Information-Based Optimal Subdata Selection for Big Data Linear Regression

HaiYing Wang, Min Yang, and John Stufken *

October 27, 2017

Abstract

Extraordinary amounts of data are being produced in many branches of science. Proven statistical methods are no longer applicable with extraordinary large data sets due to computational limitations. A critical step in big data analysis is data reduction. Existing investigations in the context of linear regression focus on subsampling-based methods. However, not only is this approach prone to sampling errors, it also leads to a covariance matrix of the estimators that is typically bounded from below by a term that is of the order of the inverse of the subdata size. We propose a novel approach, termed information-based optimal subdata selection (IBOSS). Compared to leading existing subdata methods, the IBOSS approach has the following advantages: (i) it is significantly faster; (ii) it is suitable for distributed parallel computing; (iii) the variances of the slope parameter estimators converge to 0 as the full data size increases even if the subdata size is fixed, i.e., the convergence rate depends on the full data size; (iv) data analysis for IBOSS subdata is straightforward and the sampling distribution of an IBOSS estimator is easy to assess. Theoretical results and extensive simulations demonstrate that the IBOSS approach is superior to subsampling-based methods, sometimes by orders of magnitude. The advantages of the new approach are also illustrated through analysis of real data.

Keywords: Massive data; D-optimality; Information matrix; Linear regression; Subdata

*HaiYing Wang is Assistant Professor, Department of Statistics, University of Connecticut, Storrs, Mansfield, CT 06269 (haiying.wang@uconn.edu). Min Yang is Professor, Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago, Chicago, IL 60607 (myang2@uic.edu). John Stufken is Charles Wexler Professor, School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ 85287 (jstufken@asu.edu)

1 Introduction

Technological advances have enabled an exponential growth in data collection and the size of data sets. Although computational resources have also been growing rapidly, this pales in comparison to the astonishing growth in data volume. This presents the challenge of drawing useful information and converting data into knowledge with available computational resources. We meet this challenge in the context of linear regression by identifying informative subdata, which can be fully analyzed.

For linear regression with a $n \times 1$ response vector and $n \times p$ covariate matrix, the full data volume is $n(p+1)$. In the setting of "big data", this prevents computation of parameter estimates in a traditional way due to insufficient computational resources. The scenario with $p \gg n$ is usually referred to as high-dimensional data. Multiple methods for analyzing high-dimensional data have been proposed and studied, such as LASSO (Tibshirani, 1996; Meinshausen *et al.*, 2009), Dantzig selector (Candes and Tao, 2007), and sure independence screening (Fan and Lv, 2008), among others. We focus on the scenario with $n \gg p$ for an extremely large n. This is an important problem that arises in practice. For example, for the chemical sensors data in Section 5.2, $n = 4,208,261$ and $p = 15$. As another example, the airline on-time data set from the 2009 ASA Data Expo contains $n = 123,534,969$ observations on 29 variables about flight arrival and departure information for all commercial flights within the USA, from October 1987 to April 2008. Clearly, not everyone has the computing resources to fully analyze the whole data in the aforementioned examples, and thus data reduction is a crucial step to extract useful information from the data. For the case of $n \gg p$, the required computing time for ordinary least squares (OLS) is $O(np^2)$. This time complexity is too long for big data, and may even be beyond the computational capacity of available computing facilities. To address this computational limitation, data reduction is important. Existing investigations in this direction focus on taking random subsamples from the full data. Interesting studies include Drineas *et al.* (2006, 2011); Ma and Sun (2015); Ma *et al.* (2014, 2015), among others. Parameter estimates are obtained based on a small random subsample of the full data. Normalized statistical leverage scores are often used for nonuniform subsampling probabilities, and the method is known as *algorithmic leveraging* (Ma *et al.*, 2014).

With exact statistical leverage scores, the computing time of this method is $O(np^2)$, just as for OLS on the full data. Drineas *et al.* (2012) developed a randomized algorithm to approximate leverage scores; it is $O(np \log n/\epsilon^2)$, and $o(np^2)$ if $\log n = o(p)$, where $\epsilon \in (0, 0.5]$. Thus the computing time for the algorithmic leveraging method is at least $O(np \log n/\epsilon^2)$. A subsampling-based method also induces sampling error and the information in the resultant subdata is typically proportional to its size. We find that (details to be shown in Section 2), for a subsampling-based estimator, the covariance matrix is bounded from below by a term that is typically of order $1/k$, where k is the subsample size. This order is only a function of k, so that the variance does not go to 0 with increasing full data size n.

