Statistica Sinica Preprint No: SS-2023-0302	
Title	Information-based Optimal Subdata Selection for Clusterwise Linear Regression
Manuscript ID	SS-2023-0302
URL	http://www.stat.sinica.edu.tw/statistica/
DOI	10.5705/ss.202023.0302
Complete List of Authors	Yanxi Liu, John Stufken and Min Yang
Corresponding Authors	Min Yang
E-mails	minyang.stat@gmail.com

Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302

Information-based Optimal Subdata Selection for Clusterwise Linear Regression

Yanxi Liu, John Stufken, and Min Yang

AbbVie Inc., George Mason University, and University of Illinois at Chicago

Abstract: Mixture-of-Experts (MoE) models are commonly used when there exist distinct clusters with different relationships between the independent and dependent variables. Fitting such models for large datasets, however, is computationally virtually impossible. An attractive alternative is to use a subdata selected by "maximizing" the Fisher information matrix. A major challenge is that no closed-form expression for the Fisher information matrix is available for such models. Focusing on clusterwise linear regression models, a subclass of MoE models, we develop a framework that overcomes this challenge. We prove that the proposed subdata selection approach is asymptotically optimal, i.e., no other method is statistically more efficient than the proposed one when the full data size is large.

Key words and phrases: D-optimality; Information matrix; Latent indicator; Massive data; MLE

1. Introduction

Modern information technologies, such as cloud computing, internet of things, social networking, etc., are drivers for exponential growth of the size of datasets. Size may now be measured by TB and even PB instead of MB and GB (Cai and Zhu, 2015). While the extraordinary amount of data offers unprecedented opportunities for scientific discoveries and advancement, it also poses unprecedented challenges for analysis. These challenges are typically amplified by the complexity of the data and the speed with which it must be analyzed. A critical question for the statistics community is how to detect statistical relationships within high volumes of data with a complicated structure and turn it into actionable knowledge (Bühlmann et al., 2016).

With large datasets, relationships between input and output variables may no longer be homogeneous. Linear models or generalized linear models, which are effective when relationships are homogeneous, may be inadequate in the era of big data. One strategy for dealing with heterogeneity is through Mixture-of-Experts (MoE) models. The rationale for MoE models is to uncover hidden clusters within the data, such that within each cluster relationships between input and output variables can be adequately modeled by a single regression or classification model. While any such regression or classification model may be inadequate for the entire dataset, it may be just fine for a more homogeneous cluster. Flexibility and interpretability of MoE models has resulted in their broad use in regression, classification, and fusion applications in healthcare, finance, surveillance, and recognition (Yuksel et al., 2012).

The flexibility that MoE models provide goes however hand in hand with a high computational cost. The parameters of an MoE model are usually estimated using an EM algorithm, which requires a considerable computing time for each

Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302

iteration when the data size is large. In addition, since the EM algorithm usually converges to a local rather than global optimum (Balakrishnan et al., 2017; Wu, 1983), different initial values of the parameters must be considered for better estimation results. This makes this approach inefficient and daunting for large datasets (Makkuva et al., 2019).

An attractive idea, which has received considerable attention for dealing with massive data (*full data*), is selection and analysis of a much smaller subset of the data (*subdata*). One possible strategy is to use a model-free sampling approach. Some recent work includes, but is not limited to, Chang (2023) for developing a subdata selection method for large-scale computer experiments based on expected improvement optimization; Dai et al. (2023) for proposing adaptive subsampling with the minimum energy criterion; and Chang (2024) for developing a stratified sampling approach in a supervised learning framework. Model-based sampling approaches tend to perform much better when the model is, approximately, correctly specified. They can however be poor if the model is incorrectly specified, which is why model-free methods have gained in popularity. Nonetheless, with the flexibility of MoE models, we have found that our model-based approach based on these models performs well on all datasets that we have studied.

Model-based sampling approaches can overcome the computational burden. But they may reduce the information about the parameters contained in the original full data. For example, Wang et al. (2019); Cheng et al. (2020) proved that for linear and logistic regression models the information contained in the subdata selected by using popular random subsampling methods, including uniform random sampling, is asymptotically limited by the subdata size when the full data size becomes large.

The Information-Based Optimal Subdata Selection (IBOSS) method (Wang et al., 2019), which selects subdata judiciously, is computationally efficient and does not suffer from this limitation. For fitting a linear model, it is shown in Wang et al. (2019) that, if each independent variable has a distribution in the domain of attraction of the generalized extreme value distribution, the variances of the estimators of the slope parameters based on analyzing subdata converge to zero when the full data size grows even though the subdata size is fixed. Studying properties for information-based subdata selection under generalized linear and nonlinear models is more challenging because there are no closed-form expressions for estimators and information matrices depend on the unknown parameters. Cheng et al. (2020) developed a two-stage IBOSS-based subdata selection algorithm for logistic regression models and proved, for selected cases, that the information matrices based on subdata of a fixed size increase with the full data size. Inspired by the properties of orthogonal arrays, Wang et al. (2021) proposed an orthogonal subsampling (OSS) approach for big data with a focus on linear regression models. OSS is closely related to the IBOSS strategy since it attempts to minimize the average variance of the parameter estimators. For more related literature on subdata selection, readers are referred to the recent review paper (Yu et al., 2023).

With the IBOSS strategy, the goal is to select subdata that maximizes a function of the Fisher information matrix for the parameters of interest. This is even more challenging for MoE models than for generalized linear and nonlinear models and requires novel ideas. The fact that there is no closed-form expression for the information matrix under an MoE model prevents the use of optimal design techniques for selecting efficient subdata, which is the strategy that was used for linear and logistic regression models.

Focusing on the subclass of MoE models known as clusterwise linear regression models, we address this problem by using a surrogate matrix rather than the Fisher information matrix for guiding the subdata selection. We prove that the surrogate matrix is asymptotically equivalent to the information matrix under some mild conditions. We further prove that the statistical efficiency of the selection algorithm based on the surrogate matrix is asymptotically optimal, i.e., there exists no other method with better statistical efficiency in terms of convergence rate when the full data size becomes large.

In what follows, Section 2 introduces clusterwise linear regression models, while Section 3 presents the main results. Simulation studies and the analysis of real data are presented in Sections 4 and 5, respectively. Brief conclusions and possible future work are discussed in Section 6. All technical details are presented in the Appendix.

Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302

2. Mixture-of-Experts models and Clusterwise Linear Regression

Mixture-of-Experts models, which originated in the neural network literature (Jacobs et al., 1991), are widely popular regression and classification models in machine learning due to their flexibility in modeling and appealing interpretation (Masoudnia and Ebrahimpour, 2014). Rather than using a single model, MoE models are based on multiple models (or experts), which are mixed and combined, to provide great flexibility. MoE models assess how the data may be clustered into G clusters so that separate regression or classification models can be used in each cluster. In combination with many current regression and classification algorithms, empirical evidence shows that MoE models are powerful tools to study relationships among variables in a variety of settings, including healthcare, finance, social science, etc. (Yuksel et al., 2012).

Formally, let $(\mathbf{z}_i^T, y_i)$, $i = 1, \ldots, N$, be independent, where $\mathbf{z}_i = (z_{i1}, \ldots, z_{ip})^T$ is the covariate vector and y_i is the response for the ith observation. We also use $\mathbf{x}_i = (1, \mathbf{z}_i^T)^T$. In a Mixture-of-Experts model, there are G gate functions and G regression models (experts). While y_i is modeled by $\mathbf{x}_i$ through one of the experts, it is unknown which expert is employed. A latent indicator vector can be used to describe the connection. Let $\boldsymbol{I}_i = (I_{i1}, \ldots, I_{iG})$, where

$$I_{ig} = \begin{cases} 1 & \text{if the } g\text{th expert is employed,} \\ 0 & \text{otherwise.} \end{cases} \qquad (2.1)$$

The likelihood of $I_{ig} = 1$ is modeled by the gth gate function $P(I_{ig} = 1|\mathbf{z}_i)$. While more complicated choices are possible, and sometimes advisable, a popular simple choice is

$$P(I_{ig} = 1|\mathbf{z}_i) = \pi_g, \;\; g = 1, \ldots, G, \tag{2.2}$$

with $\sum_{g=1}^{G} \pi_g = 1$.

If $I_{ig} = 1$, then we can model the response y_i by $\mathbf{z}_i$ through the gth expert. The choice of the experts depends on the nature of the responses. For example, for a continuous response, a linear model may be appropriate for an expert; for a categorical response, experts may consist of generalized linear models.

While MoE models were coined by Jacobs et al. (1991), the idea can be traced back to Fair and Jaffee (1972) and Hosmer (1974), where the experts are linear regression models. Such models, with the choice for the gate function as in (2.2), were later called "clusterwise linear regression" (CLR) models (Späth, 1979) and have been widely applied in the social sciences, environmental studies, engineering, etc. (Brusco et al., 2003; Bagirov et al., 2017; Khadka and Paz, 2017). Research on CLR models is still ongoing, especially on developing efficient algorithms for alleviating the computational burden (Di Mari et al., 2017; Park et al., 2017). If $(\mathbf{z}_i^T, y_i)$ belongs to the gth cluster, i.e., $I_{ig} = 1$, then for a CLR model we write

$$y_i = \mathbf{x}_i^T \boldsymbol{\beta}_g + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma_g^2), \tag{2.3}$$

where $\boldsymbol{\beta}_g = (\beta_{0g}, \beta_{1g}, ..., \beta_{pg})$ and for any two distinct $g, g' \in \{1, ..., G\}$, $\boldsymbol{\beta}_g \neq \boldsymbol{\beta}_{g'}$. In the remainder, we will focus on CLR models.

Analysis of a CLR model is primarily based on the maximum likelihood approach (DeSarbo and Cron, 1988). From (2.3), the distribution of y_i is given by:

$$y_i \sim \sum_{g=1}^{G} \pi_g \phi(y_i | \mathbf{x}_i^T \boldsymbol{\beta}_g, \sigma_g^2) \qquad i = 1, ..., N \tag{2.4}$$

where $\phi(\cdot|\mu, \sigma^2)$ is the density function for the normal distribution with mean μ and variance σ^2. For simplicity of notation, we will write ϕ_{ig} instead of $\phi(y_i|\mathbf{x}_i^T\boldsymbol{\beta}_g, \sigma_g^2)$. The loglikelihood function given $\boldsymbol{y} = (y_1, ..., y_N)$ is then

$$l_{\boldsymbol{y}} = \sum_{i=1}^{N} \log \left(\sum_{g=1}^{G} \pi_g \phi_{ig} \right). \tag{2.5}$$

In contrast to a linear model, for a CLR model there is no closed-form expression for the MLE due to the summation over g in the loglikelihood function (2.5). In fact, without further restrictions there is an identifiability issue. Identifiability must be considered on equivalence classes of parameter vectors, so that two parameter vectors for which one can be obtained from the other by relabeling the clusters are considered to be equivalent. But even on such equivalence classes, identifiability is not automatic. For example, if the vectors $\mathbf{z}_i$ belong to a $(p-1)$-dimensional hyperplane, then the model is not even identifiable with $G = 1$ (i.e., for a single expert). Fortunately, Hennig (2000) gave a sufficient condition for identifiability of CLR model (2.3). Let $\mathcal{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$ and

$$h := \min \left\{ q : \mathcal{Z} \subset \bigcup_{i=1}^{q} H_i : H_i \in \mathcal{H}_{p-1} \right\}, \tag{2.6}$$

where $\mathcal{H}_{p-1}$ is the set of all hyperplanes of dimension $p - 1$.

Theorem 1 (Theorem 2.2, Hennig (2000)). *The CLR model in (2.3) is identifiable if $G < h$, where G is the number of clusters and h is defined in (2.6).*

The sufficient condition in Theorem 1 is relatively mild. As long as the covariate set $\mathcal{Z}$ cannot be covered by the union of G or fewer $(p-1)$-dimensional hyperplanes, identifiablity holds. Thus, loosely speaking, if the covariate values are sufficiently rich, then the sufficient condition holds and Model (2.3) is identifiable. For a big dataset, unless there are structural restrictions on the covariate values, we can expect identifiability to be satisfied. For example, the Structural Protein data used in Section 5 has $p = 1$ where the identifiability will be satisfied if the number of clusters is less than the number of unique covariate values. In the data, there are 129,711 unique covariate values, therefore if the number of clusters, $G < 129,711$, then identifiability is satisfied.

