solving $\hat{\boldsymbol{\gamma}}^{(k)}, \hat{\boldsymbol{\beta}}^{(k)}$) is completed within one local institution, and no cross-institution communication occurs. The EM algorithm only carries out one transmission at one iteration. It is not related to the number of iterations of internal optimization of M step. The iteration of M step of Algorithms 1 and 2 introduced above to solve the optimization problem is completed across different institutions and requires cross-institution communication, so the communication cost increases with times of iterations of M step. Therefore, the number of communication times in each iteration of the EM algorithm is related to the number of internal iterations of M step.

Table 1 compares the computing complexity and communication cost of each EM iteration for Algorithms 1–3. Compared with Algorithm 1, the computing complexity and communication cost of Algorithm 2 are reduced. For Algorithm 3, the computing complexity of single local institution is related to C, and the complexity and communication cost of central institution decrease. The computing complexity and communication cost of Algorithms 2 and 3 are related to T and C.

To fully reduce communication or computing costs, set $T = C = 1$, as shown in Table 2. The computing complexity of Algorithm 2 is lower than that of Algorithm 1. For Algorithm 3, the time complexity of single local institution is higher than that of Algorithm 1 because a single local institution has to deal with the optimization problem, and their computing complexities of central institution are the same. In terms of communication cost, the communication cost of Algorithm 2 is the lowest. Since C is only related to the computing cost of a single local institution, the communication cost of Algorithm 3 is not improved.

5. Simulation study

We present the performance of the proposed algorithms, under a variety of settings and for varying sample sizes. All simulation results are calculated based on 500 replications. Section 5.1 studies the performances under the setting of homogeneous data which means that all the observations of different institutions are from the same distribution. Section 5.2 studies the heterogeneous data, which means different institutions have different data distributions.

5.1. Homogeneous data

In this section, data $\{\boldsymbol{x}_{ji}, \boldsymbol{z}_{ji}\}$ of different institutions come from the same distribution.

Table 3. Empirical performances of proposed Algorithms under different J with fixed total sample size.

			J (Number of institutions)				
			50	100	500	1000	5000
Algorithm 1	$T = 1$	$\boldsymbol{\beta}$ bias	1.578	0.498	0.343	0.354	0.210
		$\boldsymbol{\beta}$ var	1.188	0.590	0.124	0.063	0.012
		$\boldsymbol{\beta}$ mse	1.190	0.590	0.124	0.064	0.012
		$\boldsymbol{\gamma}$ bias	6.652	2.144	0.408	1.068	0.731
		$\boldsymbol{\gamma}$ var	13.258	6.638	1.306	0.622	0.141
		$\boldsymbol{\gamma}$ mse	13.302	6.642	1.306	0.663	0.141
		Times of no convergence	0	0	0	0	0
	$T = 3$	$\boldsymbol{\beta}$ bias	1.578	0.498	0.344	0.354	0.210
		$\boldsymbol{\beta}$ var	1.188	0.590	0.124	0.063	0.012
		$\boldsymbol{\beta}$ mse	1.190	0.590	0.124	0.064	0.012
		$\boldsymbol{\gamma}$ bias	6.651	2.143	0.408	1.070	0.730
		$\boldsymbol{\gamma}$ var	13.258	6.637	1.306	0.622	0.141
		$\boldsymbol{\gamma}$ mse	13.302	6.642	1.306	0.633	0.141
		Times of no convergence	0	0	0	0	0
Algorithm 2	$T = 1$	$\boldsymbol{\beta}$ bias	1.924	1.591	0.351	0.109	0.031
		$\boldsymbol{\beta}$ var	1.184	0.570	0.118	0.058	0.013
		$\boldsymbol{\beta}$ mse	1.187	0.573	0.118	0.058	0.013
		$\boldsymbol{\gamma}$ bias	7.337	1.723	1.288	1.497	0.345
		$\boldsymbol{\gamma}$ var	13.091	6.705	1.306	0.649	0.138
		$\boldsymbol{\gamma}$ mse	13.145	6.708	1.308	0.652	0.138
		Times of no convergence	195	197	188	188	196
	$T = 3$	$\boldsymbol{\beta}$ bias	1.565	1.577	0.427	0.108	0.041
		$\boldsymbol{\beta}$ var	1.187	0.579	0.119	0.058	0.013
		$\boldsymbol{\beta}$ mse	1.189	0.581	0.119	0.58	0.013
		$\boldsymbol{\gamma}$ bias	8.368	2.291	1.456	1.549	0.472
		$\boldsymbol{\gamma}$ var	13.135	6.687	1.319	0.651	0.138
		$\boldsymbol{\gamma}$ mse	13.205	6.692	1.320	0.653	0.139
		Times of no convergence	200	204	197	195	207
Algorithm 3	$C = 1$	$\boldsymbol{\beta}$ bias	51.459	51.043	51.112	51.328	51.173
		$\boldsymbol{\beta}$ var	1.091	0.555	0.120	0.060	0.011
		$\boldsymbol{\beta}$ mse	3.739	3.160	2.732	2.695	2.630
		$\boldsymbol{\gamma}$ bias	162.060	167.204	169.052	170.462	170.172
		$\boldsymbol{\gamma}$ var	9.826	4.813	0.980	0.492	0.106
		$\boldsymbol{\gamma}$ mse	36.090	32.770	29.559	29.549	29.065
		Times of no convergence	0	0	0	0	0
	$C = 3$	$\boldsymbol{\beta}$ bias	28.039	27.439	27.412	27.563	27.430
		$\boldsymbol{\beta}$ var	1.114	0.559	0.120	0.060	0.012
		$\boldsymbol{\beta}$ mse	1.900	1.312	0.871	0.820	0.764
		$\boldsymbol{\gamma}$ bias	87.265	92.627	93.822	95.150	94.858
		$\boldsymbol{\gamma}$ var	11.150	5.519	1.109	0.557	0.120
		$\boldsymbol{\gamma}$ mse	18.765	14.031	9.912	9.612	9.117
		Times of no convergence	0	0	0	0	0
	Sub	$\boldsymbol{\beta}$ bias	24.305	24.305	24.305	24.305	24.305
		$\boldsymbol{\beta}$ var	97.445	97.445	97.445	97.445	97.445
		$\boldsymbol{\beta}$ mse	98.036	98.036	98.036	98.036	98.036
		$\boldsymbol{\gamma}$ bias	535.914	535.914	535.914	535.914	535.914
		$\boldsymbol{\gamma}$ var	8989.774	8989.774	8989.774	8989.774	8989.774
		$\boldsymbol{\gamma}$ mse	9277.007	9277.007	9277.007	9277.007	9277.007
		Times of no convergence	4	4	4	4	4
	Global	$\boldsymbol{\beta}$ mse	1.578	0.498	0.343	0.354	0.210
		$\boldsymbol{\beta}$ bias	1.188	0.590	0.124	0.063	0.012
		$\boldsymbol{\beta}$ var	1.190	0.590	0.124	0.063	0.012
		$\boldsymbol{\gamma}$ bias	6.652	2.143	0.408	1.678	0.730
		$\boldsymbol{\gamma}$ var	13.258	6.637	1.306	0.662	0.014
		$\boldsymbol{\gamma}$ mse	13.302	6.642	1.306	0.663	0.014
		Times of no convergence	0	0	0	0	0

Case 1: Fix the observation number of single institution as $n_j = 100$.

We set $p = q = 3$ and let true parameters contain the intercept. Then, $\boldsymbol{\gamma}^* = [-1, 0.5, 0.7]^\top$, $\boldsymbol{\beta}^* = [0, 0.5, -1]^\top$ and $\boldsymbol{x}_{ji} \sim N_q(0_q, I_q), \boldsymbol{z}_{ji} \sim N_p(0.5 \times 1_p, I_p)$. Finally, all $y_{ji}, \boldsymbol{p}_{ji}, \boldsymbol{\lambda}_{ji}$ are generated by Equations (1) and (3).

To measure the estimation efficiency, we calculate the empirical mean square error (mse) as $\sum_{s=1}^{500} \|x^* - \hat{x}_s\|^2/500$, the empirical bias (bias) as $\| \sum_{s=1}^{500} \hat{x}_s/500 - x^*\|$ and the empirical variance (var) as $\sum_{s=1}^{500} \|\hat{x}_s - \sum_{s=1}^{500} \hat{x}_s/500\|^2/500$, where x^* stands for $\boldsymbol{\gamma}^*$ or $\boldsymbol{\beta}^*$ and $\hat{x}_s$ stands for estimtors of $\boldsymbol{\gamma}$ or $\boldsymbol{\beta}$ in the sth replicate. T/C stands for number of internal iterations of M step. 'Sub' stands for applying EM algorithm on one institution's data. 'Global' stands for applying EM algorithm on the whole data.