In this paper, we propose an alternative subdata selection approach, which we call **i**nformation-**b**ased **o**ptimal **s**ubdata **s**election (IBOSS) method from big data. The basic idea is to select the most informative data points deterministically so that subdata of a small size preserves most of the information contained in the full data. It is akin to the basic motivation of optimal experimental design (Kiefer, 1959), which aims at obtaining the maximum information with a fixed budget. Traditionally, optimal design is not a data analysis tool, but focuses on data collection. The idea of "maximizing" an information matrix, however, can be borrowed to establish a framework to identify the most informative subdata from the full data for estimating unknown parameters. Using this framework, we will also gain more insight into the popular subsampling-based methods.

As we will show, the IBOSS approach has the following advantages compared to existing methods: 1) the computing time for the IBOSS algorithm is $O(np)$, which is significantly faster than existing methods; 2) the IBOSS algorithm is very suitable for distributed parallel computing platforms; it identifies informative data points by examining each covariate individually; 3) the IBOSS method does not induce sampling error and the variance of estimators can go to 0 as the full data size n becomes large even if the subdata size k is fixed; and 4) in terms of distributional properties, IBOSS estimators inherit properties of estimators based on the full data.

The remainder of the paper is organized as follows. In Section 2, we present the IBOSS framework and use it to analyze the popular subsampling-based methods. A lower bound for covariance matrices of subsampling-based estimators will be given. In Section 3, we

characterize IBOSS subdata under the D-optimality criterion and use it to develop a computationally efficient algorithm. In Section 4, we evaluate the IBOSS algorithm by deriving its asymptotic properties. In Section 5, the performance of the IBOSS method is examined through extensive simulations and two real data applications. We offer concluding remarks in Section 6 and show all technical details in the appendix. Additional numerical results are provided in the Supplementary Material.

2 The framework

Let $(\mathbf{z}_1, y_1), ..., (\mathbf{z}_n, y_n)$ denote the full data, and assume the linear regression model:

$$y_i = \beta_0 + \mathbf{z}_i^{\mathrm{T}}\boldsymbol{\beta}_1 + \varepsilon_i = \beta_0 + \sum_{j=1}^{p} z_{ij}\beta_j + \varepsilon_i, \qquad i = 1, ..., n, \tag{1}$$

where β_0 is the scalar intercept parameter, $\boldsymbol{\beta}_1 = (\beta_1, \beta_2, ..., \beta_p)^{\mathrm{T}}$ is a p-dimensional vector of unknown slope parameters, $\mathbf{z}_i = (z_{i1}, ..., z_{ip})^{\mathrm{T}}$ is a covariate vector, y_i is a response, and ε_i is an error term. We write $\mathbf{x}_i = (1, \mathbf{z}_i^{\mathrm{T}})^{\mathrm{T}}$, $\boldsymbol{\beta} = (\beta_0, \boldsymbol{\beta}_1^{\mathrm{T}})^{\mathrm{T}}$, $\mathbf{Z} = (\mathbf{z}_1, ..., \mathbf{z}_n)^{\mathrm{T}}$, $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_n)^{\mathrm{T}}$, $\mathbf{y} = (y_1, ..., y_n)^{\mathrm{T}}$, assume that the y_i's are uncorrelated given the covariate matrix $\mathbf{Z}$, and that the error terms ε_i's satisfy $\mathrm{E}(\varepsilon_i) = 0$ and $\mathrm{V}(\varepsilon_i) = \sigma^2$.

The intercept parameter is often not of interest and could be eliminated by centralizing the full data. However, this does not work for streaming data, is not practical if the focus is on building a predictive model, and requires a computing time of $O(np)$.