For a CLR model, with the unobservable indicator vector, the EM algorithm is the workhorse for finding the MLE (Yuksel et al., 2012). For given initial values of the parameters, the MLE is obtained by alternating between the expectation and maximization steps until convergence. However, the EM algorithm typically converges to a local optimum, and not necessarily to the global optimum (Wu, 1983; Balakrishnan et al., 2017). We generally need to try a large number of initial values to improve its performance. In addition, G, the number of clusters, is unknown. If $p = 1$, the number of clusters G can be determined easily by visualization (Hastie et al., 2009). However, if $p > 1$, then graphical methods may not work anymore. Some advanced techniques may be used. For example, AIC, BIC, Complete log-likelihood, etc. (Hawkins et al., 2001). Therefore we need to try different values of G to find the best one according to some criterion, such as AIC. Consequently, the computational cost for analyzing a CLR model is very high. For example, for simulated data of size $N = 10^7$ and $p = 10$ covariates, the computing time for fitting a linear regression model is around 0.2 seconds. In comparison, on the same platform, it takes around 470 seconds for fitting a CLR model with $G = 5$ being known and only one initial value. The computation time can be significantly increased due to the inclusion of numerous initial parameter values, as well as the consideration of different values for G. In this era, it is not uncommon for the data size to be in the millions or even billions, and the structure of the data can be more complicated. While high performance computing can be helpful, fitting MoE models for such big datasets still poses a tremendous challenge. This can be alleviated by using carefully selected subdata.

As indicated in the Introduction, the IBOSS strategy for subdata selection has been proven, both theoretically and empirically, to select highly informative subdata. Extending this strategy to CLR models would be extremely appealing for big data analysis, and would drastically reduce computational costs by fitting a CLR model to subdata that retains as much information about the parameters as possible.

To describe the IBOSS strategy, let $\boldsymbol{I}(\mathbf{x}_i)$ denote the information matrix for the ith data point. With $\delta_i = 1$ if the ith data point belongs to the subdata and $\delta_i = 0$ otherwise, and under the assumption of independence, the information matrix based on the subdata is

$$\boldsymbol{I}(\boldsymbol{\delta}) = \sum_{i=1}^{N} \delta_i \boldsymbol{I}(\mathbf{x}_i). \tag{2.7}$$

We want to select $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_N)$, subject to $\sum_{i=1}^{N} \delta_i = n$, to maximize, in some way, the information matrix in (2.7). For this maximization we adopt the approach from optimal design of experiments (Kiefer and Wolfowitz, 1959), where an interpretable function of $I(\boldsymbol{\delta})$ is used to induce a complete ordering of the information matrices. If Ψ is this function, then, subject to $\sum_{i=1}^{N} \delta_i = n$, we want to find subdata with indicator vector $\boldsymbol{\delta}^{opt}$ so that

$$\boldsymbol{\delta}^{opt} = \arg\max_{\boldsymbol{\delta}} \Psi(I(\boldsymbol{\delta})). \tag{2.8}$$

We will refer to any subdata selected in this way as IBOSS subdata. Algorithms for an approximate solution to this complex optimization problem can be based on the characterization of an optimal design for the corresponding model.

For the CLR model, the information matrix for the i-th data point can be written as $\boldsymbol{I}(\mathbf{x}_i) = E(\frac{\partial l_{y_i}}{\partial \boldsymbol{\theta}} \frac{\partial l_{y_i}}{\partial \boldsymbol{\theta}^T})$, where

$$l_{y_i} = log\big(\sum_{g=1}^{G} \pi_g \phi_{ig}\big), \tag{2.9}$$

and ϕ_{ig} is defined in (2.5). Here $\boldsymbol{\theta}$ is the vector of the $G(p+3)-1$ parameters, with $G(p+1)$ of them corresponding to the $\boldsymbol{\beta}_g$'s, G to the σ_g^2's, and $G-1$ to the π_g's. However, the summation structure within the log function in (2.9) prevents the derivation of a closed-form expression for $\boldsymbol{I}(\mathbf{x}_i)$. This in turn means that finding an optimal design is elusive, so that a new approach is needed for obtaining IBOSS subdata.

3. Main results

3.1 Bounding the Fisher information matrix

Without a closed-form expression for $\boldsymbol{I}(\mathbf{x}_i)$, we define a matrix that is larger than the Fisher information matrix in terms of the Loewner order and that has a closed-form expression. We first expand a data point from $(\mathbf{z}_i^T, y_i)$ to $(\mathbf{z}_i^T, y_i, \boldsymbol{I}_i)$, where $\boldsymbol{I}_i = (I_{i1}, ..., I_{iG})^T$ and I_{ig} is defined in (2.1). (Despite using the notation $\boldsymbol{I}$ both for an information matrix and a vector of latent indicators, the meaning will always be clear from the context.) The likelihood function under the CLR model for the complete ith data point $(\mathbf{z}_i^T, y_i, \boldsymbol{I}_i)$ is then given by

$$L_{C_i} = \prod_{g=1}^{G} \left[\phi_{ig}\pi_g\right]^{I_{ig}}. \tag{3.1}$$

Observe that $L_{C_i} = L_{y_i} \times L_{\mathbf{I}_i|y_i}$, where $L_{\mathbf{I}_i|y_i}$ is the likelihood function corresponding to the conditional distribution of $\mathbf{I}_i$ given y_i.

Corresponding to this factorization of the complete data likelihood function, we can write the Fisher information matrix for the i-th data point in the form of $\boldsymbol{I}(\mathbf{x}_i) = \boldsymbol{I}_{C_i} - \boldsymbol{I}_{M_i}$, where $\boldsymbol{I}_{C_i}$ is the complete data Fisher information matrix (or complete information matrix for short) based on the complete data likelihood function in (3.1) and $\boldsymbol{I}_{M_i}$ is the information matrix corresponding to the conditional distribution of $\mathbf{I}_i$ given y_i. The detailed derivation is presented in the Appendix. The expressions for $\boldsymbol{I}_{C_i}$ and $\boldsymbol{I}_{M_i}$ can be written as follows:

$$\boldsymbol{I}_{C_i} = blkdiag\left(\boldsymbol{I}_{\boldsymbol{\beta}|C_i}, \boldsymbol{I}_{\boldsymbol{\sigma^2}|C_i}, \boldsymbol{I}_{\boldsymbol{\pi}|C_i}\right) \tag{3.2}$$

where

$$\boldsymbol{I}_{\boldsymbol{\beta}|C_i} = blkdiag\left(\pi_1\frac{\mathbf{x}_i\mathbf{x}_i^T}{\sigma_1^2}, \pi_2\frac{\mathbf{x}_i\mathbf{x}_i^T}{\sigma_2^2}, \ldots, \pi_G\frac{\mathbf{x}_i\mathbf{x}_i^T}{\sigma_G^2}\right), \tag{3.3}$$

$$\boldsymbol{I}_{\boldsymbol{\sigma}^2|C_i} = blkdiag\left(\frac{\pi_1}{2\sigma_1^4}, \frac{\pi_2}{2\sigma_2^4}, \ldots, \frac{\pi_G}{2\sigma_G^4}\right), \tag{3.4}$$

and

$$\boldsymbol{I}_{\boldsymbol{\pi}|C_i} = blkdiag\left(\frac{1}{\pi_1}, \frac{1}{\pi_2}, \ldots, \frac{1}{\pi_{G-1}}\right) + \frac{1}{\pi_G}\boldsymbol{J}, \tag{3.5}$$

where $\boldsymbol{J}$ is matrix of ones. The expression for $\boldsymbol{I}_{M_i}$ is obtained by subtraction and its diagonal is given by

$$\left(diag(\boldsymbol{I}_{\boldsymbol{\beta}_1|M_i}), \ldots, diag(\boldsymbol{I}_{\boldsymbol{\beta}_G|M_i}), \boldsymbol{I}_{\boldsymbol{\sigma}_1^2|M_i}, \ldots, \boldsymbol{I}_{\boldsymbol{\sigma}_G^2|M_i}, \boldsymbol{I}_{\boldsymbol{\pi}_1|M_i}, \ldots, \boldsymbol{I}_{\boldsymbol{\pi}_{G-1}|M_i}\right),$$

where, for a square matrix $\boldsymbol{A} = (a_{ij})$, the notation $diag(\boldsymbol{A})$ denotes the diagonal matrix with diagonal entries a_{ii},

$$\begin{aligned}
\boldsymbol{I}_{\boldsymbol{\beta}_g|M_i} &= \mathbb{E}\Big\{w_{ig}(1-w_{ig})\frac{(y_i - \mathbf{x_i}^T\boldsymbol{\beta}_g)^2\mathbf{x_i}\mathbf{x_i}^T}{\sigma_g^4}\Big\}, \\
\boldsymbol{I}_{\boldsymbol{\sigma}_g^2|M_i} &= \mathbb{E}\left\{w_{ig}(1-w_{ig})\left[-\frac{1}{2\sigma_g^2} + \frac{(y_i - \mathbf{x_i}^T\boldsymbol{\beta}_g)^2}{2\sigma_g^4}\right]^2\right\}, \\
\boldsymbol{I}_{\boldsymbol{\pi}_g|M_i} &= \mathbb{E}\left\{\frac{w_{ig}(1-w_{ig})}{\pi_g^2} + \frac{w_{iG}(1-w_{iG})}{\pi_G^2} + 2\frac{w_{ig}w_{iG}}{\pi_g\pi_G}\right\},
\end{aligned} \tag{3.6}$$

and $w_{ig} = \frac{\pi_g\phi_{ig}}{\sum_{l=1}^G \pi_l\phi_{il}}$. A detailed derivation can be found in the Appendix.

3.2 Basic Strategy

Since we do not have a closed-form expression for $\boldsymbol{I}_{M_i}$, we face a significant hurdle in identifying subdata $\boldsymbol{\delta}^*$ that maximizes $det(\boldsymbol{I}(\boldsymbol{\delta}))$. To solve this dilemma, we first observe that for any $\boldsymbol{\delta}$, in Loewner order,

$$\begin{aligned} &\boldsymbol{I}(\boldsymbol{\delta}) = \sum_{i\in\boldsymbol{\delta}} (\boldsymbol{I}_{C_i} - \boldsymbol{I}_{M_i}) \leq \sum_{i\in\boldsymbol{\delta}} \boldsymbol{I}_{C_i}, \text{ so that} \\ &det(\boldsymbol{I}(\boldsymbol{\delta})) \leq det(\sum_{i\in\boldsymbol{\delta}} \boldsymbol{I}_{C_i}). \end{aligned} \tag{3.7}$$

The notation $\sum_{i\in\boldsymbol{\delta}}$ simply means that we sum only over those i for which $\delta_i = 1$. Based on (3.7), for a full data size N, if we have a strategy to find subdata $\boldsymbol{\delta}_N^*$ such that (a) $\boldsymbol{\delta}_N^* = \arg\max_{\boldsymbol{\delta}} det(\sum_{i\in\boldsymbol{\delta}} \boldsymbol{I}_{C_i})$ and (b) $\sum_{i\in\boldsymbol{\delta}_N^*} \boldsymbol{I}_{C_i} - \boldsymbol{I}(\boldsymbol{\delta}_N^*) \to 0$ when $N \to \infty$, then the subdata $\boldsymbol{\delta}_N^*$ is asymptotically optimal for maximizing $det(\boldsymbol{I}(\boldsymbol{\delta}))$.

Thus, for a fixed N, we need to identify a subdata selection strategy that leads to a $\boldsymbol{\delta}^*$ that gives an approximate solution for (a) and that satisfies the requirement in (b). Note that $det(\sum_{i\in\boldsymbol{\delta}} \boldsymbol{I}_{C_i})$ is proportional to $\left[det(\sum_{i\in\boldsymbol{\delta}} \mathbf{x}_i\mathbf{x}_i^T)\right]^G$, so that maximizing $det(\sum_{i\in\boldsymbol{\delta}} \boldsymbol{I}_{C_i})$ is equivalently to maximizing $det(\sum_{i\in\boldsymbol{\delta}} \mathbf{x}_i\mathbf{x}_i^T)$. Wang et al. (2019) develop the computationally inexpensive IBOSS algorithm for obtaining an approximate solution to precisely this problem.

Algorithm 1 (Algorithm 1 Wang et al. (2019)). With k as the subdata size and p as the number of covariates, assume for simplicity that $r = k/(2p)$ is an integer. Execute the following steps:

1. Select the data points with the r smallest and r largest values for the first covariate;

2. Sequentially, for $j = 2, ..., p$, exclude the data points that were previously selected, and select the data points with the r smallest and r largest values for the jth covariate from the remaining data points.