In Table 3, the performances of our proposed Algorithms 1–3 are all better than that of Sub, and getting better with the increase of the number of institutions (i.e., the total sample size). Algorithms 1–2's results are close to that

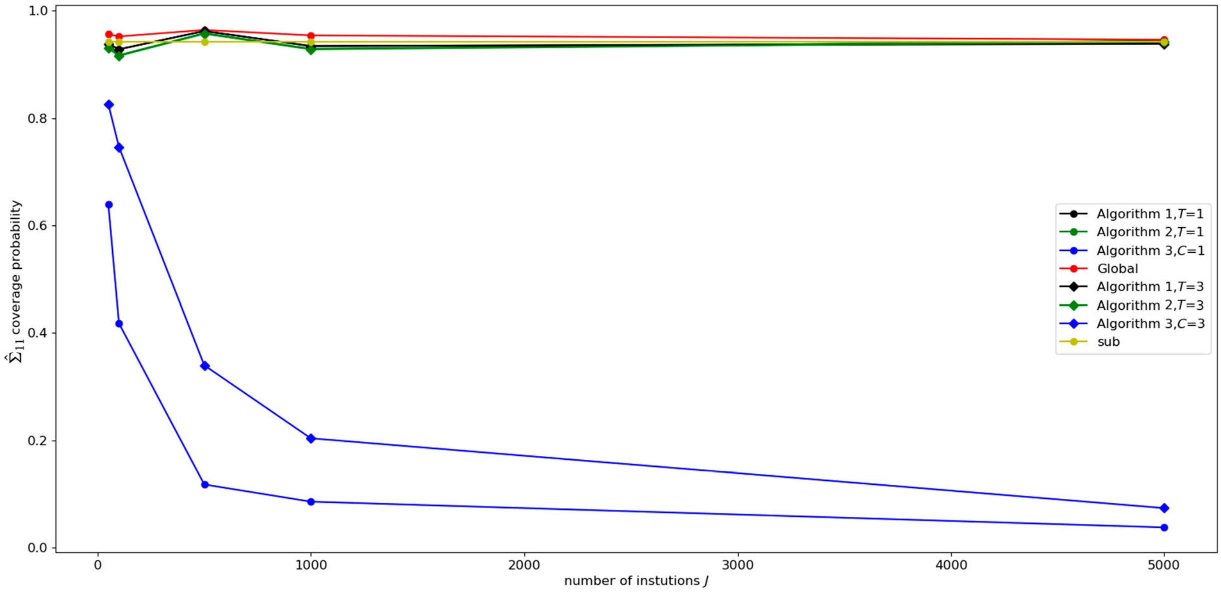

Figure 1. CPs for 95% confidence interval coverage probability for γ_1 in the setting of *Case 1*.

of Global, and the little difference between $T = 1$ and $T = 3$ means that we can get similar estimators under lower communication cost ($T = 1$) by these two algorithms. Algorithm 1 is equivalent to the Global method based on Newton optimization except for the selection of initial value and the number of internal iterations of M step, so the results of Algorithm 1 are similar to that of the Global method. The stability of Algorithm 2 is relatively poor. There are some nonconvergence times in repeated experiments, which is related to the sample size used to calculate Hessian matrix in M step, so this phenomenon does not improve with the increase of the number of institutions. There is a gap between the results of Algorithm 3 and the Global one. Increasing the number of institutions does not improve the results of Algorithm 3. However, when C increases, the results of Algorithm 3 improve significantly. The reason is that the M step results of Algorithm 3 are the weighted average of the results of local institutions. When C and the sample size of a single institution are fixed, increasing the number of institutions can reduce the variance of the results, but it does not reduce the bias. When other conditions are equal but C becomes larger, the final result will be improved because of the improvement of the results from each local institution of M step.

In summary, under the scenario of distributed data with small local sample size, Algorithm 1 is recommended if sufficient communication resources are equipped. Algorithm 2 with $T = 1$ is recommended for high restriction on communication and calculation. Algorithm 3 with large C is suggested to be selected if each institution has strong computing power and relatively low limitations on the communication cost.

Figure 1 shows the empirical coverage probabilities (CPs) of 95% confidence interval for the first parameter of $\boldsymbol{\gamma}$, i.e., γ_1. The way to construct the confidence interval is $\left[\hat{\boldsymbol{\gamma}}_1 - 1.96 \times \sqrt{\widehat{\Sigma}_{11}}/\sqrt{n}, \hat{\boldsymbol{\gamma}}_1 + 1.96 \times \sqrt{\widehat{\Sigma}_{11}}/\sqrt{n}\right]$.$\hat{\boldsymbol{\gamma}}_1$ comes from different algorithms. $\widehat{\Sigma}_{11}$ comes from Theorem 3.2. The empirical CPs for other parameters are similar and the details are omitted here.

The CPs of Algorithms 1 and 2 and the Global one approach 0.95 (the nominal level) as the total sample size increases. Since the sample size of the single institution does not change, the results of Sub method present a horizontal line. Based on the calculation principle of Algorithm 3, the increase of the number of institutions does not improve the bias but reduces the variance, and so makes the CPs worse.

Case 2: Fix the total sample size $n = 100000$.

Data generation is the same as *Case 1*. We fix the total sample size n but change the number of institutions.

In *Case 2*, we only change the number of institutions, i.e., the degree of data aggregation. All empirical mse, bias and variance are listed in Table 4. The results of Algorithms 1–2 change little, because the total sample size n does not change. However, the results of Algorithm 3 are improved when the sample size of a single institution increases. Algorithms 1–2's performance is still similar to the Global one, and there is little difference between $T = 1$ and $T = 3$. With the increase of the sample size of a single institution, the number of nonconvergence of Algorithm 2 decreases significantly, which is consistent with the analysis in *Case 1*. Due to the increasing of the data volume of a single institution, the results of each institution used in Algorithm 3 have been improved, which makes the corresponding weighted average result improve. Besides, the addition of C can also improve Algorithm 3's performance.

Table 4. Empirical performances of proposed Algorithms under different *a* with fixed total sample size.

			a (Sample size per institution)				
			100	500	1000	5000	10000
Algorithm 1	$T = 1$	β bias	0.354	0.354	0.314	0.354	0.354
		β var	0.063	0.063	0.064	0.063	0.063
		β mse	0.064	0.063	0.064	0.064	0.064
		γ bias	1.068	1.068	0.902	1.068	1.068
		γ var	0.662	0.662	0.667	0.662	0.662
		γ mse	0.663	0.663	0.668	0.663	0.663
		Times of no convergence	0	0	12	0	0
	$T = 3$	β bias	0.354	0.354	0.313	0.354	0.354
		β var	0.063	0.063	0.064	0.063	0.063
		β mse	0.064	0.064	0.064	0.064	0.064
		γ bias	1.070	1.069	0.902	1.068	1.068
		γ var	0.662	0.662	0.667	0.662	0.662
		γ mse	0.663	0.663	0.668	0.663	0.663
		Times of no convergence	0	0	12	0	0
Algorithm 2	$T = 1$	β bias	0.109	0.317	0.299	0.355	0.354
		β var	0.058	0.063	0.064	0.063	0.063
		β mse	0.058	0.063	0.064	0.064	0.064
		γ bias	1.497	1.002	0.884	1.072	1.070
		γ var	0.649	0.671	0.666	0.662	0.662
		γ mse	0.652	0.672	0.667	0.663	0.663
		Times of no convergence	188	17	13	0	0
	$T = 3$	β bias	0.108	0.310	0.298	0.354	0.354
		β var	0.058	0.063	0.064	0.063	0.063
		β mse	0.058	0.063	0.064	0.064	0.064
		γ bias	1.549	0.954	0.883	1.068	1.068
		γ var	0.651	0.670	0.666	0.662	0.662
		γ mse	0.653	0.671	0.667	0.663	0.663
		Times of no convergence	195	18	13	0	0
Algorithm 3	$C = 1$	β bias	51.328	9.474	4.686	1.121	0.522
		β var	0.060	0.063	0.063	0.063	0.063
		β mse	2.695	0.152	0.085	0.065	0.064
		γ bias	170.462	37.374	19.021	4.460	1.846
		γ var	0.492	0.624	0.650	0.660	0.615
		γ mse	29.549	2.020	1.012	0.680	0.649
		Times of no convergence	0	0	12	0	0
	$C = 3$	β bias	27.563	4.960	2.497	0.726	0.702
		β var	0.60	0.063	0.063	0.063	0.063
		β mse	0.820	0.087	0.070	0.064	0.064
		γ bias	95.150	19.603	10.049	2.745	2.650
		γ var	0.557	0.641	0.658	0.661	0.661
		γ mse	9.611	1.025	0.759	0.668	0.668
		Times of no convergence	0	0	12	0	0
Sub		β bias	24.305	5.808	2.135	1.578	0.498
		β var	97.445	13.253	6.296	1.188	0.590
		β mse	98.036	13.287	6.300	1.190	0.590
		γ bias	535.941	56.343	29.383	6.652	2.143
		γ var	8989.774	142.272	66.520	13.258	6.638
		γ mse	9277.007	145.446	67.383	13.302	6.642
		Times of no convergence	4	0	0	0	0
Global		β bias	0.354	0.354	0.354	0.354	0.354
		β var	0.063	0.063	0.063	0.063	0.063
		β mse	0.064	0.064	0.064	0.064	0.064
		γ bias	1.068	1.068	1.068	1.068	1.068
		γ var	0.662	0.662	0.662	0.662	0.662
		γ mse	0.663	0.663	0.663	0.663	0.663
		Times of no convergence	0	0	0	0	0

In particular, the simulation results show that the performances of these three algorithms are similar when the sample size of single institution is large enough. Therefore, for a distributed data structure, when the sample size of a single institution is large, Algorithm 3 with $T = 1$ is more recommended to save calculation and communication costs. Otherwise, recommendations from *Case 1* are useful.