When using the full data and model (1), the least-squares estimator of $\boldsymbol{\beta}$, which is also its best linear unbiased estimator (BLUE), is

$$\hat{\boldsymbol{\beta}}_f = \left(\sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}\right)^{-1} \sum_{i=1}^{n} \mathbf{x}_i y_i.$$

The covariance matrix of this unbiased estimator is equal to the inverse of

$$\mathbf{M}_f = \frac{1}{\sigma^2}\sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}.$$

This is the observed Fisher information matrix for $\boldsymbol{\beta}$ from the full data if the ε_i's are normally distributed. While we do not require the normality assumption, for simplicity we will still call $\mathbf{M}_f$ the information matrix. Let $(\mathbf{z}_1^*, y_1^*), ..., (\mathbf{z}_k^*, y_k^*)$ be subdata of size

k selected deterministically from the full data, in which the rule to determine whether a data point is included or not depends on $\mathbf{Z}$ only. Then the subdata also follow the linear regression model

$$y_i^* = \beta_0 + \mathbf{z}_i^{*\mathrm{T}}\boldsymbol{\beta}_1 + \varepsilon_i^*, \qquad i = 1, ..., k, \tag{2}$$

with the same assumptions and unknown parameters as for model (1). The LS estimator

$$\hat{\boldsymbol{\beta}}_s = \left(\sum_{i=1}^{k} \mathbf{x}_i^* \mathbf{x}_i^{*\mathrm{T}}\right)^{-1} \sum_{i=1}^{k} \mathbf{x}_i^* y_i^*,$$

is the BLUE of $\boldsymbol{\beta}$ for model (2) based on the subdata, where $\mathbf{x}_i^* = (1, \mathbf{z}_i^{*\mathrm{T}})^{\mathrm{T}}$. The observed information matrix for $\boldsymbol{\beta}$ based on the subdata is

$$\mathbf{M}_s = \frac{1}{\sigma^2} \sum_{i=1}^{k} \mathbf{x}_i^* \mathbf{x}_i^{*\mathrm{T}},$$

which is the inverse of the covariance matrix of $\hat{\boldsymbol{\beta}}_s$, namely,

$$\mathrm{V}(\hat{\boldsymbol{\beta}}_s|\mathbf{Z}) = \mathbf{M}_s^{-1} = \sigma^2 \left(\sum_{i=1}^{k} \mathbf{x}_i^* \mathbf{x}_i^{*\mathrm{T}}\right)^{-1}. \tag{3}$$

To estimate a linear function of $\boldsymbol{\beta}$ using subdata, plugging in the LS estimator yields the estimator with the minimum variance among all linear unbiased estimators. Since this minimum variance is a function of the covariate values in the subdata, one can judiciously select the subdata to minimize the minimum variance. This is akin to the basic idea behind optimal experimental design (Kiefer, 1959). We implement this here by seeking subdata that, in some sense, maximize $\mathbf{M}_s$.

Some additional notation will help to formulate this idea as an optimization problem. Let δ_i be the indicator variable that signifies whether $(\mathbf{z}_i, y_i)$ is included in the subdata, i.e., $\delta_i = 1$ if $(\mathbf{z}_i, y_i)$ is included and $\delta_i = 0$ otherwise. The information matrix with subdata of size k can then be written as

$$\mathbf{M}(\boldsymbol{\delta}) = \frac{1}{\sigma^2} \sum_{i=1}^{n} \delta_i \mathbf{x}_i \mathbf{x}_i^{\mathrm{T}}, \tag{4}$$

where $\boldsymbol{\delta} = \{\delta_1, \delta_2, ..., \delta_n\}$ such that $\sum_{i=1}^{n} \delta_i = k$. To have an optimal estimator based on a subdata, one can choose a $\boldsymbol{\delta}$ that "maximizes" the information matrix (4). Since $\mathbf{M}(\boldsymbol{\delta})$ is a

matrix, in optimal experimental design (Kiefer, 1959), this is typically done by maximizing a univariate *optimality criterion function* of the matrix.

Let ψ denote an optimality criterion function. The problem is presented as the following optimization problem given the observed big data:

$$\boldsymbol{\delta}^{opt} = \arg\max_{\boldsymbol{\delta}} \psi\{\mathbf{M}(\boldsymbol{\delta})\}, \quad \text{subject to} \quad \sum_{i=1}^{n} \delta_i = k. \tag{5}$$

A popular optimality criterion is the D-optimality criterion. It maximizes the determinant of $\mathbf{M}(\boldsymbol{\delta})$, which has the interpretation of minimizing the expected volume of the joint confidence ellipsoid for $\boldsymbol{\beta}$. We will come back to this criterion in greater detail in Section 3.