Thus, $\boldsymbol{\delta}^*$ obtained by using Algorithm 1 gives an approximate solution to the maximization of $det(\sum_{i\in\boldsymbol{\delta}} \boldsymbol{I}_{C_i})$. We still need to show that it also satisfies $\sum_{i\in\boldsymbol{\delta}^*} \boldsymbol{I}_{M_i} \to \mathbf{0}$ for $N \to \infty$. To circumvent that $\sum_{i\in\boldsymbol{\delta}^*} \boldsymbol{I}_{M_i}$ does not have a closed-form expression, we will show that, in the Loewner ordering, it is dominated by a diagonal matrix that converges to $\mathbf{0}$ when $N \to \infty$. This would immediately imply that $\sum_{i\in\boldsymbol{\delta}^*} \boldsymbol{I}_{M_i}$, which is a non-negative definite matrix, also converges to 0.

For $1 \le g_1, g_2 \le G$, let

$$\mathbf{f}_{i1}(g_1, g_2) = \; diag\left(\mathbf{x}_i\mathbf{x}_i^T \int \tilde{w}_i(g_1,g_2)\Delta^2_{\boldsymbol{\beta}_{ig_1}} dy_i\right), \tag{3.8}$$

$$f_{i2}(g_1, g_2) = \int \tilde{w}_i(g_1,g_2)\Delta^2_{\boldsymbol{\sigma}_{ig_1}} dy_i, \text{ and} \tag{3.9}$$

$$f_{i3}(g_1, g_2) = \int \tilde{w}_i(g_1,g_2) dy_i, \tag{3.10}$$

where $\tilde{w}_i(g_1,g_2) = \sqrt{\pi_{g_1}\phi_{ig_1}\pi_{g_2}\phi_{ig_2}}$, $\Delta_{\boldsymbol{\beta}_{ig}} = \frac{y_i - \mathbf{x}_i^T\beta_g}{\sigma_g^2}$ and $\Delta_{\boldsymbol{\sigma}_{ig}} = \frac{(y_i-\mathbf{x}_i^T\beta_g)^2-\sigma_g^2}{2\sigma_g^4}$. We consider

$$\boldsymbol{Q}^i = diag\left(blkdiag\left(\boldsymbol{Q}^i_{\boldsymbol{\beta}}, \boldsymbol{Q}^i_{\boldsymbol{\sigma^2}}, \boldsymbol{Q}^i_{\boldsymbol{\pi}}\right)\right), \tag{3.11}$$

where, for matrices or scalars $\boldsymbol{A}_\ell$, $\ell = 1, ..., L$, which can be of different dimensions, $blkdiag(\boldsymbol{A}_1, ..., \boldsymbol{A}_L)$ denotes the block diagonal matrix with $\boldsymbol{A}_1, ..., \boldsymbol{A}_L$ along the diagonal,

$$\boldsymbol{Q}^i_{\boldsymbol{\beta}} = blkdiag\left(\boldsymbol{Q}^i_{\boldsymbol{\beta}_1}, \ldots, \boldsymbol{Q}^i_{\boldsymbol{\beta}_G}\right) \tag{3.12}$$

with $\boldsymbol{Q}^i_{\boldsymbol{\beta}_g} = \frac{1}{2}\sum\limits_{g^*:g^*\neq g} \mathbf{f}_{i1}(g, g^*)$,

$$\boldsymbol{Q}^i_{\boldsymbol{\sigma^2}} = blkdiag\left(Q^i_{\sigma_1^2}, \ldots, Q^i_{\sigma_G^2}\right), \tag{3.13}$$

Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302

with $Q^i_{\sigma_g^2} = \frac{1}{2} \sum\limits_{g^*:g^* \neq g} f_{i2}(g, g^*)$, and

$$\boldsymbol{Q}^i_{\boldsymbol{\pi}} = blkdiag\left(Q^i_{\pi_1}, \ldots, Q^i_{\pi_{G-1}}\right), \tag{3.14}$$

with, for $1 \leq g \leq G-1$, $Q^i_{\pi_g} = \frac{1}{2}\left(\sum\limits_{g^*:g^* \neq g} \frac{f_{i3}(g,g^*)}{\pi_g^2}\right) + \frac{1}{2}\left(\sum\limits_{g^*:g^* \neq G} \frac{f_{i3}(G,g^*)}{\pi_G^2}\right) + \frac{f_{i3}(g,G)}{\pi_g \pi_G}$. With this notation, the following theorem holds.

Theorem 2. *Assuming that* $y_i \sim \sum_{g=1}^G \pi_g \phi(\mathbf{x}_i^T \beta_g, \sigma_g^2)$, *then, for any* $\boldsymbol{\delta}$, *it holds that* $diag(\sum_{i \in \boldsymbol{\delta}} \mathbf{I}_{M_i}) \leq \sum_{i \in \boldsymbol{\delta}} \mathbf{Q}^i$ *in terms of the Loewner ordering.*

With the help of Theorem 2, we can show that $\sum_{i \in \boldsymbol{\delta}} \boldsymbol{I}_{M_i}$ vanishes under certain conditions for subdata selected by Algorithm 1.

3.3 Main Theorems

Let $\boldsymbol{\mu}_z = (\mu_{z1}, ..., \mu_{zp})^T$ and $\boldsymbol{\Sigma}_z = \boldsymbol{\Phi}_z \boldsymbol{\rho} \boldsymbol{\Phi}_z$ be a full rank covariance matrix, where $\boldsymbol{\Phi}_z = blkdiag(\sigma_{z1}, \ldots, \sigma_{zp})$ is a diagonal matrix of standard deviations and $\boldsymbol{\rho} = (\rho_{jj'})_{p \times p}$ is a correlation matrix.

Theorem 3. *Let* $\mathbf{z}_1, ..., \mathbf{z}_N$ *be iid, where* $\mathbf{z}_i = (z_{i1}, z_{i2}, ..., z_{ip})^T$. *Assuming that* $y_i \sim \sum_{g=1}^G \pi_g \phi(\mathbf{x}_i^T \boldsymbol{\beta}_g, \sigma_g^2)$, *where* $\mathbf{x}_i^T = (1, \mathbf{z}_i^T)^T$, *and* $\boldsymbol{\delta}^*$ *corresponds to subdata selected by Algorithm 1, then the convergence in probability,* $\sum_{i \in \boldsymbol{\delta}^*} \mathbf{I}_{M_i} \xrightarrow{\mathbb{P}} \mathbf{0}_{(Gp+3G-1) \times (Gp+3G-1)}$, *will be achieved when* $N \to \infty$ *under one of the following conditions:*

(a) $\mathbf{z}_i \sim \mathbf{N}(\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$ *and for any triplet* (g, g', j) *with* $g, g' \in \{1, ..., G\}, g \neq g'$ *and* $j \in \{1, ..., p\}$, *it holds that* $\sum\limits_{l=1}^{p} \rho_{lj} \sigma_{zj} (\beta_{g,l} - \beta_{g',l}) \neq 0$;

Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302

(b) $\mathbf{z}_i \sim \mathbf{LN}(\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$ *and for any triplet* (g, g', j) *with* $g, g' \in \{1, ..., G\}, g \neq g'$ *and* $j \in \{1, ..., p\}$*, it holds that* $\beta_{g,j} - \beta_{g',j} \neq 0$ *and* $\sum_{l \in \mathcal{L}_{\min,j}} \left(\beta_{g,l} - \beta_{g',l}\right) \neq 0$*, where* $\mathcal{L}_{min,j} = \left\{l \mid \rho_{lj} = \rho_{\min,j}\ ;\ l = 1, ..., p\right\}$ *and* $\rho_{\min,j} = \min_l \rho_{lj} < 0$*.*

The condition in (a) on the parameter space $\Theta \subset \mathbb{R}^{G(p+3)-1}$ is rather mild. If the condition is not satisfied, the parameter space will be reduced to a lower-dimensional subspace. The condition in (b) is more restrictive due to the requirement $\rho_{\min,j} < 0$, which is needed for technical reasons.

In view of Theorem 3, and guided by the basic strategy formulated at the beginning of this subsection, we propose the following algorithm for fitting a CLR model for a large dataset:

Algorithm 2. With k as the subdata size and p as the number of covariates, assume for simplicity that $r = k/(2p)$ is an integer. Execute the following steps:

1. Run Algorithm 1 to select the subdata $\boldsymbol{\delta}^*$;

2. Using the EM algorithm, fit the CLR model using the subdata selected in Step 1.

While Theorem 3 establishes that the basic strategy works, it sheds no light on the statistical or computational efficiency of Algorithm 2. The next theorem and the empirical results in Sections 4 and 5 show that the statistical efficiency of Algorithm 2 is asymptotically optimal. We will return to the computational efficiency in Section 4.

Theorem 4. *Let $\mathbf{z}_1, ..., \mathbf{z}_N$, where $\mathbf{z}_i = (z_{i1}, z_{i2}, ..., z_{ip})$, be iid and let k be the size of the subdata. Assume that $r = k/(2p)$ is an integer. Let $y_i \sim \sum_{g=1}^{G} \pi_g \phi(\mathbf{x}_i^T \boldsymbol{\beta}_g, \sigma_g^2)$, where $\mathbf{x}_i^T = (1, \mathbf{z}_i^T)^T$, and let $\hat{\boldsymbol{\beta}}_g^{\boldsymbol{\delta}^*}$ be the estimator of $\boldsymbol{\beta}_g$, $g = 1, \ldots, G$, under Algorithm 2.*

(a) If condition (a) in Theorem 3 holds, then, when $N \to \infty$,

$$V(\boldsymbol{A}_N \hat{\boldsymbol{\beta}}_g^{\boldsymbol{\delta}^*}) \to \frac{\sigma_g^2}{\pi_g} \begin{pmatrix} \frac{1}{k} & \mathbf{0} \\ \mathbf{0} & \frac{1}{4r}(\boldsymbol{\Phi}_z \boldsymbol{\rho}^2 \boldsymbol{\Phi}_z)^{-1} \end{pmatrix} \qquad (3.15)$$

where $\boldsymbol{A}_N = blkdiag(1, \sqrt{\log N}, \ldots, \sqrt{\log N})$.

(b) If condition (b) in Theorem 3 holds then, when $N \to \infty$,

$$V(\boldsymbol{B}_N \hat{\boldsymbol{\beta}}_g^{\boldsymbol{\delta}^*}) \to \frac{2\sigma_g^2}{k\pi_g} \begin{pmatrix} 1 & -\boldsymbol{\nu}^T \\ -\boldsymbol{\nu} & p\boldsymbol{\Psi} + \boldsymbol{\nu}\boldsymbol{\nu}^T \end{pmatrix}, \qquad (3.16)$$

where $\boldsymbol{B}_N = blkdiag\Big(1, \exp(\sigma_{z1}\sqrt{2\log N}), ..., \exp(\sigma_{zp}\sqrt{2\log N})\Big)$, $\boldsymbol{\nu} = \Big(e^{-\mu_{z1}}, ..., e^{-\mu_{zp}}\Big)^T$, and $\boldsymbol{\Psi} = blkdiag\Big(e^{-2\mu_{z1}}, ..., e^{-2\mu_{zp}}\Big)$.

In addition, in both cases, the convergence rate for $V(\hat{\boldsymbol{\beta}}_{g,j}^{\boldsymbol{\delta}^*})$, $g = 1, \ldots, G$, is asymptotically optimal.

Remark: Theorem 4 delivers two important messages. First, in terms of statistical efficiency, the convergence rate of the proposed algorithm is asymptotically optimal. Second, it shows that for a fixed subdata size, we retain rich information about the regression parameters in the subdata. These desirable theoretical properties are confirmed by simulation studies in Section 4.

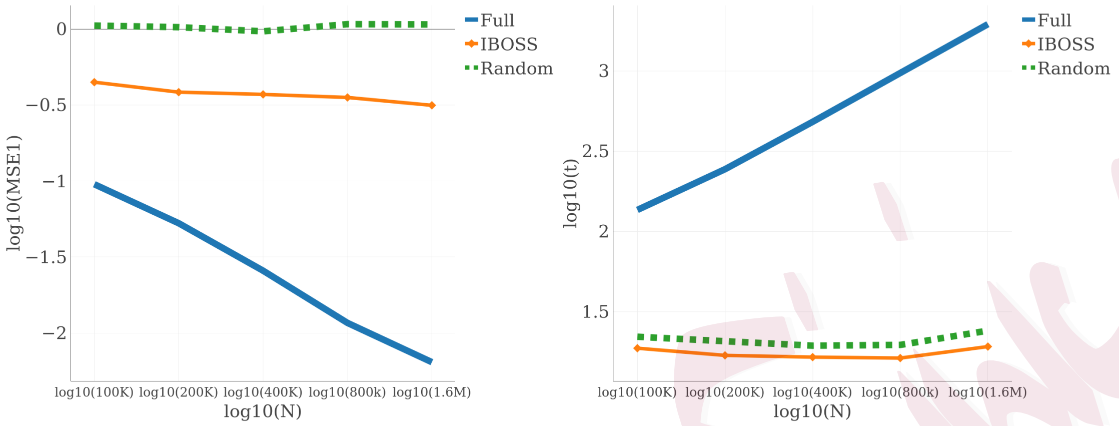
Notice that, while the $\rho_{\min,j} < 0$ condition in (b) is more restrictive due to the technical reasons, the simulation studies in Section 4 indicate the asymptotic results still hold even this condition is not satisfied.