Similar to *Case 1*, Figure 2 shows the empirical coverage probabilities (CPs) of 95% confidence interval for γ_1. In a summary, CPs of all different algorithms approach 0.95 with the increasing of sample size of a single institution. For Algorithm 3, with the decrease of the number of institutions, the bias of parameter estimation decreases significantly, and the variance slightly increases. At the same time, the sample size used to calculate the variance increases, making the CPs improved.

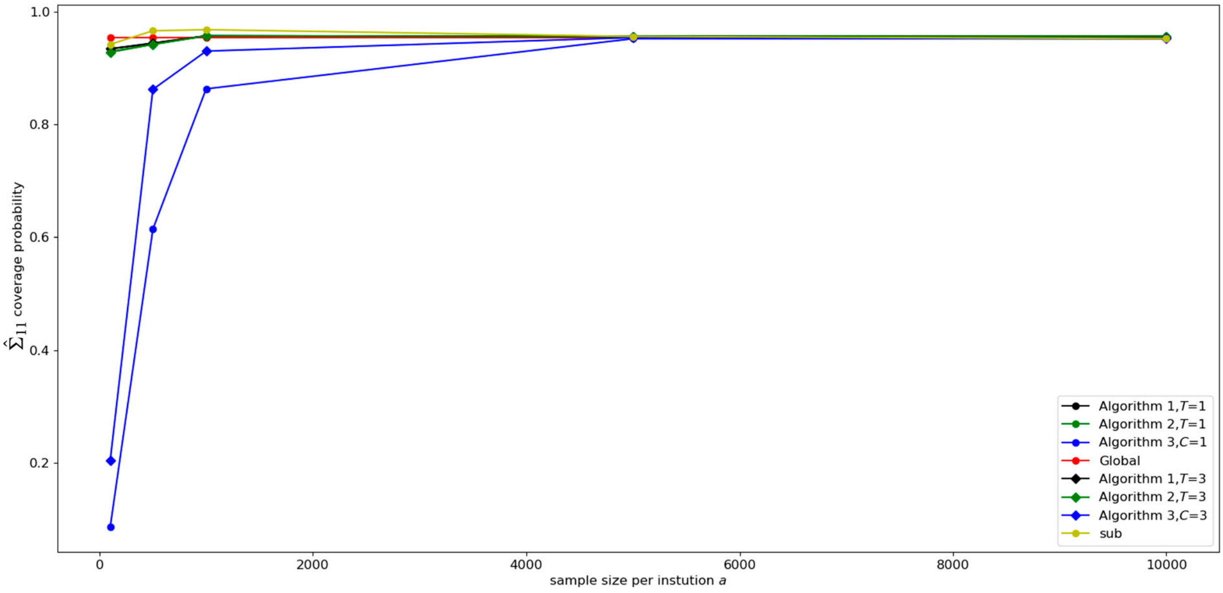

Figure 2. CPs for 95% confidence interval coverage probability for γ_1 in the setting of *Case 2*.

5.2. Heterogeneous data

In order to fully compare the algorithms in different scenarios, we set up simulation scenarios that are inconsistent with the theoretical conditions. This section shows the performance of the proposed algorithms when the data's distribution in each institution is slightly different. Specifically, $\{\boldsymbol{x}_{ji}, \boldsymbol{z}_{ji}\}$ of different institution (i.e., $j = 1, \ldots, J$) comes from different distributions: $\boldsymbol{x}_{ij} \sim N_q(10^{-3} \times j, I_q + \mathrm{diag}(10^{-3} \times j)), \boldsymbol{z}_{ij} \sim N_p(10^{-3} \times j, I_p + \mathrm{diag}(10^{-3} \times j))$, where $\mathrm{diag}(A)$ denotes as a square matrix with A as diagonal element and 0 for the rest.

Case 3: Fix the total sample size $n = 100000$ with heterogeneous data.

Except the generation of $\{\boldsymbol{x}_{ji}, \boldsymbol{z}_{ji}\}$, all other settings are the same as in *Case 2*. All empirical mse, bias and variance are listed in Table 5. We find that Algorithms 1 – 2 can also achieve similar results to the Global one when the data in different institutions comes from different distributions. Unlike *Case 2*, Algorithm 2 is less stable. When $a = 100$, most of the 500 repetitions do not converge. The reason is that when the sample size of a single institution is small and the data in each institution is distributed differently, the approximation of Hessian matrix adopted by M step of Algorithm 2 will become very poor. However, with the increasing of sample size in single institution, the stability of Algorithm 2 is acceptable, indicating that it will perform well if a is large enough, even if its theoretical assumptions are not satisfied.

When $a = 5000$, nonconvergence times for Algorithms 1–2 are the same, while Algorithm 3 has zero nonconvergence times. It shows that Algorithm 3 is more stable under the same condition in the case of heterogeneous data. Therefore, if the sample size of a single institution is small, it is suggested to choose Algorithm 1. Under the condition of limited communication and computing costs, if the sample size of a single institution is not too small, Algorithm 3 with large C will be a better choice. If the local sample size is large enough, Algorithm 2 can be an alternative.

Figure 3 shows the 95% confidence interval coverage probability for the first parameter in $\boldsymbol{\gamma}$, i.e., γ_1. The trend of different algorithms in this graph is similar to that in Figure 2.

6. Real case study

In this section, we apply proposed algorithms to car insurance data. The data is collected from Kaggle[1] with total sample size $n = 10, 300$, and the data were randomly divided into K groups, $K = 20, 10, 5$. The conclusion is similar, only the results of $K = 10$ are presented here, and other results are presented in the appendix. CLM_FREQ is the response variable we are interested in, representing the number of past policy claims from the insured person, which is zero-inflated with 61% claims being zero. All possible covariates are shown in Table 6.

We set $X = Z$, and Tables 7–8 show the parameters and standard deviation estimates of the different algorithms with $T/C = 3, 1, K = 10$.

[1] https://www.kaggle.com/kerneler/starter-car-insurance-claim-data-62f4f91c-d.

Table 5. Empirical performances of proposed Algorithms for heterogeneous data with fixed total sample size.

			a (Sample size per institution)			
			100	500	5000	10000
Algorithm 1	$T = 1$	β bias	0.849	0.374	0.212	0.406
		β var	0.061	0.049	0.054	0.051
		β mse	0.062	0.049	0.054	0.051
		γ bias	3.71	0.764	0.596	1.903
		γ var	0.728	0.789	0.804	0.708
		γ mse	0.742	0.789	0.804	0.712
		Times of no convergence	0	0	8	0
	$T = 3$	β bias	0.849	0.374	0.212	0.406
		β var	0.061	0.049	0.054	0.051
		β mse	0.062	0.049	0.054	0.051
		γ bias	3.715	0.762	0.595	0.190
		γ var	0.728	0.789	0.804	0.708
		γ mse	0.742	0.789	0.804	0.712
		Times of no convergence	0	0	8	0
Algorithm 2	$T = 1$	β bias	0.744	0.408	0.212	0.406
		β var	0.065	0.055	0.049	0.051
		β mse	0.065	0.055	0.049	0.051
		γ bias	2.817	1.975	0.684	1.901
		γ var	0.758	0.759	0.795	0.708
		γ mse	0.766	0.763	0.796	0.711
		Times of no convergence	329	8	8	0
	$T = 3$	β bias	0.977	0.416	0.212	0.406
		β var	0.067	0.049	0.054	0.051
		β mse	0.068	0.049	0.054	0.051
		γ bias	3.642	0.703	0.595	1.903
		γ var	0.832	0.795	0.804	0.708
		γ mse	0.848	0.795	0.804	0.712
		Times of no convergence	382	8	8	0
Algorithm 3	$C = 1$	β bias	56.673	11.113	1.007	0.845
		β var	0.346	0.052	0.054	0.051
		β mse	3.558	0.176	0.055	0.051
		γ bias	188.624	44.185	4.905	3.851
		γ var	1.568	0.737	0.796	0.704
		γ mse	37.147	2.689	0.820	0.719
		Times of no convergence	0	0	0	0
	$C = 3$	β bias	30.534	5.837	0.528	0.622
		β var	0.155	0.050	0.054	0.051
		β mse	1.087	0.084	0.054	0.051
		γ bias	106.560	23.072	2.750	2.868
		γ var	0.960	0.755	0.800	0.706
		γ mse	12.315	1.287	0.807	0.714
		Times of no convergence	0	0	0	0
Sub		β bias	14.790	4.604	0.668	0.574
		β var	63.133	11.882	1.033	0.517
		β mse	63.351	11.903	1.033	0.517
		γ bias	651.831	53.909	6.186	2.537
		γ var	11589.205	188.760	16.519	7.800
		γ mse	12014.147	191.666	16.558	7.807
		Times of no convergence	16	0	0	0
Global		β bias	0.128	0.372	0.212	0.406
		β var	0.058	0.049	0.054	0.051
		β mse	0.058	0.049	0.054	0.051
		γ bias	2.565	0.750	0.606	1.903
		γ var	0.069	0.789	0.804	0.708
		γ mse	0.070	0.789	0.805	0.712
		Times of no convergence	3	0	0	0

Table 6. Covariates details in real case study.

Covariates name	Covariates details
HOMEKIDS	Number of children at home.
MSTATUS	Marital status, 1 = married, 0 = unmarried.
REVOKED	License revoked, 1 = yes, 0 = no.
MVR_PTS	Motor vehicle record points.
URBANICITY	Urban vs. rural home/work area, 1 = Rural, 0 = Urban.
CAR_USE	Vehicle use, 1 = Commercial, 0 = Private.