2.1 Analysis of existing subsampling-based methods

Based on our information-based subdata selection framework, we can also gain insights into popular random subsampling-based methods. To see this, let $\boldsymbol{\eta}_L$ be the n-dimensional count-vector whose ith entry denotes the number of times that the ith data point is included in a subsample of size k, which is obtained using a random subsampling method with probabilities proportional to π_i, $i = 1, ..., n$, such that $\sum_{i=1}^{n} \pi_i = 1$. A subsampling-based estimator has the general form

$$\tilde{\boldsymbol{\beta}}_L = \left(\sum_{i=1}^{n} w_i \eta_{Li} \mathbf{x}_i \mathbf{x}_i^{\mathrm{T}} \right)^{-1} \sum_{i=1}^{n} w_i \eta_{Li} \mathbf{x}_i y_i, \tag{6}$$

where the weight w_i is often taken to be $1/\pi_i$. Corresponding to different choices of π_i and w_i, some popular subsampling-based methods (Ma *et al.*, 2015) include: uniform subsampling (UNI) in which $\pi_i = 1/n$ and $w_i = 1$; leverage-based subsampling (LEV) in which $\pi_i = h_{ii}/(p+1)$, $w_i = 1/\pi_i$ and $h_{ii} = \mathbf{x}_i^{\mathrm{T}}(\mathbf{X}\mathbf{X}^{\mathrm{T}})^{-1}\mathbf{x}_i$; shrinked leveraging estimator (SLEV) in which $\pi_i = \alpha h_{ii}/(p+1) + (1-\alpha)/n$, $w_i = 1/\pi_i$ and $\alpha \in [0, 1]$; unweighted leveraging estimator (LEVUNW) in which $\pi_i = h_{ii}/(p+1)$ and $w_i = 1$.

The distribution of $\tilde{\boldsymbol{\beta}}_L$ is complicated, but we can study its performance using the proposed information-based framework. The "information matrix" $\mathbf{M}(\boldsymbol{\eta}_L)$ given $\mathbf{Z}$ is random because $\boldsymbol{\eta}_L$ is random. While the η_{Li}'s are correlated, we will only need to use the marginal distribution of each η_{Li}. If subsampling is with replacement, then each η_{Li} has a binomial

distribution $\text{Bin}(k, \pi_i)$; if the subsampling is without replacement, the marginal distribution of each η_{Li} is $\text{Bin}(1, k\pi_i)$ under the condition that $k\pi_i \leq 1$. Either way, $\text{E}(\eta_{Li}) = k\pi_i$. Hence, taking expectations with respect to $\boldsymbol{\eta}_L$, the expected observed information matrix for a subsampling-based method is

$$\mathbf{M}_{EL} = \text{E}\{\mathbf{M}(\boldsymbol{\eta}_L)|\mathbf{Z}\} = \frac{1}{\sigma^2}\sum_{i=1}^{n}\text{E}(\eta_{Li})\mathbf{x}_i\mathbf{x}_i^{\text{T}} = \frac{k}{\sigma^2}\sum_{i=1}^{n}\pi_i\mathbf{x}_i\mathbf{x}_i^{\text{T}}. \tag{7}$$

Unlike the IBOSS approach, the inverse of $\mathbf{M}_{EL}$ is not the variance covariance matrix of $\text{V}(\tilde{\boldsymbol{\beta}}_L|\mathbf{Z})$. In fact, for subsampling with replacement there is a small probability that $\text{V}(\tilde{\boldsymbol{\beta}}_L|\mathbf{Z})$ is not well defined because $\boldsymbol{\beta}_L$ is not estimable. To solve this issue, we consider only subsamples with full-rank covariate matrices to define the covariance matrix of $\tilde{\boldsymbol{\beta}}_L$. The following theorem states a relationship between $\mathbf{M}_{EL}$ and the covariance matrix of $\tilde{\boldsymbol{\beta}}_L$.