4. Simulation Studies

This section presents simulation studies to evaluate the performance of the proposed algorithm in terms of mean squared error for parameter estimation and computing time. We compare our method to obtaining subdata by random sampling (Random) to analyzing the full data (Full), with the latter serving as a benchmark.

In this simulation, we assume that the number of clusters G is known. The full data of size N is generated from a CLR model with $p = 10$, $G = 5$, and $\pi_1 = 0.1$, $\pi_2 = 0.1$, $\pi_3 = 0.2$, $\pi_4 = 0.3$, and $\pi_5 = 0.3$. We set $\sigma_g = g$ and $\boldsymbol{\beta}_g^T = \left(\beta_{g,0}, \boldsymbol{\beta}_{g,1}^T\right)$ where $\boldsymbol{\beta}_{g,1}^T = \left(g,\ g{+}1, \ldots, g{+}9\right)$ and $\beta_{g,0} = g$ for $g = 1, 2, 3, 4, 5$. For the covariance matrix of the covariates, $\boldsymbol{\Sigma}_z$, we use $\boldsymbol{\Sigma}_{z_{ij}} = 0.5^{\mathbf{1}_{\{i\neq j\}}}$. The covariate vectors $\boldsymbol{z}_i$ are independent and identically distributed as $N(\mathbf{0}, \boldsymbol{\Sigma}_z)$ or $LN(\mathbf{0}, \boldsymbol{\Sigma}_z)$. For each of these, the simulation is repeated 100 times and empirical mean squared errors (MSE) for estimating the intercept and slope parameters are computed as $MSE_{\beta_0} = \frac{1}{100}\sum\limits_{s=1}^{100}\sum\limits_{g=1}^{5}(\hat{\beta}_{g,0}^{(s)} - \beta_{g,0})^2$ and $MSE_{\boldsymbol{\beta}_1} = \frac{1}{100}\sum\limits_{s=1}^{100}\sum\limits_{g=1}^{5}\left|\left|\hat{\boldsymbol{\beta}}_{g,1}^{(s)} - \boldsymbol{\beta}_{g,1}\right|\right|_2^2$, respectively.

For full data sizes $N = 10^5, 2\times 10^5, 4\times 10^5, 8\times 10^5, 1.6\times 10^6$ with fixed subdata size $k = 10000$, Figures 1 and 2 display the comparison of different methods for



(a) MSE

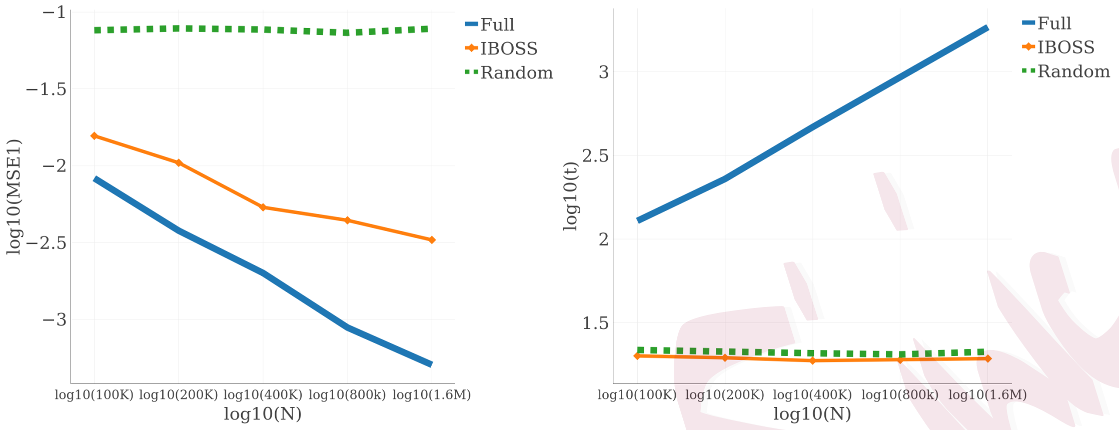
(b) CPU Time (in seconds)

Figure 1: Comparing different methods for estimating slope parameters when covariates are multivariate normal, subdata size $k = 10000$, and full data size N varies

estimating the slope parameters with multivariate normal and lognormal covariate distributions, respectively. In both Figure 1 (a) and Figure 2 (a), it is seen that the MSE for the IBOSS method decreases as the full data size increases. This is consistent with the result of Theorem 4.

Both Figure 1 (b) and Figure 2 (b) show the computing time t (in seconds) for each method across different full data sizes. Computing times were obtained by running Julia 1.8.5 code on an Inspiron 16 plus with 32GB ram and Intel Core i7-12700H. The computing times for FULL increase linearly with the full data sizes on the log-scale. The computing time (including subdata selection and data analysis) for the IBOSS and Random methods are virtually constant across different full

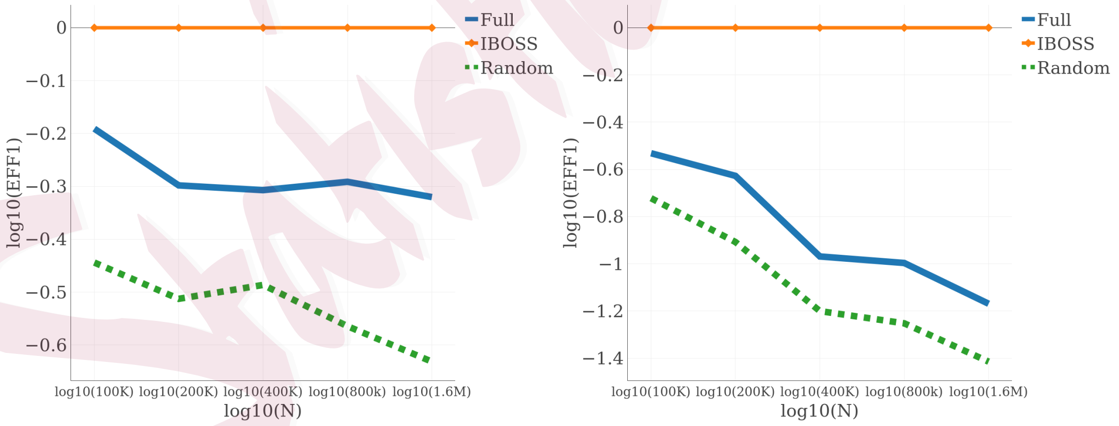
Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302



(a) MSE

(b) CPU Time (in seconds)

Figure 2: Comparing different methods for estimating slope parameters when covariates are multivariate lognormal, subdata size $k = 10000$, and full data size N varies



(a) Normal

(b) LogNormal

Figure 3: Relative Efficiencies of different methods for slope parameters, subdata size $k = 10000$, and full data size N varies

Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302

data sizes. The computing time for IBOSS is even shorter than that for Random, which is due to faster convergence of the EM algorithm with IBOSS subdata than with Random subdata.

To address the trade-off between computing time and statistical efficiency, one could define the relative efficiency for method A compared to IBOSS as

$$Eff_A = \frac{MSE_{IBOSS}/MSE_A}{Time_A/Time_{IBOSS}},$$

where $Time_A$ is the CPU time for method A. If $Eff_A = 0.5$, say, one could think of this as IBOSS only needing half the CPU time of method A to achieve the same MSE, or as IBOSS achieving half the MSE of method A with the same CPU time. Figure 3 presents these relative efficiencies (on a log-scale) for Random and Full for different full data sizes N and subdata size $k = 10000$. Figure 3 shows that the relative efficiencies for Random and Full are smaller if covariates follow the multivariate Lognormal distribution. Also, over the range studied here, the relative efficiencies for Random and Full tend to decrease when the full data size N increases.

5. Application on Structural Protein Data

In this section, we compare the performance of different methods on Structural Protein Data that was originally made available through the Protein Data Bank (PDB)[1]. Biomedical researchers can use the PDB to investigate various illnesses

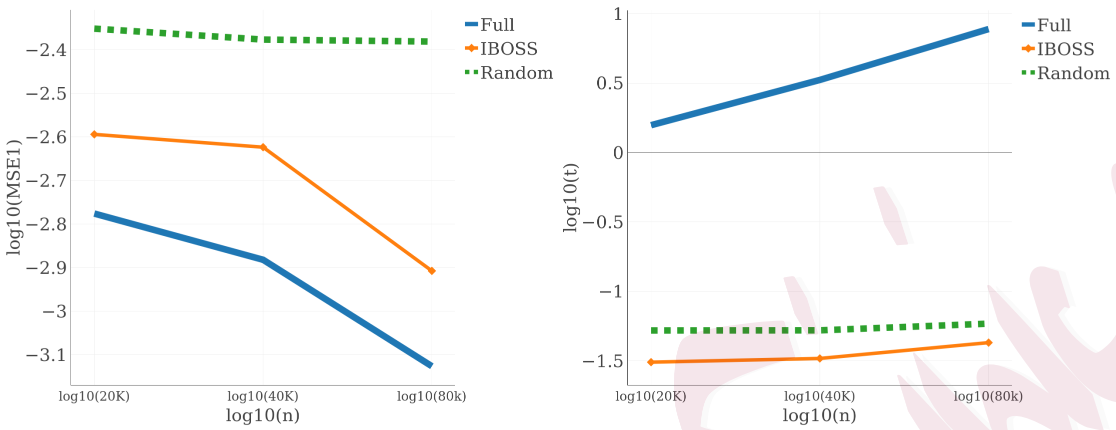
[1] Data is retrieved from https://www.kaggle.com/shahir/protein-data-set

and develop new medicines and solutions that are vital to human existence. In this data set, we analyze the relationships between two variables: the explanatory variable, Structure Molecular Weight, and the response variable, Residue Count. After data cleaning, the full data size is $N = 140,913$. Notice that while the response variable is a count, the response values cover a wide range from 2 to 313,236 with 129,711 unique values.

Considering the choice $G = 3$, the estimated parameters for two of the three clusters exhibit remarkable similarity. This observation strongly suggests that $G = 2$ is a more suitable choice. To compare this method to Random, we compute the MSEs for the slope parameters by using 500 bootstrap samples of size n, using $n = 2 \times 10^4, 4 \times 10^4$, and 8×10^4. The original data is treated as the population of interest. Bootstrap samples then function as samples drawn from that population. This mirrors the relationship between a population distribution and a randomly drawn sample in simulation studies in Section 4. Subdata of size $k = 1000$ is used, both for IBOSS and Random. The MSEs for the slopes are defined as in Section 4 except that we replace $\boldsymbol{\beta}_{g,1}$ by the slope estimates from the full data, $\hat{\boldsymbol{\beta}}_{g,1}^{FULL}$.

Figure 4 (a) shows that IBOSS has a smaller MSE for the estimation of slope parameters than Random. Also, as n increases, the MSE for IBOSS decreases, which is consistent with Theorem 4. For comparing computing time, Figure 4 (b) demonstrates a similar pattern as in the simulation studies. Figure 5 shows that relative efficiencies for Random and Full tend to decrease when n increases, which is also consistent with results in the simulation studies.

Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302



(a) MSE

(b) CPU Time (in seconds)

Figure 4: Comparing different methods for estimating slope parameters based on 500 bootstrap samples of different size n for the Structural Protein Data

6. Conclusions and Future Work

The size of data sets continues to grow, along with increased heterogeneity in data sets. Mixture-of-Experts (MoE) models are powerful and versatile for modeling and understanding heterogeneous data, but fitting them is computationally expensive, especially for large data sets. One efficient strategy to address this issue is the IBOSS strategy proposed by Wang et al. (2019). It not only reduces the computational burden by selecting subdata but also retains high statistical efficiency. This paper developed the IBOSS subdata strategy for Clusterwise Linear Regression (CLR) models, a subclass of the MoE models. We proved that, under relatively mild conditions, the IBOSS subdata selection algorithm proposed by Wang et al. (2019) can be used for Clusterwise Linear Regression Model (CLR)

Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302

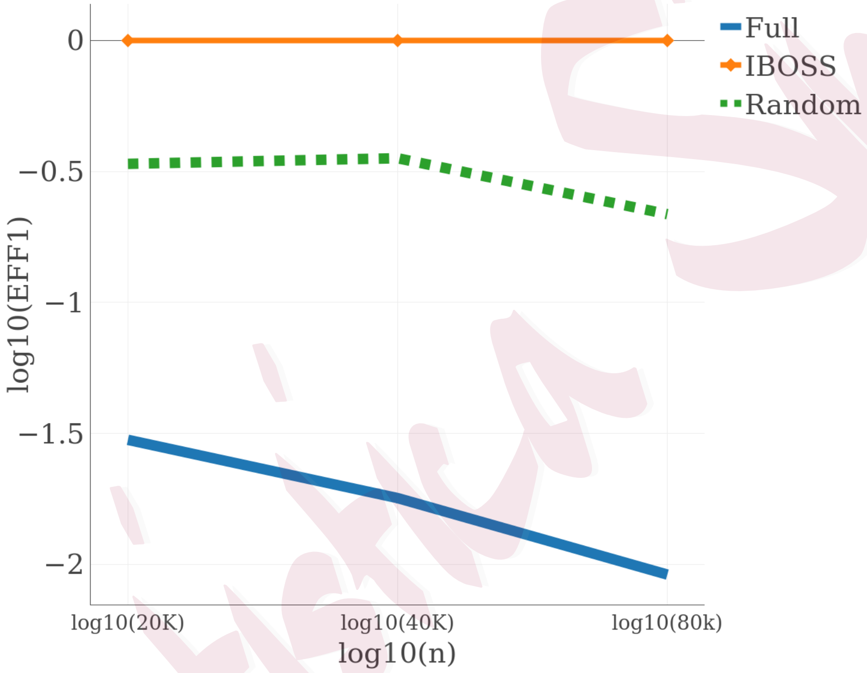


Figure 5: Relative Efficiencies of different methods for slope parameters based on 500 bootstrap samples of different size n for the Structural Protein Data

models. More importantly, we proved that this strategy is asymptotically optimal. The theoretical results are confirmed by simulation studies and a real example.

There remain important unanswered questions that are beyond the scope of this paper and that need more research. First, different clusters may have different support in the covariate space for a general MoE model with gate functions that depend on the covariates. In this case, IBOSS as applied for CLR models may not work well. For example, if there is a cluster in which none of the points have any extreme covariate values, we will completely miss that cluster in the subdata. Deriving an IBOSS strategy for general MoE models will be much harder because the more complicated gate functions make the information matrix even more complicated. The path of finding an appropriate matrix that has a closed-form expression and that bounds the actual information matrix could still work, but how to find an appropriate bounding matrix will need additional research. Second, the model in each cluster can be a generalized linear regression model or other nonlinear model rather than a linear regression model. This too will make the information matrix and developing an IBOSS subdata selection strategy only more complicated.

While we do not have answers to these questions yet, we expect that these can be resolved in the future by methods akin to those used in this paper. Also, the IBOSS strategy is motivated by results in the optimal design of expriments literature, and we believe that the wealth of knowledge and resources in that literature will continue to provide great guidance for developing innovative and superior subdata techniques and algorithms for general MoE models and many other models.

Acknowledgments

The authors gratefully acknowledge the constructive feedback provided by the reviewers on an earlier version of this manuscript. Their insightful comments have been invaluable in enhancing the quality and clarity of our work. John Stufken was supported NSF Grant DMS-23-04767 and Min Yang was supported by NSF Grant DMS-22-10546.

Appendix

A. The Fisher Information Matrix

We start with the first derivatives of the log-likelihood with respect to the parameters:

$$\begin{aligned}
\frac{\partial l_{y_i}}{\partial \boldsymbol{\beta}_g} &= \frac{\pi_g \frac{\partial \phi_{ig}}{\partial \boldsymbol{\beta}_g}}{\sum_{l=1}^G \pi_l \phi_{il}} = w_{ig} \frac{\partial log \phi_{ig}}{\partial \boldsymbol{\beta}_g} \text{ for } g = 1, \ldots, G, \\
\frac{\partial l_{y_i}}{\partial \sigma_g^2} &= \frac{\pi_g \frac{\partial \phi_{ig}}{\partial \sigma_g^2}}{\sum_{l=1}^G \pi_l \phi_{il}} = w_{ig} \frac{\partial log \phi_{ig}}{\partial \sigma_g^2} \text{ for } g = 1, \ldots, G, \text{ and} \\
\frac{\partial l_{y_i}}{\partial \pi_g} &= \frac{\phi_{ig} - \phi_{iG}}{\sum_{l=1}^G \pi_l \phi_{il}} = (\frac{w_{ig}}{\pi_g} - \frac{w_{iG}}{\pi_G}) \text{ for } g = 1, \ldots, G-1.
\end{aligned}$$

This leads to the following expressions for the second derivatives of the log-likelihood with respect to the parameters:

$$\begin{aligned}\frac{\partial^2 l_{y_i}}{\partial\boldsymbol{\beta}_g\partial\boldsymbol{\beta}_g^T} =& \frac{\partial log\phi_{ig}}{\partial\boldsymbol{\beta}_g}\frac{\partial w_{ig}}{\partial\boldsymbol{\beta}_g^T} + w_{ig}\frac{\partial^2 log\phi_{ig}}{\partial\boldsymbol{\beta}_g\partial\boldsymbol{\beta}_g^T}\\ =& w_{ig}(1-w_{ig})\frac{\partial log\phi_{ig}}{\partial\boldsymbol{\beta}_g}\frac{\partial log\phi_{ig}}{\partial\boldsymbol{\beta}_g^T} + w_{ig}\frac{\partial^2 log\phi_{ig}}{\partial\boldsymbol{\beta}_g\partial\boldsymbol{\beta}_g^T},\end{aligned}$$

where $\frac{\partial log\phi_{ig}}{\partial\boldsymbol{\beta}_g} = \frac{(y_i-\mathbf{x_i}^T\boldsymbol{\beta}_g)\mathbf{x_i}}{\sigma_g^2}$ and $\frac{\partial^2 log\phi_{ig}}{\partial\boldsymbol{\beta}_g\partial\boldsymbol{\beta}_g^T} = -\frac{\mathbf{x_i}\mathbf{x_i}^T}{\sigma_g^2}$ for $1 \le g \le G$,

$$\frac{\partial^2 l_{y_i}}{\partial\sigma_g^2\partial\sigma_g^2} = w_{ig}(1-w_{ig})\left[\frac{\partial log\phi_{ig}}{\partial\sigma_g^2}\right]^2 + w_{ig}\cdot\frac{\partial^2 log\phi_{ig}}{\partial(\sigma_g^2)^2},$$

where $\frac{\partial log\phi_{ig}}{\partial\sigma_g^2} = -\frac{1}{2\sigma_g^2} + \frac{(y_i-\mathbf{x_i}^T\boldsymbol{\beta}_g)^2}{2\sigma_g^4}$ and $\frac{\partial^2 log\phi_{ig}}{\partial(\sigma_g^2)^2} = \frac{1}{2\sigma_g^4} - \frac{(y_i-\mathbf{x_i}^T\boldsymbol{\beta}_g)^2}{\sigma_g^6}$ for $1 \le g \le G$,

and

$$\frac{\partial^2 l_{y_i}}{(\partial\pi_g)^2} = -\frac{(\phi_{ig}-\phi_{iG})^2}{(\sum_{l=1}^G \pi_l\phi_{il})^2},$$

for $1 \le g \le G-1$.

The Fisher information matrix is now obtained by taking the negative expectation for all second-order derivatives, leading to the form

$$\boldsymbol{I}(\mathbf{x}_i) = \begin{pmatrix} \boldsymbol{I}_{\boldsymbol{\beta}}(\mathbf{x}_i) & \boldsymbol{I}_{\boldsymbol{\beta},\boldsymbol{\sigma^2}}(\mathbf{x}_i) & \boldsymbol{I}_{\boldsymbol{\beta},\boldsymbol{\pi}}(\mathbf{x}_i) \\ \boldsymbol{I}^T_{\boldsymbol{\beta},\boldsymbol{\sigma^2}}(\mathbf{x}_i) & \boldsymbol{I}_{\boldsymbol{\sigma^2}}(\mathbf{x}_i) & \boldsymbol{I}_{\boldsymbol{\sigma^2},\boldsymbol{\pi}}(\mathbf{x}_i) \\ \boldsymbol{I}^T_{\boldsymbol{\beta},\boldsymbol{\pi}}(\mathbf{x}_i) & \boldsymbol{I}^T_{\boldsymbol{\sigma^2},\boldsymbol{\pi}}(\mathbf{x}_i) & \boldsymbol{I}_{\boldsymbol{\pi}}(\mathbf{x}_i) \end{pmatrix}$$

Furthermore,

$$\boldsymbol{I}_{\boldsymbol{\beta}}(\mathbf{x}_i) = \begin{pmatrix} \boldsymbol{I}_{\boldsymbol{\beta}_1}(\mathbf{x}_i) & \boldsymbol{I}_{\boldsymbol{\beta}_1\boldsymbol{\beta}_2}(\mathbf{x}_i) & \cdots & \boldsymbol{I}_{\boldsymbol{\beta}_1\boldsymbol{\beta}_G}(\mathbf{x}_i) \\ \boldsymbol{I}_{\boldsymbol{\beta}_1\boldsymbol{\beta}_2}(\mathbf{x}_i) & \boldsymbol{I}_{\boldsymbol{\beta}_2}(\mathbf{x}_i) & \cdots & \boldsymbol{I}_{\boldsymbol{\beta}_2\boldsymbol{\beta}_G}(\mathbf{x}_i) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{I}_{\boldsymbol{\beta}_1\boldsymbol{\beta}_G}(\mathbf{x}_i) & \boldsymbol{I}_{\boldsymbol{\beta}_2\boldsymbol{\beta}_G}(\mathbf{x}_i) & \cdots & \boldsymbol{I}_{\boldsymbol{\beta}_G}(\mathbf{x}_i) \end{pmatrix}$$

where

$$\begin{aligned}\boldsymbol{I}_{\boldsymbol{\beta}_g}(\mathbf{x}_i) =& -\mathbb{E}\left(w_{ig}(1-w_{ig})\frac{\partial log\phi_{ig}}{\partial\boldsymbol{\beta}_g}\frac{\partial log\phi_{ig}}{\partial\boldsymbol{\beta}_g^T}+w_{ig}\frac{\partial^2 log\phi_{ig}}{\partial\boldsymbol{\beta}_g\partial\boldsymbol{\beta}_g^T}\right)\\ =&\pi_g\frac{\mathbf{x}_i\mathbf{x}_i^T}{\sigma_g^2}-\mathbb{E}\left(w_{ig}(1-w_{ig})\frac{(y_i-\mathbf{x_i}^T\boldsymbol{\beta}_g)^2\mathbf{x_i}\mathbf{x_i}^T}{\sigma_g^4}\right)\end{aligned} \tag{A.17}$$

for $g = 1, ..., G$;

$$\boldsymbol{I}_{\boldsymbol{\sigma^2}}(\mathbf{x}_i) = \begin{pmatrix} \boldsymbol{I}_{\sigma_1^2}(\mathbf{x}_i) & \boldsymbol{I}_{\sigma_1^2\sigma_2^2}(\mathbf{x}_i) & \cdots & \boldsymbol{I}_{\sigma_1^2\sigma_G^2}(\mathbf{x}_i) \\ \boldsymbol{I}_{\sigma_1^2\sigma_2^2}(\mathbf{x}_i) & \boldsymbol{I}_{\sigma_2^2}(\mathbf{x}_i) & \cdots & \boldsymbol{I}_{\sigma_2^2\sigma_G^2}(\mathbf{x}_i) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{I}_{\sigma_1^2\sigma_G^2}(\mathbf{x}_i) & \boldsymbol{I}_{\sigma_2^2\sigma_G^2}(\mathbf{x}_i) & \cdots & \boldsymbol{I}_{\sigma_G^2}(\mathbf{x}_i) \end{pmatrix}$$