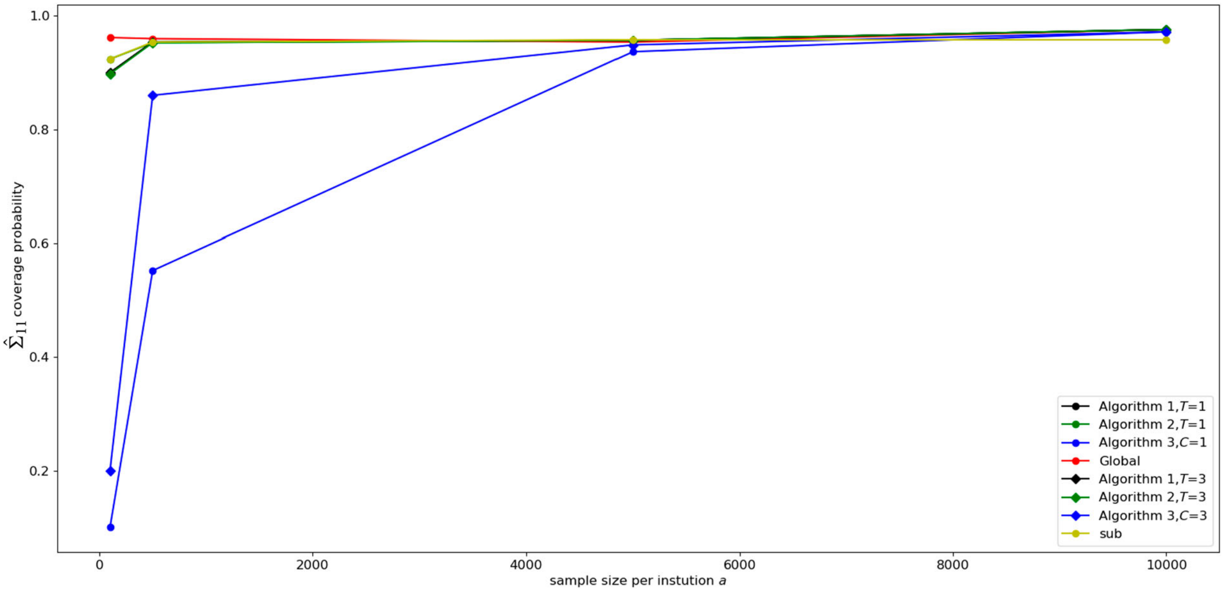

Figure 3. Case 3 95% confidence interval coverage probability for γ_1 .

Table 7. Estimationg results with $T/C = 3, K = 10$ for the car insurance data.

		β	β sd.	γ	γ sd.
INTERCEPT	Algorithm 1	0.516	0.024	0.885	0.062
	Algorithm 2	0.516	0.024	0.885	0.062
	Algorithm 3	0.519	0.023	0.881	0.062
	Global	0.516	0.024	0.885	0.062
	Sub	0.526	0.076	1.202	0.208
HOMEKIDS	Algorithm 1	−0.020	0.010	−0.175	0.032
	Algorithm 2	−0.020	0.010	−0.175	0.032
	Algorithm 3	−0.019	0.010	−0.173	0.032
	Global	−0.020	0.009	−0.175	0.030
	Sub	-0.071	0.035	−0.195	0.103
MSTATUS	Algorithm 1	−0.026	0.020	0.347	0.063
	Algorithm 2	−0.026	0.020	0.347	0.063
	Algorithm 3	−0.028	0.020	0.343	0.062
	Global	−0.026	0.021	0.347	0.063
	Sub	−0.068	0.065	0.059	0.202
REVOKED	Algorithm 1	−0.063	0.027	−0.411	0.090
	Algorithm 2	−0.063	0.027	−0.411	0.090
	Algorithm 3	−0.059	0.027	−0.404	0.089
	Global	−0.063	0.030	−0.411	0.098
	Sub	0.109	0.082	−0.530	0.302
MVR_PTS	Algorithm 1	0.019	0.004	−0.631	0.017
	Algorithm 2	0.019	0.004	−0.631	0.017
	Algorithm 3	0.019	0.004	−0.625	0.017
	Global	0.019	0.004	−0.631	0.017
	Sub	0.008	0.012	−0.699	0.060
URBANICITY	Algorithm 1	−0.044	0.039	2.008	0.102
	Algorithm 2	−0.044	0.039	2.008	0.102
	Algorithm 3	−0.046	0.040	1.984	0.101
	Global	−0.044	0.044	2.008	0.092
	Sub	−0.041	0.152	1.651	0.290
CAR_USE	Algorithm 1	0.032	0.020	−0.382	0.065
	Algorithm 2	0.032	0.020	−0.382	0.065
	Algorithm 3	0.033	0.020	−0.378	0.064
	Global	0.032	0.021	−0.382	0.064
	Sub	0.046	0.066	−0.566	0.214

Based on our model, the parameter $\boldsymbol{\gamma}$ is related to p_i. Our estimation results for $\boldsymbol{\gamma}$ indicate that, if a policyholder has a small number of children at home, is married, has never had their driver's license revoked, has few motor vehicle record points, lives in a rural area, and uses their vehicle for personal purposes, they are more likely to have a large p_i. A larger p_i means that the policyholder is more likely to come from the 0 state, meaning they have a lower probability of making a claim. This result is reasonable. If the policyholder fits the above description, they are likely to be a young person living in a rural area, owning a private car, and with good driving habits. There are fewer

Table 8. Estimationg results with $T/C = 1, K = 10$ for the car insurance data.

		β	β sd.	γ	γ sd.
INTERCEPT	Algorithm 1	0.516	0.024	0.885	0.062
	Algorithm 2	0.516	0.024	0.885	0.062
	Algorithm 3	0.521	0.024	0.877	0.061
	Global	0.516	0.024	0.885	0.062
	Sub	0.526	0.076	1.202	0.208
HOMEKIDS	Algorithm 1	−0.020	0.010	−0.175	0.032
	Algorithm 2	−0.020	0.010	−0.175	0.032
	Algorithm 3	−0.018	0.010	−0.171	0.032
	Global	−0.020	0.009	−0.175	0.030
	Sub	−0.071	0.035	−0.195	0.103
MSTATUS	Algorithm 1	−0.026	0.020	0.347	0.063
	Algorithm 2	−0.026	0.020	0.347	0.063
	Algorithm 3	−0.030	0.020	0.338	0.062
	Global	−0.026	0.021	0.347	0.063
	Sub	−0.068	0.065	0.059	0.202
REVOKED	Algorithm 1	−0.063	0.027	−0.411	0.090
	Algorithm 2	−0.063	0.027	−0.411	0.090
	Algorithm 3	−0.055	0.027	−0.396	0.089
	Global	−0.063	0.030	−0.411	0.098
	Sub	0.109	0.082	−0.530	0.302
MVR_PTS	Algorithm 1	0.019	0.004	−0.631	0.017
	Algorithm 2	0.019	0.004	−0.631	0.017
	Algorithm 3	0.019	0.004	−0.619	0.017
	Global	0.019	0.004	−0.631	0.017
	Sub	0.008	0.012	−0.699	0.060
URBANICITY	Algorithm 1	−0.044	0.039	2.008	0.102
	Algorithm 2	−0.044	0.039	2.008	0.102
	Algorithm 3	−0.049	0.040	1.962	0.099
	Global	−0.044	0.044	2.008	0.092
	Sub	−0.041	0.152	1.651	0.290
CAR_USE	Algorithm 1	0.032	0.020	−0.382	0.065
	Algorithm 2	0.032	0.020	−0.382	0.065
	Algorithm 3	0.033	0.020	−0.373	0.064
	Global	0.032	0.021	−0.382	0.064
	Sub	0.046	0.066	−0.566	0.214

vehicles in rural areas, and policyholders are more likely to have good driving habits, which would result in a lower likelihood of major accidents. Furthermore, insurance companies often offer attractive NCD policy for private cars, which would also result in a larger p_i for this type of policyholder.

The mean of the Poisson distribution, λ_i, which represents the average number of claims for a policyholder, is determined by the parameter $\boldsymbol{\beta}$. According to the estimating results, various factors, such as having a small number of children at home, being unmarried, having a clean driving record, having a high number of motor vehicle record points, residing in a city, and using the vehicle for commercial purposes, can result in a higher λ_i, indicating a greater average number of claims. Living in an urban area, using the vehicle for commercial purposes, and not having a private car can all increase the likelihood of being involved in road accidents. Additionally, commercial policyholders may not be attracted to the insurance company's NCD policy, further contributing to a higher number of claims.

As can be seen from Tables 7–8, the results of Sub differ furthest from Global, and the results of Algorithms 1–3 are relatively close to Global, where results in Algorithms 1 and 2 are closer than Algorithm 3. When T or C is 1 or 3, the estimation results of Algorithms 1 and 2 are basically the same, and the results of Algorithm 3 are slightly changed. These results are in good agreement with the theoretical and simulation results.

7. Conclusion and discussion

The study of data structures with zero-inflation is crucial in practical applications. In particular, when the data has a distributed structure, it becomes imperative to develop efficient algorithms with effective communication. This paper presents three distributed algorithms, with two of them being improved to enhance communication efficiency through various techniques. Subsequently, several simulation scenarios were established to compare the algorithms and identify their respective strengths and limitations. The performance of the algorithms in real-life cases is also reported.

The results of the study indicate that while Algorithm 2 has the lowest communication cost, it is less stable when the sample size of a single institution is small, especially for heterogeneous data. Algorithms 1 and 3 are comparatively stable but consume more resources. The development of a more robust algorithm remains a challenge to be addressed. Additionally, this paper only focuses on the ZIP model applied to zero-inflation data and does not cover ZIB model or models with random effect terms. These limitations provide opportunities for further research and improvement.

Funding

This work was supported by the National Natural Science Foundation of China [Grant Number 11771268].

ORCID

Yang Bai http://orcid.org/0000-0002-4660-4542

References

Cohen, A. C. (1963). Estimation in mixtures of discrete distributions. In *Proceedings of the international symposium on discrete distributions* (pp. 373–378).

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, *39*(1), 1–22.

Gu, D. (2008). Distributed EM algorithm for Gaussian mixtures in sensor networks. *IEEE Transactions on Neural Networks*, *19*(7), 1154–1166. https://doi.org/10.1109/TNN.2008.915110

Hall, D. B. (2000). Zero-inflated Poisson and binomial regression with random effects: A case study. *Biometrics*, *56*(4), 1030–1039. https://doi.org/10.1111/j.0006-341X.2000.01030.x

Johnson, N. L., & Kotz, S. (1970). Distributions in statistics: Discrete distributions. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, *133*(3), 482–483.

Jordan, M. I., Lee, J. D., & Yang, Y. (2018). Communication-efficient distributed statistical inference. *Journal of the American Statistical Association*, *114*(526), 668–681. https://doi.org/10.1080/01621459.2018.1429274

Lambert, D. (1992). Zero-inflated Poisson regression, with an application to defects in manufacturing. *Technometrics*, *34*(1), 1–14. https://doi.org/10.2307/1269547

Lee, A. H., Wang, K., Scott, J. A., Yau, K. K., & McLachlan, G. J. (2006). Multi-level zero-inflated Poisson regression modelling of correlated count data with excess zeros. *Statistical Methods in Medical Research*, *15*(1), 47–61. https://doi.org/10.1191/0962280206sm429oa

Mota, J. F., Xavier, J. M., Aguiar, P. M., & Püschel, M. (2013). D-ADMM: A communication-efficient distributed algorithm for separable optimization. *IEEE Transactions on Signal Processing*, *61*(10), 2718–2723. https://doi.org/10.1109/TSP.2013.2254478

Nowak, R. D. (2003). Distributed EM algorithms for density estimation and clustering in sensor networks. *IEEE Transactions on Signal Processing*, *51*(8), 2245–2253. https://doi.org/10.1109/TSP.2003.814623

Redner, R. A., & Walker, H. F. (1984). Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review*, *26*(2), 195–239. https://doi.org/10.1137/1026034

Shamir, O., Srebro, N., & Zhang, T. (2014). Communication-efficient distributed optimization using an approximate Newton-type method. *Proceedings of the 31st International Conference on Machine Learning*, 32(2):1000–1008.

Tang, Y., Xiang, L., & Zhu, Z. (2014). Risk factor selection in rate making: EM adaptive LASSO for zero-inflated poisson regression models. *Risk Analysis*, *34*(6), 1112–1127. https://doi.org/10.1111/risa.2014.34.issue-6

Wu, C. J. (1983). On the convergence properties of the EM algorithm. *The Annals of Statistics*, *11*(1), 95–103. https://doi.org/10.1214/aos/1176346060

Zangwill, W. I. (1969). *Nonlinear Programming: A Unified Approach* (Vol. 52). New Jersey: Prentice-Hall.

Zhang, Y., Duchi, J. C., & Wainwright, M. J. (2013). Communication-efficient algorithms for statistical optimization. *The Journal of Machine Learning Research*, *14*(1), 3321–3363.

Zhu, X., Li, F., & Wang, H. (2021). Least squares approximation for a distributed system. *Journal of Computational and Graphical Statistics*, *30*(4), 1004–1018. https://doi.org/10.1080/10618600.2021.1923517

Appendices

Appendix 1. Proof the consistency and asymptotic normality of proposed estimators

This section deduces the theoretical results in Section 3. Firstly, we introduce some notations.

Let $\boldsymbol{\theta} = [\boldsymbol{\gamma}^\top, \boldsymbol{\beta}^\top]^\top$, and then Equation (4) can be denoted as $L(\boldsymbol{\theta})$. Equation (7) can be denoted as $L_c(\boldsymbol{\theta})$. The dimension of $\boldsymbol{\theta}$ is $r = p + q$. Denote the complete data $\{\mathbf{x}, y, \mathbf{z}, u\}$ as d_c, and the density function as $f(d_c \mid \boldsymbol{\theta}), \boldsymbol{\theta} \in \Omega$. Denote the uncomplete data $\{\mathbf{x}, y, \mathbf{z}\}$ as d_p, and the corresponding density function as $g(d_p \mid \boldsymbol{\theta})$.

More importantly, we denote the conditional density of $d_c \mid d_p$ as $b(d_c \mid d_p, \boldsymbol{\theta}) = f(d_c \mid \boldsymbol{\theta})/g(d_p \mid \boldsymbol{\theta})$. Then, we have $L(\boldsymbol{\theta}) = \log g(d_p \mid \boldsymbol{\theta}) = \log f(d_c \mid \boldsymbol{\theta}) - \log b(d_c \mid d_p, \boldsymbol{\theta}) = L_c(\boldsymbol{\theta}) - \log b(d_c \mid d_p, \boldsymbol{\theta})$.

We define $Q(\boldsymbol{\theta}' \mid \boldsymbol{\theta}) = E(\log f(d_c \mid \boldsymbol{\theta}') \mid d_p, \boldsymbol{\theta}) = E(L_c(\boldsymbol{\theta}') \mid d_p, \boldsymbol{\theta})$ and $H(\boldsymbol{\theta}' \mid \boldsymbol{\theta}) = E(\log b(d_c \mid d_p, \boldsymbol{\theta}') \mid d_p, \boldsymbol{\theta})$. Then, we have $Q(\boldsymbol{\theta}' \mid \boldsymbol{\theta}) = L(\boldsymbol{\theta}') + H(\boldsymbol{\theta}' \mid \boldsymbol{\theta})$.

Finally, denote the iteration sequence obtained by EM algorithm in the process of solving $\boldsymbol{\theta}$ as $\{\boldsymbol{\theta}^{(k)}\}$. According to the above notations, the E step of the kth-step of EM algorithm calculates $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(k)})$, and M step calculates $\boldsymbol{\theta}^{(k+1)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(k)})$.

A.1 Convergence of EM algorithm

Assumptions:

(1) $\Omega \subset \mathbb{R}^r$: Ω is a subset in the r-dimensional Euclidean space.
(2) $\Omega_{\boldsymbol{\theta}^{(0)}} := \{\boldsymbol{\theta} \in \Omega : L(\boldsymbol{\theta}) \geq L(\boldsymbol{\theta}^{(0)})\}$ is compact $\forall L(\boldsymbol{\theta}^{(0)}) > -\infty$ and $\Omega_{\boldsymbol{\theta}^{(0)}}$ is in the interior of Ω for $\boldsymbol{\theta}^{(0)} \in \Omega$ which is the initial value of EM algorithm.
(3) L is continuous in Ω and differentiable in the interior of Ω.

We can finish proving the convergence of EM Algorithm by the following two proofs.

(1) $L(\boldsymbol{\theta}^{(k)})$ is monotonic.

Proof: Firstly, prove $\forall(\boldsymbol{\theta}^{(k+1)}, \boldsymbol{\theta}^{(k)}) \in \Omega \times \Omega$. We have $H(\boldsymbol{\theta}^{(k+1)} \mid \boldsymbol{\theta}^{(k)}) \leq H(\boldsymbol{\theta}^{(k)} \mid \boldsymbol{\theta}^{(k)})$. And the equal sign can be obtained if and only if $b(d_c \mid d_p, \boldsymbol{\theta}^{(k+1)}) = b(d_c \mid d_p, \boldsymbol{\theta}^{(k)})$. Substitute relevant expressions as follows:

$$\begin{aligned}
&E\left[\log b(d_c \mid d_p, \boldsymbol{\theta}^{(k+1)}) \mid d_p, \boldsymbol{\theta}^{(k)}\right] - E\left[\log b(d_c \mid d_p, \boldsymbol{\theta}^{(k)}) \mid d_p, \boldsymbol{\theta}^{(k)}\right] \\
&= E\left[\log \frac{b(d_c \mid d_p, \boldsymbol{\theta}^{(k+1)})}{b(d_c \mid d_p, \boldsymbol{\theta}^{(k)})} \middle| d_p, \boldsymbol{\theta}^{(k)}\right] \\
&= \int \log \frac{b(d_c \mid d_p, \boldsymbol{\theta}^{(k+1)})}{b(d_c \mid d_p, \boldsymbol{\theta}^{(k)})} b(\mathrm{d}_c \mid d_p, \boldsymbol{\theta}^{(k)}) \mathrm{d}(d_c) \\
&\leq \log \int \frac{b(d_c \mid d_p, \boldsymbol{\theta}^{(k+1)})}{b(d_c \mid d_p, \boldsymbol{\theta}^{(k)})} b(d_c \mid d_p, \boldsymbol{\theta}^{(k)}) \mathrm{d}(d_c) \\
&= 0.
\end{aligned}$$

Since that log is a concave function, the inequality sign above can be obtained by applying Jensen's inequality. When $b(d_c \mid d_p, \boldsymbol{\theta}^{(k+1)})/b(d_c \mid d_p, \boldsymbol{\theta}^{(k)})$ is a constant, we have '$=$'. Because of the property of the distribution density function, we have $b(d_c \mid d_p, \boldsymbol{\theta}^{(k+1)}) = b(d_c \mid d_p, \boldsymbol{\theta}^{(k)})$.