where

$$\begin{aligned}\boldsymbol{I}_{\sigma_g^2}(\mathbf{x}_i) =& -\mathbb{E}\left\{w_{ig}(1-w_{ig})\left[\frac{\partial log\phi_{ig}}{\partial\sigma_g^2}\right]^2+w_{ig}\frac{\partial^2 log\phi_{ig}}{\partial(\sigma_g^2)^2}\right\}\\ =&\mathbb{E}\left\{w_{ig}\left[\frac{(y_i-\mathbf{x_i}^T\boldsymbol{\beta}_g)^2}{\sigma_g^6}-\frac{1}{2\sigma_g^4}\right]\right\}-\mathbb{E}\left\{w_{ig}(1-w_{ig})\left[\frac{\partial log\phi_{ig}}{\partial\sigma_g^2}\right]^2\right\}\\ =&\int_{\mathbb{R}}\frac{\pi_g\phi_{ig}}{\sum_{l=1}^G\pi_l\phi_{il}}\left[\frac{(y_i-\mathbf{x_i}^T\boldsymbol{\beta}_g)^2}{\sigma_g^6}-\frac{1}{2\sigma_g^4}\right](\sum_{l=1}^G\pi_l\phi_{il})dy_i-\mathbb{E}\left\{w_{ig}(1-w_{ig})\left[\frac{\partial log\phi_{ig}}{\partial\sigma_g^2}\right]^2\right\}\\ =&\int_{\mathbb{R}}\pi_g\phi_{ig}\left[\frac{(y_i-\mathbf{x_i}^T\boldsymbol{\beta}_g)^2}{\sigma_g^6}-\frac{1}{2\sigma_g^4}\right]dy_i-\mathbb{E}\left\{w_{ig}(1-w_{ig})\left[\frac{\partial log\phi_{ig}}{\partial\sigma_g^2}\right]^2\right\}\\ =&\frac{\pi_g}{2\sigma_g^4}-\mathbb{E}\left\{w_{ig}(1-w_{ig})\left[-\frac{1}{2\sigma_g^2}+\frac{(y_i-\mathbf{x_i}^T\boldsymbol{\beta}_g)^2}{2\sigma_g^4}\right]^2\right\}\end{aligned} \tag{A.18}$$

for $g = 1, ..., G$; and

$$\boldsymbol{I}_{\boldsymbol{\pi}}(\mathbf{x}_i) = \begin{pmatrix} \boldsymbol{I}_{\pi_1}(\mathbf{x}_i) & \boldsymbol{I}_{\pi_1\pi_2}(\mathbf{x}_i) & \cdots & \boldsymbol{I}_{\pi_1\pi_{G-1}}(\mathbf{x}_i) \\ \boldsymbol{I}_{\pi_1\pi_2}(\mathbf{x}_i) & \boldsymbol{I}_{\pi_2}(\mathbf{x}_i) & \cdots & \boldsymbol{I}_{\pi_2\pi_{G-1}}(\mathbf{x}_i) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{I}_{\pi_1\pi_{G-1}}(\mathbf{x}_i) & \boldsymbol{I}_{\pi_2\pi_{G-1}}(\mathbf{x}_i) & \cdots & \boldsymbol{I}_{\pi_{G-1}}(\mathbf{x}_i) \end{pmatrix}$$

where

$$\begin{aligned}\boldsymbol{I}_{\pi_g}(\mathbf{x}_i) =& -\mathbb{E}\Bigg\{-\frac{(\phi_{ig}-\phi_{iG})^2}{(\sum_{g=1}^{G}\pi_g\phi_{ig})^2}\Bigg\}\\ =&\mathbb{E}\Bigg\{\frac{w_{ig}^2}{\pi_g^2}+\frac{w_{iG}^2}{\pi_G^2}-2\frac{w_{ig}w_{iG}}{\pi_g\pi_G}\Bigg\}\\ =&\frac{1}{\pi_g}+\frac{1}{\pi_G}-\mathbb{E}\Bigg\{\frac{w_{ig}(1-w_{ig})}{\pi_g^2}+\frac{w_{iG}(1-w_{iG})}{\pi_G^2}+2\frac{w_{ig}w_{iG}}{\pi_g\pi_G}\Bigg\}\end{aligned}\tag{A.19}$$

for $g=1,...,G-1$.

B. The proofs of main results

Before we present a proof of Theorem 2, we need the following lemma.

Lemma 1. *Assuming $y_i \sim \sum_{g=1}^{G}\pi_g\phi(\mathbf{x}_i^T\beta_g,\sigma_g^2)$, then the following inequalities hold for any $1\le g_1,g_2\le G$, $g_1\ne g_2$:*

$$\begin{aligned}&diag\left(\mathbb{E}\mathbf{x}_i\mathbf{x}_i^T w_{ig_1}w_{ig_2}\Delta^2_{\boldsymbol{\beta}_{ig_1}}\right)\le\frac{1}{2}\mathbf{f}_{i1}(g_1,g_2),\\ &\mathbb{E}\left(w_{ig_1}w_{ig_2}\Delta^2_{\boldsymbol{\sigma}_{ig_1}}\right)\le\frac{1}{2}f_{i2}(g_1,g_2),\\ &\mathbb{E}\left(w_{ig_1}w_{ig_2}\right)\le\frac{1}{2}f_{i3}(g_1,g_2).\end{aligned}\tag{B.20}$$

Here the first inequality is under the Loewner ordering.

Proof. Since the proofs of all inequalities are similar, we only provide the proof for the first inequality.

$$\begin{aligned}& diag\left(\mathbb{E}\left(\mathbf{x}_i\mathbf{x}_i^T w_{ig_1}w_{ig_2}\Delta^2_{\boldsymbol{\beta}_{ig_1}}\right)\right)\\ =\;& diag\left(\mathbf{x}_i\mathbf{x}_i^T\int\frac{\pi_{g_1}\phi_{ig_1}\pi_{g_2}\phi_{ig_2}}{\sum_{g=1}^{G}\pi_g\phi_{ig}}\Delta^2_{\boldsymbol{\beta}_{ig_1}}dy_i\right)\\ \le\;& diag\left(\mathbf{x}_i\mathbf{x}_i^T\int\frac{\pi_{g_1}\phi_{ig_1}\pi_{g_2}\phi_{ig_2}}{\pi_{g_1}\phi_{ig_1}+\pi_{g_2}\phi_{ig_2}}\Delta^2_{\boldsymbol{\beta}_{ig_1}}dy_i\right)\\ \le\;& diag\left(\mathbf{x}_i\mathbf{x}_i^T\int\tfrac{1}{2}\sqrt{\pi_{g_1}\phi_{ig_1}\pi_{g_2}\phi_{ig_2}}\Delta^2_{\boldsymbol{\beta}_{ig_1}}dy_i\right)=\tfrac{1}{2}\mathbf{f}_{i1}(g_1,g_2).\end{aligned}\tag{B.21}$$

□

Now we are ready to prove Theorem 2.

Proof of Theorem 2. By (A.17) and the definition of $\Delta_{\boldsymbol{\beta}_{ig}}$, we have

$$diag\left(\boldsymbol{I}_{\boldsymbol{\beta}_g|M_i}\right) = diag\left(\mathbf{x_i}\mathbf{x_i}^T \sum_{g^*:g^*\neq g} \mathbb{E}w_{ig}w_{ig^*}\Delta^2_{\boldsymbol{\beta}_{ig}}\right).$$

Similarly, by (A.18) and the definition of $\Delta_{\boldsymbol{\sigma}_{ig}}$, we have

$$\boldsymbol{I}_{\boldsymbol{\sigma}^2_g|M_i} = \sum_{g^*:g^*\neq g} \mathbb{E}w_{ig}w_{ig^*}\Delta^2_{\boldsymbol{\sigma}_{ig}}$$

and by (A.19), we have

$$\boldsymbol{I}_{\boldsymbol{\pi}_g|M_i} = \mathbb{E}\left(\sum_{g^*:g^*\neq g}\left(\frac{w_{ig}w_{ig^*}}{\pi_g^2} + \frac{w_{iG}w_{ig^*}}{\pi_G^2}\right) + 2\frac{w_{ig}w_{iG}}{\pi_g\pi_G}\right).$$

By Lemma 1 and the definition of $\boldsymbol{Q}^i$, the conclusion follows. □

Proof of Theorem 3. By Theorem 2, the result follows if we show that $\sum_{i\in\boldsymbol{\delta}^*}\boldsymbol{Q}^i \xrightarrow{\mathbb{P}} \mathbf{0}_{(Gp+3G-1)\times(Gp+3G-1)}$. This follows if, for all $i\in\boldsymbol{\delta}^*$ and $g\neq g'$,

$$\begin{aligned} \mathbf{f}_{i1}(g,g') &\xrightarrow{\mathbb{P}} \mathbf{0}_{(p+1)\times(p+1)}, \\ f_{i2}(g,g') &\xrightarrow{\mathbb{P}} 0, \text{ and} \qquad \text{(B.22)} \\ f_{i3}(g,g') &\xrightarrow{\mathbb{P}}, 0 \end{aligned}$$

where $\mathbf{f}_{i1}$, f_{i2} and f_{i3} are defined in (3.8) - (3.10). We prove the two cases separately.

Case (a):

For any covariate, Algorithm I is guaranteed to select r data points with the r largest values for the covariate in the full data and r data points with the r smallest values of the covariate in the full data. However, when selecting data points based on covariate l, $l \geq 2$, some or all of the data points with the r largest and r smallest values for the lth covariate may already have been selected. So, Algorithm I may select data points in which none of the values are among the r largest or r smallest values for any covariate. However, what we can guarantee for the subdata $\boldsymbol{\delta}^*$ selected by Algorithm I is the following. For any $i \in \boldsymbol{\delta}^*$, there exists a $j_i \in \{1, ..., p\}$ and $m_i \in \{1, ..rp, N-rp+1, ..., N\}$ so that $\mathbf{x}_i = (1, z_{j_i}^{(m_i)1}, ..., z_{(m_i)j_i}, ..., z_{j_i}^{(m_i)p})$, where $z_{(m_i)j}$ is the m_i^{th} order statistic of $\{z_{1j}, ..., z_{Nj}\}$ and $z_j^{(m_i)l}$ is the concomitant of $z_{(m_i)j}$ for the lth covariate, $l \neq j$. Without loss of generality, let $j_i = 1$. For $i = 1, ..., N$ and $g = 1, ..., G$, define $\gamma_{ig} = \mathbf{x}_i^T \beta_g$. Then we have

$$
\begin{aligned}
\mathbf{f}_{i1}(g,g') &= diag\left(\mathbf{x}_i\mathbf{x}_i^T \int \tilde{w}_i(g,g')\Delta^2_{\boldsymbol{\beta}_{ig}} dy_i\right) \\
&= diag\left(\mathbf{x}_i\mathbf{x}_i^T\right) \int_{\mathbb{R}} \sqrt{\pi_g\pi_{g'}} \frac{(y_i-\gamma_{ig})^2}{\sigma_g^4} \frac{1}{\sqrt{2\pi\sigma_g\sigma_{g'}}} \exp\left\{ - \frac{(y_i-\gamma_{ig})^2}{4\sigma_g^2} - \frac{(y_i-\gamma_{ig'})^2}{4\sigma_{g'}^2} \right\} dy_i \\
&= diag\left(\mathbf{x}_i\mathbf{x}_i^T\right) \int_{\mathbb{R}} \sqrt{\pi_g\pi_{g'}} \frac{(y_i-\gamma_{ig})^2}{\sigma_g^4} \frac{1}{\sqrt{2\pi\sigma_g\sigma_{g'}}} \exp\left\{ - \frac{y_i^2 - 2\frac{\sigma_{g'}^2\gamma_{ig}+\sigma_g^2\gamma_{ig'}}{\sigma_g^2+\sigma_{g'}^2} y_i + \frac{\gamma_{ig}^2\sigma_{g'}^2+\gamma_{ig'}^2\sigma_g^2}{\sigma_g^2+\sigma_{g'}^2}}{2\cdot\frac{2\sigma_g^2\sigma_{g'}^2}{\sigma_g^2+\sigma_{g'}^2}} \right\} dy_i \\
&= diag\left(\mathbf{x}_i\mathbf{x}_i^T\right) \int_{\mathbb{R}} \sqrt{\pi_g\pi_{g'}} \sqrt{\frac{2\sigma_g\sigma_{g'}}{\sigma_g^2+\sigma_{g'}^2}} \frac{(y_i-\gamma_{ig})^2}{\sigma_g^4} \phi\left(\frac{\sigma_{g'}^2\gamma_{ig}+\sigma_g^2\gamma_{ig'}}{\sigma_g^2+\sigma_{g'}^2}, \frac{2\sigma_g^2\sigma_{g'}^2}{\sigma_g^2+\sigma_{g'}^2} \right) \exp\left\{ - \frac{(\gamma_{ig}-\gamma_{ig'})^2}{4(\sigma_g^2+\sigma_{g'}^2)} \right\} dy_i \\
&= diag\left(\mathbf{x}_i\mathbf{x}_i^T\right) \sqrt{\frac{2\pi_g\pi_{g'}\sigma_g\sigma_{g'}}{\sigma_g^2+\sigma_{g'}^2}} \left[\frac{2\sigma_{g'}^2/\sigma_g^2}{\sigma_g^2+\sigma_{g'}^2} + \frac{(\mathbf{x}_i^T\boldsymbol{\beta}_g - \mathbf{x}_i^T\boldsymbol{\beta}_{g'})^2}{(\sigma_g^2+\sigma_{g'}^2)^2} \right] \exp\left\{ - \frac{(\mathbf{x}_i^T\boldsymbol{\beta}_g - \mathbf{x}_i^T\boldsymbol{\beta}_{g'})^2}{4(\sigma_g^2+\sigma_{g'}^2)} \right\} \\
&= diag\left(\mathbf{x}_i\mathbf{x}_i^T\right) \sqrt{\frac{2\pi_g\pi_{g'}\sigma_g\sigma_{g'}}{\sigma_g^2+\sigma_{g'}^2}} \left[\frac{2\sigma_{g'}^2/\sigma_g^2}{\sigma_g^2+\sigma_{g'}^2} + \frac{\left(\beta_{g,0}-\beta_{g',0}+z_{(m_i)1}(\beta_{g,1}-\beta_{g',1})+\sum_{l=2}^p z_1^{(m_i)l}(\beta_{g,l}-\beta_{g',l})\right)^2}{(\sigma_g^2+\sigma_{g'}^2)^2} \right] \times \\
&\qquad \exp\left\{ - \frac{\left(\beta_{g,0}-\beta_{g',0}+z_{(m_i)1}(\beta_{g,1}-\beta_{g',1})+\sum_{l=2}^p z_1^{(m_i)l}(\beta_{g,l}-\beta_{g',l})\right)^2}{4(\sigma_g^2+\sigma_{g'}^2)} \right\} \qquad \text{(B.23)}
\end{aligned}
$$