Next, we prove that the GEM algorithm (defined by Dempster et al. (1977)) satisfies $L(M(\boldsymbol{\theta})) \geq L(\boldsymbol{\theta}), \forall \boldsymbol{\theta} \in \Omega$. Equal sign is established when $Q(M(\boldsymbol{\theta}) \mid \boldsymbol{\theta}) = Q(\boldsymbol{\theta} \mid \boldsymbol{\theta})$ and $b(\mathbf{x} \mid \mathbf{y}, M(\boldsymbol{\theta})) = b(\mathbf{x} \mid \mathbf{y}, \boldsymbol{\theta})$ are true almost everywhere. M represents an iterative algorithm, i.e., $\boldsymbol{\theta}^{(k)} \to \boldsymbol{\theta}^{(k+1)}$ defined by $\boldsymbol{\theta}^{(k+1)} \in M(\boldsymbol{\theta}^{(k)})$. ■

Definition A.1: If an iterative algorithm defined by M satisfies

$$Q(M(\boldsymbol{\theta}) \mid \boldsymbol{\theta}') \geq Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}'), \quad \forall\, \boldsymbol{\theta} \in \Omega,$$

it is called the GEM algorithm (generalized EM algorithm, Dempster et al. (1977)).

Based on this definition, the proof is as follows:

$$L(M(\boldsymbol{\theta})) - L(\boldsymbol{\theta}) = Q(M(\boldsymbol{\theta}) \mid \boldsymbol{\theta}') - Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}') + H(\boldsymbol{\theta} \mid \boldsymbol{\theta}') - H(M(\boldsymbol{\theta}) \mid \boldsymbol{\theta}').$$

In the above equation, $Q(M(\boldsymbol{\theta}) \mid \boldsymbol{\theta}') - Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}') \geq 0$ is guaranteed by the GEM algorithm definition. $H(\boldsymbol{\theta} \mid \boldsymbol{\theta}') - H(M(\boldsymbol{\theta}) \mid \boldsymbol{\theta}') \geq 0$ is guaranteed by the previous proof. '$=$' is true only if $Q(M(\boldsymbol{\theta}) \mid \boldsymbol{\theta}') = Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}')$ and $H(\boldsymbol{\theta} \mid \boldsymbol{\theta}') = H(M(\boldsymbol{\theta}) \mid \boldsymbol{\theta}')$. Based on the proof before, this condition is $b(d_c \mid d_p, \boldsymbol{\theta}^{(k+1)}) = b(d_c \mid d_p, \boldsymbol{\theta}^{(k)})$. According to the definition of EM algorithm, it belongs to GEM algorithm. Therefore the monotony of $L(\boldsymbol{\theta}^{(k)})$ is proved.

(2) $L(\boldsymbol{\theta}^{(k)})$ is bounded.

Proof: From $\Omega_{\boldsymbol{\theta}^{(0)}} \subset \Omega$, we have $\Omega_{\boldsymbol{\theta}^{(0)}} \subset \mathbb{R}^r$. $\Omega_{\boldsymbol{\theta}^{(0)}}$ is a compact set, and then $\Omega_{\boldsymbol{\theta}^{(0)}}$ is a bounded closed set. Since L is continuous in Ω, and then L is upper bounded in $\Omega_{\boldsymbol{\theta}^{(0)}}$. Since $\{\boldsymbol{\theta}^{(k)}\} \subset \Omega_{\boldsymbol{\theta}^{(0)}}$, and then L is upper bounded in $\{\boldsymbol{\theta}^{(k)}\}$, $\forall \boldsymbol{\theta}^{(0)}$. ■

Based on above Proofs (1)–(2), the convergence of the algorithm can be proved. Now we need to prove that the convergence point is the maximum point of $L(\boldsymbol{\theta})$. Since $L(\boldsymbol{\theta})$ is unimodal on Ω and has a unique maximum value, it is only necessary to prove that the convergence point is the stagnation point of $L(\boldsymbol{\theta})$.

A.2 The convergence point of $\{\theta^{(k)}\}$ is the stagnation point of $L(\theta)$

Let's start with the Global Convergence Theorem (GCT), and the details of the proof refer to page 91 of Zangwill (1969).

Proposition A.1 (Global Convergence Theorem): *Let the sequence $\{x_k\}_{k=0}^{\infty}$ be generated by $x_{k+1} \in M(x_k)$, where M is a point-to-set map on X. Let a solution set $\Gamma \subset X$ be given, and suppose that*

(i) *all points x_k are contained in a compact set $S \subset X$;*
(ii) *M is closed over the complement of Γ;*
(iii) *there is a continuous function α on X such that(a) if $x \notin \Gamma, \alpha(y) > \alpha(x)$ for all $y \in M(x)$, and (b) if $x \in \Gamma, \alpha(y) \geq \alpha(x)$ for all $y \in M(x)$.*

Then all the limit points of x_k are in the solution set Γ and $\alpha(x_k)$ converges monotonically to $\alpha(x)$ for some $x \in \Gamma$.

Based on GCT, we take M as the mapping in the EM iterative algorithm, and denote $\alpha(x)$ as log-likelihood L, and corresponding solution set as Γ. Defining point set $\mathcal{M} = \{$local maxima in the interior of $\Omega\}$ and $\mathcal{T} = \{$stationary points in the interior of $\Omega\}$, following conclusions can be obtained.

Proposition A.2: *$\{\boldsymbol{\theta}^{(k)}\}$ is the GEM sequence, and the generation mode is $\boldsymbol{\theta}^{(k+1)} \in M\left(\boldsymbol{\theta}^{(k)}\right)$. If*

(i) *M is a point-to-set map on the complement of $\mathcal{T}$ (or $\mathcal{M}$);*
(ii) *$L\left(\boldsymbol{\theta}^{(k+1)}\right) > L\left(\boldsymbol{\theta}^{(k)}\right) \forall \boldsymbol{\theta}^{(k)} \notin \mathcal{T}$ (or $\mathcal{M}$), then all limit points of $\boldsymbol{\theta}^{(k)}$ are stationary points (or maximum points) of L, and $L\left(\boldsymbol{\theta}^{(k)}\right)$ monotonically converges to $L^* = L(\boldsymbol{\theta}^*)$, where $\boldsymbol{\theta}^* \in \mathcal{T}$ (or $\mathcal{M}$).*

Proposition A.3: *If $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}')$ is continuous at $\boldsymbol{\theta}, \boldsymbol{\theta}'$, all the limit points of EM algorithm sequence $\left\{\boldsymbol{\theta}^{(k)}\right\}$ are stationary points of L, and $L\left(\boldsymbol{\theta}^{(k)}\right)$ converges to $L^* = L(\boldsymbol{\theta}^*)$, where $\boldsymbol{\theta}^*$ is a stationary point of L.*

To finish proving that the convergence point of $\{\boldsymbol{\theta}^{(k)}\}$ is the stagnation point of $L(\boldsymbol{\theta})$, we only need to prove $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}')$ is continuous at $\boldsymbol{\theta}, \boldsymbol{\theta}'$.

$$
\begin{aligned}
E(u_i \mid y_i) = \mathrm{P}(u_i = 1 \mid y_i) = \frac{\mathrm{P}(u_i = 1, y_i)}{y_i} &= \frac{\mathrm{P}(y_i \mid u_i = 1)\mathrm{P}(u_i = 1)}{\mathrm{P}(y_i, u_i = 1) + \mathrm{P}(y_i, u_i = 0)} \\
&= \frac{\mathrm{P}(y_i \mid u_i = 1)\mathrm{P}(u_i = 1)}{\mathrm{P}(y_i \mid u_i = 1)\mathrm{P}(u_i = 1) + \mathrm{P}(y_i \mid u_i = 0)\mathrm{P}(u_i = 0)} \\
&= \begin{cases} 0, & y_i > 0, \\ \dfrac{p_i}{p_i + (1 - p_i)\mathrm{e}^{-\lambda_i}}, & y_i = 0. \end{cases}
\end{aligned}
$$

Substituting the result into $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}') = E[L_c(\boldsymbol{\theta} \mid y, u) \mid y, \boldsymbol{\theta}']$ gives

$$
Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}') = \begin{cases} \displaystyle\sum_{i=1}^{n} - \log\left(1 + \mathrm{e}^{z_i^\top \boldsymbol{\gamma}}\right) + y_i x_i^\top \boldsymbol{\beta} - \mathrm{e}^{x_i^\top \boldsymbol{\beta}} - \log(y_i!), & y_i > 0, \\ \displaystyle\sum_{i=1}^{n} \left[z_i^\top \boldsymbol{\gamma} + \mathrm{e}^{x_i^\top \boldsymbol{\beta}}\right] \frac{\exp\left(z_i^\top \boldsymbol{\gamma}' + \mathrm{e}^{x_i^\top \boldsymbol{\beta}'}\right)}{1 + \exp\left(z_i^\top \boldsymbol{\gamma}' + \mathrm{e}^{x_i^\top \boldsymbol{\beta}'}\right)} - \log\left(1 + \mathrm{e}^{z_i^\top \boldsymbol{\gamma}}\right) - \mathrm{e}^{x_i^\top \boldsymbol{\beta}}, & y_i = 0. \end{cases}
$$

The expression right here gives us $Q(\boldsymbol{\theta}|\boldsymbol{\theta}')$ is continuous at $\boldsymbol{\theta}, \boldsymbol{\theta}'$.

A.3 Asymptotic equivalence of EM Jordan approximation

In this section, we illustrate that the M step of EM iteration of Algorithm 2 is equivalent to the result before using Jordan approximation.

For the M step of the kth EM iteration, we make the following notations.

- F is a second-order differentiable loss function defined on a sample W, and $\boldsymbol{\theta}$ is parameter. $F^{*(k)} = E\left[F\left(\boldsymbol{\theta}; W, \boldsymbol{\theta}^{(k-1)}\right)\right]$.
- True parameter of M step $\boldsymbol{\theta}^{*(k)} := \arg\max_{\boldsymbol{\theta} \in \Omega} E\left[F\left(\boldsymbol{\theta}; W, \tilde{\boldsymbol{\theta}}^{(k-1)}\right)\right]$.
- Local loss function $F_j^{(k)} = \frac{1}{n_j} \sum_{i=1}^{n_j} F\left(\boldsymbol{\theta}; w_{ji}, \tilde{\boldsymbol{\theta}}^{(k-1)}\right)$.
- Global loss function $F_n^{(k)} = \frac{1}{n} \sum_{j=1}^{J} \sum_{i=1}^{n_j} F\left(\boldsymbol{\theta}; w_{ji}, \tilde{\boldsymbol{\theta}}^{(k-1)}\right)$, and F_n is the objective function of M step.

Based on Algorithm 2, $F_n^{(k)} = E\left[L_c(\boldsymbol{\theta})|\boldsymbol{z}, \boldsymbol{x}, y, \tilde{\boldsymbol{\theta}}^{(k-1)}\right]/n$, $F\left(\boldsymbol{\theta}; w_{ji}, \tilde{\boldsymbol{\theta}}^{(k-1)}\right) = E\left[u_{ji}\boldsymbol{z}_{ji}^\top \boldsymbol{\gamma} - \log\left(1 + \mathrm{e}^{\boldsymbol{z}_{ji}^\top \boldsymbol{\gamma}}\right) + (1 - u_{ji}) \left(y_{ji}\boldsymbol{x}_{ji}^\top \boldsymbol{\beta} - \mathrm{e}^{\boldsymbol{x}_{ji}^\top \boldsymbol{\beta}}\right) - (1 - u_{ji}) \log(y_{ji}!) \mid \boldsymbol{x}_{ji}, \boldsymbol{z}_{ji}, y_{ji}, \tilde{\boldsymbol{\theta}}^{(k-1)}\right]$. Since that n_j has the same order with a, then we can suppose $n_j = a, j = 1, \ldots, J$, and $\hat{\boldsymbol{\theta}}^{(k)}$ is M step result before approximating. The following proof is for the result of each EM iteration. For convenience, superscript has been omitted. Under our proposed notations, the following properties are satisfied.

- *(Local convexity)* The Hessian matrix $I(\boldsymbol{\theta}) = -\nabla^2 F^*(\boldsymbol{\theta})$ is invertible at $\boldsymbol{\theta}^*$: there exist two positive constants (μ_-, μ_+), such that $\mu_- I_r \preccurlyeq I(\boldsymbol{\theta}^*) \preccurlyeq \mu_+ I_r$.
- *(Identifiability)* For any $\delta > 0$, there exists $\epsilon > 0$, such that $\liminf_{n\to\infty} \mathrm{P}\{\inf_{\|\boldsymbol{\theta}^*-\boldsymbol{\theta}\|_2 \geq \delta}(F(\boldsymbol{\theta}^*) - F(\boldsymbol{\theta})) \geq \epsilon\} = 1$.
- *(Smoothness)* Let $U(\rho) = \{\boldsymbol{\theta} \in \mathbb{R}^r | \parallel \boldsymbol{\theta} - \boldsymbol{\theta}^* \parallel_2 \leq \rho\} \subset \Omega$, and there exist constants (G, L) and a function $N(w)$ such that $E[\|\nabla F(\boldsymbol{\theta}; W)\|_2^{16}] \leq G^{16}$, $E[\|\nabla^2 F(\boldsymbol{\theta}; W) - I(\boldsymbol{\theta})\|_2^{16}] \leq L^{16}$, $\forall \boldsymbol{\theta} \in U(\rho)$; $\|\nabla^2 F(\boldsymbol{\theta}; w) - \nabla^2 F(\boldsymbol{\theta}'; w)\|_2 \leq N(w) \times \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_2$, $\forall \boldsymbol{\theta}, \boldsymbol{\theta}' \in U(\rho)$. Moreover, $N(w)$ satisfies $E[N^{16}(W)] \leq N^{16}$ for some constant $N > 0$.

Then, we have the following two results.

Proposition A.4: *Suppose that the initial estimator $\bar{\boldsymbol{\theta}}$ satisfies $\parallel \bar{\boldsymbol{\theta}} - \boldsymbol{\theta}^* \parallel_2 \leq \min\{\rho, (16N)^{-1}(1-\rho)\mu_-\}$. Then M step result $\tilde{\boldsymbol{\theta}}$ satisfies*

$$\left\|\tilde{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}\right\|_2 \leq C_2 \left(\|\bar{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}\|_2 + \left\|\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\right\|_2 + \left\|\nabla^2 F_1\left(\boldsymbol{\theta}^*\right) - \nabla^2 F_n\left(\boldsymbol{\theta}^*\right)\right\|_2\right) \|\bar{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}\|_2,$$

with probability at least $1 - C_1 a^{-8}$, where C_1, C_2 are independent of (J, n, a).

Based on the results of the above proposition and $\|\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2 = O_p(1/\sqrt{n})$, if $\|\bar{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2 = O_p(1/\sqrt{a})$, we have $\|\tilde{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}\|_2 \leq C_3/\sqrt{a}\|\bar{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}\|_2$, where $\|\nabla^2 F_1(\boldsymbol{\theta}^*) - \nabla^2 F_n(\boldsymbol{\theta}^*)\|_2 = O_p(1/\sqrt{a})$ can be derived based on Lemma 7 in Zhang et al. (2013). This conclusion means that after one iteration in M step, that is, $T = 1$, the error of the estimator with respect to $\hat{\boldsymbol{\theta}}$ will drop to $1/\sqrt{a}$ times.

Proposition A.5: *Suppose that the M step initial estimator $\bar{\boldsymbol{\theta}}$ satisfies $\parallel \bar{\boldsymbol{\theta}} - \boldsymbol{\theta}^* \parallel_2 \leq \min\{\rho, (16N)^{-1}(1-\rho)\mu_-\}$, $\parallel \bar{\boldsymbol{\theta}} - \boldsymbol{\theta}^* \parallel_2 = O_p(\frac{1}{\sqrt{a}})$. Then M step result $\tilde{\boldsymbol{\theta}}$ satisfies*

$$\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^* = -I'(\boldsymbol{\theta}^*)^{-1}\nabla F_n(\boldsymbol{\theta}^*) + O_p(n^{-1} + a^{-1/2} \parallel \bar{\boldsymbol{\theta}} - \boldsymbol{\theta}^* \parallel_2);$$

moreover, if $\parallel \bar{\boldsymbol{\theta}} - \boldsymbol{\theta}^* \parallel_2 = o_p\left(\sqrt{\frac{a}{n}}\right)$, then

$$\sqrt{n}(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*) \xrightarrow{d} N(0, \Sigma'), n \to \infty,$$

where $\Sigma' := I'(\boldsymbol{\theta}^*)^{-1}E[\nabla F(\boldsymbol{\theta}^*; w)\nabla F(\boldsymbol{\theta}^*; w)^\top]I'(\boldsymbol{\theta}^*)^{-1}$ and $I'(\boldsymbol{\theta}^*) := \nabla^2 F^*(\boldsymbol{\theta}^*)$.

For completeness of the elaboration and to avoid ambiguity, the proof of Propositions A.4–5 is omitted, the details of which can be derived based on Lemmas 6–8 in Zhang et al. (2013).

A.4 Consistency and asymptotic normality of the estimator from Algorithm 2

The proof in A.3 ensures that the Jordan approximation of the M step of each EM iteration converges to the result before approximation. According to the results from A.1 and A.2, the Jordan approximation of the EM Algorithm converges to the MLE of the original objective function, that is, the result of Algorithm 2 converges to the MLE of the original objective function. Hence, the asymptotic properties of the proposed estimation are obtained.

For now, the proofs of Theorems 3.1–3.2 are finished.

Appendix 2. Results of real case study with $K = 20, 5$

This section shows the result of Real Case Study with $K = 20, 5$, and the conclusion is similar to that of $K = 10$.

According to the results of $K = 20, 10, 5$, under the same T/C, the results of Algorithm 1 are slightly different due to the change of initial value. Algorithms 2 and 3 are improved by increasing the sample size of a single institution due to the improvement of the Hessian matrix approximation and the optimization results of a single institution, respectively. Global method remains unchanged because it uses the same data. Sub is also improving due to the increase in the amount of data used.