where $\mathbf{x}_i^T = (1, z_{(m_i)1}, z_1^{(m_i)2} ..., z_1^{(m_i)p})$. From the results in Examples 2.8.1 and 5.5.1

of Galambos (1987), when $(z_{i1}, ..., z_{ip}) \sim \boldsymbol{N}(\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$

$$z_{(m_i)1} = \mu_{z1} - \sigma_{z1}\sqrt{2logN} + O_P(1), \qquad m_i = 1, ...rp, \tag{B.24}$$

$$z_{(m_i)1} = \mu_{z1} + \sigma_{z1}\sqrt{2logN} + O_P(1), \qquad m_i = N - rp + 1, ..., N, \tag{B.25}$$

$$z_1^{(m_i)l} = \mu_{z1} - \rho_{l1}\sigma_{z1}\sqrt{2logN} + O_P(1), \qquad m_i = 1, ...rp, \tag{B.26}$$

$$z_1^{(m_i)l} = \mu_{z1} + \rho_{l1}\sigma_{z1}\sqrt{2logN} + O_P(1), \qquad m_i = N - rp + 1, ..., N. \tag{B.27}$$

We distinguish between $m_i \in \{1, ..., rp\}$ and $m_i \in \{N - rp + 1, ..., N\}$. First, for $m_i \in \{1, ...rp\}$, by (B.24) and (B.26) we have $\mathbf{x}_i^T = \Big(1, -\rho_{11}\sigma_{z1}\sqrt{2logN} + O_p(1), \ldots, -\rho_{p1}\sigma_{z1}\sqrt{2logN} + O_p(1)\Big)$, so that (B.23) can be written as

$$diag\left(\mathbf{x}_i\mathbf{x}_i^T\right)\sqrt{\tfrac{2\pi_g\pi_{g'}\sigma_g\sigma_{g'}}{\sigma_g^2+\sigma_{g'}^2}}\left[\tfrac{2\sigma_{g'}^2/\sigma_g^2}{\sigma_g^2+\sigma_{g'}^2} + \tfrac{\left(-\sqrt{2logN}\sum\limits_{l=1}^{p}\rho_{l1}\sigma_{z1}(\beta_{g,l}-\beta_{g',l})+O_p(1)\right)^2}{(\sigma_g^2+\sigma_{g'}^2)^2}\right] \times$$

$$\exp\left\{-\tfrac{\left(-\sqrt{2logN}\sum\limits_{l=1}^{p}\rho_{l1}\sigma_{z1}(\beta_{g,l}-\beta_{g',l})+O_p(1)\right)^2}{4(\sigma_g^2+\sigma_{g'}^2)}\right\}. \tag{B.28}$$

Second, for $m_i \in \{N - rp + 1, ..., N\}$, by (B.25) and (B.27) we have $\mathbf{x}_i = \Big(1, \rho_{11}\sigma_{z1}\sqrt{2logN} + O_p(1), \ldots, \rho_{p1}\sigma_{z1}\sqrt{2logN} + O_p(1)\Big)$, so that (B.23) can be written as

$$diag\left(\mathbf{x}_i\mathbf{x}_i^T\right)\sqrt{\tfrac{2\pi_g\pi_{g'}\sigma_g\sigma_{g'}}{\sigma_g^2+\sigma_{g'}^2}}\left[\tfrac{2\sigma_{g'}^2/\sigma_g^2}{\sigma_g^2+\sigma_{g'}^2} + \tfrac{\left(\sqrt{2logN}\sum\limits_{l=1}^{p}\rho_{l1}\sigma_{z1}(\beta_{g,l}-\beta_{g',l})+O_p(1)\right)^2}{(\sigma_g^2+\sigma_{g'}^2)^2}\right] \times \tag{B.29}$$

$$\exp\left\{-\tfrac{\left(\sqrt{2logN}\sum\limits_{l=1}^{p}\rho_{l1}\sigma_{z1}(\beta_{g,l}-\beta_{g',l})+O_p(1)\right)^2}{4(\sigma_g^2+\sigma_{g'}^2)}\right\}. \tag{B.30}$$

With the condition for Case (a), $\sum\limits_{l=1}^{p}\rho_{l1}\sigma_{z1}(\beta_{g,l} - \beta_{g',l}) \neq 0$, this implies that when $N \to \infty$, (B.28) $\xrightarrow{\mathbb{P}} \mathbf{0}_{(p+1)\times(p+1)}$ and (B.30) $\xrightarrow{\mathbb{P}} \mathbf{0}_{(p+1)\times(p+1)}$. Consequently $\mathbf{f}_{i1}(g, g') \xrightarrow{\mathbb{P}} \mathbf{0}_{(p+1)\times(p+1)}$.

Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302

Case (b): By the same argument as in the proof of Case (a), it suffices to show that, for all $i \in \boldsymbol{\delta}^*$,

$$\begin{aligned} &\mathbf{f}_{i1}\big(g, g'\big) \xrightarrow{\mathbb{P}} \mathbf{0}_{(p+1)\times(p+1)} \\ &f_{i2}(g, g') \xrightarrow{\mathbb{P}} 0 \qquad \text{(B.31)} \\ &f_{i3}(g, g') \xrightarrow{\mathbb{P}} 0 \end{aligned}$$

for any pair (g, g'). Since proofs of the three convergences are similar, we only show a proof of the first one and use the same notation as in the proof for part (a) of Theorem 3. Without loss of generality, set $j_i = 1$. By the same argument as used in (B.23), we have

$$\begin{aligned} \mathbf{f}_{i1}(g, g') = \; & diag\left(\mathbf{x}_i\mathbf{x}_i^T\right)\sqrt{\tfrac{2\pi_g\pi_{g'}\sigma_g\sigma_{g'}}{\sigma_g^2+\sigma_{g'}^2}}\left[\tfrac{2\sigma_{g'}^2/\sigma_g^2}{\sigma_g^2+\sigma_{g'}^2} + \tfrac{\left(\beta_{g,0}-\beta_{g',0}+z_{(m_i)1}(\beta_{g,1}-\beta_{g',1})+\sum_{l=2}^p z_1^{(m_i)l}(\beta_{g,l}-\beta_{g',l})\right)^2}{(\sigma_g^2+\sigma_{g'}^2)^2}\right] \times \\ & \exp\left\{-\tfrac{\left(\beta_{g,0}-\beta_{g',0}+z_{(m_i)1}(\beta_{g,1}-\beta_{g',1})+\sum_{l=2}^p z_1^{(m_i)l}(\beta_{g,l}-\beta_{g',l})\right)^2}{4(\sigma_g^2+\sigma_{g'}^2)}\right\}, \qquad \text{(B.32)} \end{aligned}$$

where $\mathbf{x}_i^T = (1, z_{(m_i)1}, z_1^{(m_i)2} ..., z_1^{(m_i)p})$. From the results in Theorem 6 of Wang et al. (2019), when $(z_{i1}, ..., z_{ip}) \sim LN(\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$,

$$z_{(m_i)1} = exp\big(-\sigma_{z1}\sqrt{2logN}\big)O_P(1), \qquad m_i \in \{1, ..., rp\}; \qquad \text{(B.33)}$$

$$z_{(m_i)1} = exp\big(\sigma_{z1}\sqrt{2logN}\big)O_P(1), \qquad m_i \in \{N - rp + 1, ..., N\}; \text{(B.34)}$$

$$z_1^{(m_i)l} = exp\big(-\rho_{l1}\sigma_{z1}\sqrt{2logN}\big)O_P(1), \qquad m_i \in \{1, ..., rp\}; \qquad \text{(B.35)}$$

$$z_1^{(m_i)l} = exp\big(\rho_{l1}\sigma_{z1}\sqrt{2logN}\big)O_P(1), \qquad m_i \in \{N - rp + 1, ..., N\}. \text{(B.36)}$$

As in the proof for Case (a), we consider the cases $m_i \in \{1, ..., rp\}$ and $m_i \in \{N - rp + 1, ..., N\}$. First, for $m_i \in \{1, ..., rp\}$, by (B.33) and (B.35), (B.32) can be written as

$$diag\left(\mathbf{x}_i\mathbf{x}_i^T\right)\sqrt{\frac{2\pi_g\pi_{g'}\sigma_g\sigma_{g'}}{\sigma_g^2+\sigma_{g'}^2}}\left[\frac{2\sigma_{g'}^2/\sigma_g^2}{\sigma_g^2+\sigma_{g'}^2}+\frac{A_1^2}{(\sigma_g^2+\sigma_{g'}^2)^2}\right]\times\exp\left\{-\frac{A_1^2}{4(\sigma_g^2+\sigma_{g'}^2)}\right\}, \qquad \text{(B.37)}$$

where

$$\begin{aligned}\mathbf{x}_i =& \Big(1, \quad exp\big\{-\rho_{11}\sigma_{z1}\sqrt{2logN}\big\}O_P(1) \quad ,\ldots, \quad exp\big\{-\rho_{p1}\sigma_{z1}\sqrt{2logN}\big\}O_P(1)\Big) \text{ and}\\ A_1 =& \qquad \beta_{g,0}-\beta_{g',0}+O_P(1)\Bigg[exp\Big\{-\rho_{\min,1}\sigma_{z1}\sqrt{2logN}\Big\}\times\sum_{l\in\mathcal{L}_{\min,1}}\left(\beta_{g,l}-\beta_{g',l}\right)+\\ &\sum_{l\notin\mathcal{L}_{\min,1}}\left(exp\Big\{-\rho_{lj}\sigma_{z1}\sqrt{2logN}\Big\}(\beta_{g,l}-\beta_{g',l})\right)\Bigg].\end{aligned}$$

With the condition on the parameters for Case (b), we have that $\rho_{\min,j}<0$ and $\sum_{l\in\mathcal{L}_{\min,j}}\left(\beta_{g,l}-\beta_{g',l}\right)\neq 0$. Thus (B.37) $\xrightarrow{\mathbb{P}}\mathbf{0}_{(p+1)\times(p+1)}$ when $N\to\infty$.