Table A1. Estimationg results with $T/C = 3, K = 20$ for the car insurance data.

		β	β sd.	γ	γ sd.
INTERCEPT	Algorithm 1	0.516	0.024	0.885	0.062
	Algorithm 2	0.516	0.024	0.885	0.062
	Algorithm 3	0.520	0.024	0.876	0.061
	Global	0.516	0.024	0.885	0.062
	Sub	0.477	0.103	1.288	0.292
HOMEKIDS	Algorithm 1	−0.020	0.011	−0.175	0.034
	Algorithm 2	−0.020	0.011	−0.175	0.034
	Algorithm 3	−0.018	0.009	−0.175	0.030
	Global	−0.020	0.009	−0.175	0.030
	Sub	−0.072	0.050	−0.110	0.143
MSTATUS	Algorithm 1	−0.026	0.021	0.347	0.067
	Algorithm 2	−0.026	0.021	0.347	0.067
	Algorithm 3	−0.030	0.021	0.338	0.066
	Global	−0.026	0.021	0.347	0.063
	Sub	−0.029	0.096	−0.183	0.295
REVOKED	Algorithm 1	−0.063	0.028	−0.411	0.082
	Algorithm 2	−0.063	0.028	−0.411	0.082
	Algorithm 3	−0.053	0.027	−0.392	0.081
	Global	−0.063	0.030	−0.411	0.098
	Sub	0.247	0.105	−0.533	0.379
MVR_PTS	Algorithm 1	0.019	0.004	−0.631	0.018
	Algorithm 2	0.019	0.004	−0.631	0.018
	Algorithm 3	0.020	0.004	−0.619	0.018
	Global	0.019	0.004	−0.631	0.017
	Sub	0.009	0.017	−0.718	0.089
URBANICITY	Algorithm 1	−0.044	0.040	2.008	0.104
	Algorithm 2	−0.044	0.040	2.008	0.104
	Algorithm 3	−0.044	0.040	1.955	0.101
	Global	−0.044	0.044	2.008	0.092
	Sub	−0.151	0.217	1.527	0.383
CAR_USE	Algorithm 1	0.032	0.020	−0.382	0.066
	Algorithm 2	0.032	0.020	−0.382	0.066
	Algorithm 3	0.033	0.020	−0.374	0.065
	Global	0.032	0.021	−0.382	0.064
	Sub	0.040	0.093	-0.350	0.305

Table A2. Estimationg results with $T/C = 1, K = 20$ for the car insurance data.

		β	β sd.	γ	γ sd.
INTERCEPT	Algorithm 1	0.516	0.024	0.885	0.062
	Algorithm 2	0.516	0.024	0.885	0.062
	Algorithm 3	0.524	0.023	0.869	0.061
	Global	0.516	0.024	0.885	0.062
	Sub	0.477	0.103	1.288	0.292
HOMEKIDS	Algorithm 1	−0.020	0.011	−0.175	0.034
	Algorithm 2	−0.020	0.011	−0.175	0.034
	Algorithm 3	−0.017	0.010	−0.166	0.033
	Global	−0.020	0.009	−0.175	0.030
	Sub	−0.072	0.050	−0.110	0.143
MSTATUS	Algorithm 1	−0.026	0.021	0.347	0.067
	Algorithm 2	−0.026	0.021	0.347	0.067
	Algorithm 3	−0.033	0.021	0.329	0.065
	Global	−0.026	0.021	0.347	0.063
	Sub	−0.029	0.096	−0.183	0.295
REVOKED	Algorithm 1	−0.063	0.028	−0.411	0.082
	Algorithm 2	−0.063	0.028	−0.411	0.082
	Algorithm 3	−0.045	0.027	−0.377	0.081
	Global	−0.063	0.030	−0.411	0.098
	Sub	0.247	0.105	−0.534	0.379
MVR_PTS	Algorithm 1	0.019	0.004	−0.631	0.018
	Algorithm 2	0.019	0.004	−0.631	0.018
	Algorithm 3	0.020	0.004	−0.607	0.017
	Global	0.019	0.004	−0.631	0.017
	Sub	0.009	0.017	−0.718	0.089
URBANICITY	Algorithm 1	−0.044	0.040	2.008	0.104
	Algorithm 2	−0.044	0.040	2.008	0.104
	Algorithm 3	−0.044	0.040	1.908	0.098
	Global	−0.044	0.044	2.008	0.092
	Sub	−0.151	0.217	1.527	0.382
CAR_USE	Algorithm 1	0.032	0.020	−0.382	0.066
	Algorithm 2	0.032	0.020	−0.382	0.066
	Algorithm 3	0.034	0.020	−0.367	0.065
	Global	0.032	0.021	−0.382	0.064
	Sub	0.040	0.093	−0.350	0.305

Table A3. Estimationg results with $T/C = 3, K = 5$ for the car insurance data.

		β	β sd.	γ	γ sd.
INTERCEPT	Algorithm 1	0.516	0.024	0.885	0.062
	Algorithm 2	0.516	0.024	0.885	0.062
	Algorithm 3	0.517	0.024	0.882	0.062
	Global	0.516	0.024	0.885	0.062
	Sub	0.520	0.053	1.032	0.139
HOMEKIDS	Algorithm 1	−0.020	0.010	−0.175	0.031
	Algorithm 2	−0.020	0.010	−0.175	0.031
	Algorithm 3	−0.020	0.010	−0.174	0.031
	Global	−0.020	0.009	−0.175	0.030
	Sub	−0.035	0.023	−0.209	0.070
MSTATUS	Algorithm 1	−0.026	0.021	0.347	0.064
	Algorithm 2	−0.026	0.021	0.347	0.064
	Algorithm 3	−0.027	0.021	0.344	0.063
	Global	−0.026	0.021	0.347	0.063
	Sub	−0.068	0.046	0.117	0.138
REVOKED	Algorithm 1	−0.063	0.028	−0.411	0.094
	Algorithm 2	−0.063	0.028	−0.411	0.094
	Algorithm 3	−0.060	0.028	−0.405	0.094
	Global	−0.063	0.030	−0.411	0.098
	Sub	0.084	0.058	−0.427	0.211
MVR_PTS	Algorithm 1	0.019	0.004	−0.631	0.017
	Algorithm 2	0.019	0.004	−0.631	0.017
	Algorithm 3	0.019	0.004	−0.628	0.017
	Global	0.019	0.004	−0.631	0.017
	Sub	0.018	0.008	−0.633	0.039
URBANICITY	Algorithm 1	−0.044	0.041	2.008	0.104
	Algorithm 2	−0.044	0.041	2.008	0.104
	Algorithm 3	−0.046	0.041	1.992	0.103
	Global	−0.044	0.044	2.008	0.092
	Sub	−0.111	0.106	1.650	0.202
CAR_USE	Algorithm 1	0.032	0.020	−0.382	0.066
	Algorithm 2	0.032	0.020	−0.382	0.066
	Algorithm 3	0.033	0.020	−0.379	0.066
	Global	0.032	0.021	−0.382	0.064
	Sub	0.051	0.046	−0.363	0.144

Table A4. Estimationg results with $T/C = 1, K = 5$ for the car insurance data.

		β	β sd.	γ	γ sd.
INTERCEPT	Algorithm 1	0.516	0.024	0.885	0.062
	Algorithm 2	0.516	0.024	0.885	0.062
	Algorithm 3	0.518	0.024	0.880	0.061
	Global	0.516	0.024	0.885	0.062
	Sub	0.520	0.053	1.032	0.139
HOMEKIDS	Algorithm 1	−0.020	0.010	−0.175	0.031
	Algorithm 2	−0.020	0.010	−0.175	0.031
	Algorithm 3	−0.019	0.010	−0.172	0.031
	Global	−0.020	0.009	−0.175	0.030
	Sub	−0.035	0.023	−0.209	0.070
MSTATUS	Algorithm 1	−0.026	0.021	0.347	0.064
	Algorithm 2	−0.026	0.021	0.347	0.064
	Algorithm 3	−0.028	0.021	0.340	0.063
	Global	−0.026	0.021	0.347	0.063
	Sub	−0.068	0.046	0.117	0.138
REVOKED	Algorithm 1	−0.063	0.028	−0.411	0.094
	Algorithm 2	−0.063	0.028	−0.411	0.094
	Algorithm 3	−0.056	0.028	−0.400	0.094
	Global	−0.063	0.030	−0.411	0.098
	Sub	0.084	0.058	−0.427	0.211
MVR_PTS	Algorithm 1	0.019	0.004	−0.631	0.017
	Algorithm 2	0.019	0.004	−0.631	0.017
	Algorithm 3	0.019	0.004	−0.624	0.017
	Global	0.019	0.004	−0.631	0.017
	Sub	0.018	0.008	−0.633	0.039
URBANICITY	Algorithm 1	−0.044	0.041	2.008	0.104
	Algorithm 2	−0.044	0.041	2.008	0.104
	Algorithm 3	−0.048	0.041	1.975	0.102
	Global	−0.044	0.044	2.008	0.092
	Sub	−0.111	0.106	1.650	0.202
CAR_USE	Algorithm 1	0.032	0.020	−0.382	0.066
	Algorithm 2	0.032	0.020	−0.382	0.066
	Algorithm 3	0.034	0.020	−0.376	0.065
	Global	0.032	0.021	−0.382	0.064
	Sub	0.051	0.046	−0.363	0.144