Second, for $m_i\in\{N-rp+1,...,N\}$, by (B.34) and (B.36), (B.32) can be written as

$$diag\left(\mathbf{x}_i\mathbf{x}_i^T\right)\sqrt{\frac{2\pi_g\pi_{g'}\sigma_g\sigma_{g'}}{\sigma_g^2+\sigma_{g'}^2}}\left[\frac{2\sigma_{g'}^2/\sigma_g^2}{\sigma_g^2+\sigma_{g'}^2}+\frac{A_2^2}{(\sigma_g^2+\sigma_{g'}^2)^2}\right]\times\exp\left\{-\frac{A_2^2}{4(\sigma_g^2+\sigma_{g'}^2)}\right\} \qquad \text{(B.38)}$$

where

$$\begin{aligned}\mathbf{x}_i =& \Big(1, \quad exp\big\{\rho_{11}\sigma_{z1}\sqrt{2logN}\big\}O_P(1) \quad ,\ldots, \quad exp\big\{\rho_{p1}\sigma_{z1}\sqrt{2logN}\big\}O_P(1)\Big) \text{ and}\\ A_2 =& \qquad \beta_{g,0}-\beta_{g',0}+O_P(1)\Bigg[exp\Big\{\sigma_{z1}\sqrt{2logN}\Big\}\times\left(\beta_{g,1}-\beta_{g',1}\right)+\\ &\sum_{l>1}\left(exp\Big\{\rho_{l1}\sigma_{z1}\sqrt{2logN}\Big\}(\beta_{g,l}-\beta_{g',l})\right)\Bigg]\end{aligned}$$

With the condition on the parameters for Case (b), we have $\beta_{g,1}-\beta_{g',1}\neq 0$. Thus (B.38) $\xrightarrow{\mathbb{P}}\mathbf{0}_{(p+1)\times(p+1)}$ when $N\to\infty$. Thus the conclusion follows.

□

Proof of Theorem 4. For Case (a), by Theorem 6 in Wang et al. (2019), when $\mathbf{z}_i \sim N(\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$,

$$\sum_{i\in\boldsymbol{\delta}^*} \mathbf{x}_i\mathbf{x}_i^T = \begin{pmatrix} k & \mathbf{0} \\ \mathbf{0} & 4r\log N \boldsymbol{\Phi}_z \boldsymbol{\rho}^2 \boldsymbol{\Phi}_z \end{pmatrix} + O_P(\sqrt{\log N}) \tag{B.39}$$

and

$$\boldsymbol{A}_N \left(\sum_{i\in\boldsymbol{\delta}^*} \mathbf{x}_i\mathbf{x}_i^T\right)^{-1} \boldsymbol{A}_N = \begin{pmatrix} \frac{1}{k} & \mathbf{0} \\ \mathbf{0} & \frac{1}{4r}(\boldsymbol{\Phi}_z \boldsymbol{\rho}^2 \boldsymbol{\Phi}_z)^{-1} \end{pmatrix} + O_P\Big(\frac{1}{(\sqrt{\log N}}\Big). \tag{B.40}$$

Notice that $\boldsymbol{I}(\boldsymbol{\delta}^*) = \sum_{i\in\boldsymbol{\delta}^*} \boldsymbol{I}_{C_i} - \sum_{i\in\boldsymbol{\delta}^*} \boldsymbol{I}_{M_i}$. By Theorems 2 and 3, we have $\sum_{i\in\boldsymbol{\delta}^*} \boldsymbol{I}_{M_i} \xrightarrow{\mathbb{P}} \mathbf{0}_{(Gp+3G-1)\times(Gp+3G-1)}$ when $N \to \infty$, which implies that $\boldsymbol{I}(\boldsymbol{\delta}^*) \xrightarrow{\mathbb{P}} \sum_{i\in\boldsymbol{\delta}^*} \boldsymbol{I}_{C_i}$ when $N \to \infty$. By the expressions for $\boldsymbol{I}_{C_i}$ and $\boldsymbol{I}_{\boldsymbol{\beta}|C_i}$ in (3.2) and (3.3), respectively, the desired conclusion follows from (B.40).

For Case (b), also by Theorem 6 in Wang et al. (2019), when $\mathbf{z}_i \sim LN(\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$,

$$\sum_{i\in\boldsymbol{\delta}^*} \mathbf{x}_i\mathbf{x}_i^T = \begin{pmatrix} k & \mathbf{v}^T \\ \mathbf{v} & \boldsymbol{\Omega}, \end{pmatrix} \tag{B.41}$$

where, with $\mathbf{v}^T = (v_1, \ldots, v_p)$ and $\boldsymbol{\Omega} = (\Omega_{j_1j_2})_{p\times p}$,

$$\Omega_{jj} = r\exp\left(2\sigma_{zj}\sqrt{2\log N}\right)\left\{e^{2\mu_{zj}} + o_p(1)\right\},$$
$$\Omega_{j_1j_2} = 2r\exp\left\{(\sigma_{zj_1} + \sigma_{zj_2})\sqrt{2\log N}\right\}o_p(1), \text{ and}$$
$$v_j = r\exp\left(\sigma_{zj}\sqrt{2\log N}\right)\left\{e^{\mu_{zj}} + o_p(1)\right\}$$

and

$$\boldsymbol{B}_N \left(\sum_{i\in\boldsymbol{\delta}^*} \mathbf{x}_i\mathbf{x}_i^T\right)^{-1} \boldsymbol{B}_N = \frac{2}{k}\begin{pmatrix} 1 & -\boldsymbol{\nu}^T \\ -\boldsymbol{\nu} & p\boldsymbol{\Psi} + \boldsymbol{\nu}\boldsymbol{\nu}^T \end{pmatrix} + o_P(1). \tag{B.42}$$

By a similar argument as for Case (a), the desired conclusion follows.

Next we want to show that $\boldsymbol{\delta}^*$ provides the fastest convergence rate for $V(\hat{\beta}_{g,j}^{\boldsymbol{\delta}}) \xrightarrow{\mathbb{P}} 0$ among all subdata $\boldsymbol{\delta}$ of size k. We consider Case (a) only since the proof for Case (b) is similar. From (3.7), for any $\boldsymbol{\delta}$ with subdata size k, we have $\boldsymbol{I}(\boldsymbol{\delta})^{-1} \geq \left(\sum_{i\in\boldsymbol{\delta}} \boldsymbol{I}_{C_i}\right)^{-1}$ in Loewner order, and further we have $\{\boldsymbol{I}(\boldsymbol{\delta})^{-1}\}_{jj} \geq \{\left(\sum_{i\in\boldsymbol{\delta}} \boldsymbol{I}_{C_i}\right)^{-1}\}_{jj} \geq \left(\{\sum_{i\in\boldsymbol{\delta}} \boldsymbol{I}_{C_i}\}_{jj}\right)^{-1}$ for all j. Then for estimating the slope parameters of the gth cluster with any subdata $\boldsymbol{\delta}$, we have

$$\begin{aligned} V(\hat{\beta}_{g,j}^{\boldsymbol{\delta}}) &\geq \frac{\sigma_g^2}{\pi_g}(\sum_{i\in\boldsymbol{\delta}} z_{ij}^2)^{-1} \geq \frac{\sigma_g^2}{\pi_g} \min\left((kz_{(1)j}^2)^{-1}, (kz_{(N)j}^2)^{-1}\right) \\ &= \frac{\sigma_g^2}{k\pi_g} \min\left((\mu_{z1} + \sigma_{z1}\sqrt{2logN} + o_P(1))^{-2}, (\mu_{z1} + \sigma_{z1}\sqrt{2logN} + o_P(1))^{-2}\right) \end{aligned} \tag{B.43}$$

for $j = 1, ..., p$. From (B.43), for any $\boldsymbol{\delta}$, the lower bound of the convergence rate of $V(\hat{\beta}_{g,j}^{\boldsymbol{\delta}})$ is $1/\log N$. On the other hand, from (3.15), it is clear $V(\hat{\beta}_{g,j}^{\boldsymbol{\delta}^*})$ achieves this lower bound. □

References

Bagirov, A. M., Mahmood, A. and Barton, A. (2017) Prediction of monthly rainfall in Victoria, Australia: Clusterwise linear regression approach. *Atmospheric Research*, **188**, 20–29.

Balakrishnan, S., Wainwright, M. J. and Yu, B. (2017) Statistical guarantees for the EM algorithm: From population to sample-based analysis. *The Annals of Statistics*, **45**, 77 – 120.

Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302

Brusco, M. J., Cradit, J. D. and Tashchian, A. (2003) Multicriterion Clusterwise Regression for Joint Segmentation Settings: An Application to Customer Value. *Journal of Marketing Research*, **40**, 225–234.

Bühlmann, P., Drineas, P., Kane, M. and Laan, M. v. d. (2016) *Handbook of Big Data*. Chapman and Hall/CRC.

Cai, L. and Zhu, Y. (2015) The Challenges of Data Quality and Data Quality Assessment in the Big Data Era. *Data Science Journal*, **14**, 2.

Chang, M.-C. (2023) Predictive Subdata Selection for Computer Models. *Journal of Computational and Graphical Statistics*, **32**, 613–630.

Chang, M.-C. (2024) Supervised Stratified Subsampling for Predictive Analytics. *Journal of Computational and Graphical Statistics*, **33**, 1017–1036.

Cheng, Q., Wang, H. and Yang, M. (2020) Information-based optimal subdata selection for big data logistic regression. *Journal of Statistical Planning and Inference*, **209**, 112–122.

Dai, W., Song, Y. and Wang, D. (2023) A subsampling method for regression problems based on minimum energy criterion. *Technometrics*, **65**, 192–205.

DeSarbo, W. S. and Cron, W. L. (1988) A maximum likelihood methodology for clusterwise linear regression. *Journal of Classification*, **5**, 249–282.

Di Mari, R., Rocci, R. and Gattone, S. A. (2017) Clusterwise linear regression modeling with soft scale constraints. *International Journal of Approximate Reasoning*, **91**, 160–178.

Fair, R. C. and Jaffee, D. M. (1972) Methods of Estimation for Markets in Disequilibrium. *Econometrica*, **40**, 177–90.

Galambos, J. (1987) *The Asymptotic Theory of Extreme Order Statistics*. R.E. Krieger Publishing Company.

Hastie, T., Tibshirani, R. and Friedman, J. (2009) *The Elements of Statistical Learning*. New York, NY: Springer New York.

Hawkins, D. S., Allen, D. M. and Stromberg, A. J. (2001) Determining the number of components in mixtures of linear models. *Computational Statistics & Data Analysis*, **38**, 15–48.

Hennig, C. (2000) Identifiability of models for clusterwise linear regression. *Journal of Classification*, **17**, 273–296.

Hosmer, D. W. (1974) Maximum likelihood estimates of the parameters of a mixture of two regression lines. *Communications in Statistics*, **3**, 995–1006.

Jacobs, R. A., Jordan, M. I., Nowlan, S. J. and Hinton, G. E. (1991) Adaptive Mixtures of Local Experts. *Neural Computation*, **3**, 79–87.

Khadka, M. and Paz, A. (2017) Comprehensive Clusterwise Linear Regression for Pavement Management Systems. *Journal of Transportation Engineering, Part B: Pavements*, **143**, 1–13.

Kiefer, J. and Wolfowitz, J. (1959) Optimum Designs in Regression Problems. *The Annals of Mathematical Statistics*, **30**, 271–294.

Makkuva, A., Viswanath, P., Kannan, S. and Oh, S. (2019) Breaking the gridlock in Mixture-of-Experts: Consistent and Efficient Algorithms. In *International Conference on Machine Learning*.

Masoudnia, S. and Ebrahimpour, R. (2014) Mixture of experts: a literature survey. *Artificial Intelligence Review*, **42**, 275–293.

Park, Y. W., Jiang, Y., Klabjan, D. and Williams, L. (2017) Algorithms for Generalized Clusterwise Linear Regression. *INFORMS Journal on Computing*, **29**, 197–376.

Späth, H. (1979) Algorithm 39 Clusterwise linear regression. *Computing*, **22**, 367–373.

Wang, H., Yang, M. and Stufken, J. (2019) Information-Based Optimal Subdata Selection for Big Data Linear Regression. *Journal of the American Statistical Association*, **114**, 393–405.

Wang, L., Elmstedt, J., Wong, W. K. and Xu, H. (2021) Orthogonal subsampling for big data linear regression. *The Annals of Applied Statistics*, **15**, 1273–1290.

Wu, C. F. J. (1983) On the Convergence Properties of the EM Algorithm. *The Annals of Statistics*, **11**, 95–103.

Statistica Sinica: Preprint
doi:10.5705/ss.202023.0302

Yu, J., Ai, M. and Ye, Z. (2023) A review on design inspired subsampling for big data. *Statistical Papers*, **65**, 467–510.

Yuksel, S. E., Wilson, J. N. and Gader, P. D. (2012) Twenty Years of Mixture of Experts. *IEEE Transactions on Neural Networks and Learning Systems*, **23**, 1177–1193.