Theorem 1. *Suppose that a subsample of size k is taken using a random subsampling procedure with probabilities proportional to π_i, $i = 1, ..., n$, such that $\sum_{i=1}^{n}\pi_i = 1$. Consider the set $\Delta = \{\boldsymbol{\eta}_L : \sum_{i=1}^{n}\eta_{Li}\mathbf{x}_i\mathbf{x}_i^{\text{T}}$ is non-singular$\}$, where $\boldsymbol{\eta}_L$ is the n-dimensional vector that counts how often each data point is included. Let $I_\Delta(\boldsymbol{\eta}_L) = 1$ if and only if $\boldsymbol{\eta}_L \in \Delta$. Given $I_\Delta(\boldsymbol{\eta}_L) = 1$, $\tilde{\boldsymbol{\beta}}_L$ is unbiased for $\boldsymbol{\beta}$, and*

$$\text{V}\{\tilde{\boldsymbol{\beta}}_L|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L) = 1\} \geq P\{I_\Delta(\boldsymbol{\eta}_L) = 1|\mathbf{Z}\}\mathbf{M}_{EL}^{-1} = \frac{\sigma^2 P\{I_\Delta(\boldsymbol{\eta}_L) = 1|\mathbf{Z}\}}{k}\left\{\sum_{i=1}^{n}\pi_i\mathbf{x}_i\mathbf{x}_i^{\text{T}}\right\}^{-1} \tag{8}$$

in the Loewner ordering.

Remark 1. Theorem 1 is true regardless of the choice for the weights w_i's in $\tilde{\boldsymbol{\beta}}_L$ or whether subsampling is with or without replacement. It provides a lower bound for covariance matrices of subsampling-based estimators, which provides a feasible way to evaluate the best performance of a subsampling-based estimator. Note that when $k \gg p + 1$, $P\{I_\Delta(\boldsymbol{\eta}_L) = 1|\mathbf{Z}\}$ is often close to 1, so that the lower bound is close to the inverse of the expected observed information $\mathbf{M}_{EL}$.

Remark 2. Existing investigations on the random subsampling approach focus on sampling with replacement where the subsample is independent given the full data. For subsampling without replacement with fixed sample size, the subsample is no longer independent given

the full data and the properties of the resultant estimator are more complicated. As a result, to the best of our knowledge, subsampling without replacement for a fixed subsample size from big data has never been investigated. Our IBOSS framework, however, is applicable here to assess the performance of an estimator based on subsampling without replacement.

Applying Theorem 1 to the popular sampling-based methods, we obtain the following results. For the UNI method,

$$\mathrm{V}\left\{\tilde{\boldsymbol{\beta}}_L^{\mathrm{UNI}}\middle|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\right\} \geq \frac{\sigma^2 P\{I_\Delta(\boldsymbol{\eta}_L)=1|\mathbf{Z}\}}{k}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}\right)^{-1}; \tag{9}$$

for the LEV (or LEVUNW) method,

$$\mathrm{V}\left\{\tilde{\boldsymbol{\beta}}_L^{\mathrm{LEV}}\middle|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\right\} \geq \frac{(p+1)\sigma^2 P\{I_\Delta(\boldsymbol{\eta}_L)=1|\mathbf{Z}\}}{k}\left\{\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}\right\}^{-1}; \tag{10}$$

for the SLEV method,

$$\begin{aligned}\mathrm{V}\left\{\tilde{\boldsymbol{\beta}}_L^{\mathrm{SLEV}}\middle|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\right\} \geq & \frac{\sigma^2 P\{I_\Delta(\boldsymbol{\eta}_L)=1|\mathbf{Z}\}}{k}\left\{\frac{\alpha}{p+1}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}\right. \\ & \left.+\frac{1-\alpha}{n}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}\right\}^{-1}. \end{aligned} \tag{11}$$

If $\mathbf{x}_i$, $i=1,...,n$, are generated independently from the same distribution as random vector $\mathbf{x}$ with finite second moment, i.e., $\mathrm{E}\|\mathbf{x}\|^2<\infty$, then from the strong law of large numbers,

$$\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}} \to \mathrm{E}(\mathbf{x}\mathbf{x}^{\mathrm{T}}), \tag{12}$$

almost surely as $n\to\infty$. If we further assume that the fourth moment of $\mathbf{x}$ is finite, i.e., $\mathrm{E}\|\mathbf{x}\|^4<\infty$, then we have

$$\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}} \to \mathrm{E}\left[\mathbf{x}\mathbf{x}^{\mathrm{T}}\{\mathrm{E}(\mathbf{x}\mathbf{x}^{\mathrm{T}})\}^{-1}\mathbf{x}\mathbf{x}^{\mathrm{T}}\right] \tag{13}$$

almost surely as $n\to\infty$. Let $P_\eta=\liminf_{n\to\infty}P\{I_\Delta(\boldsymbol{\eta}_L)=1|\mathbf{Z}\}$. Note that $P_\eta=1$ under some mild condition, e.g., the covariate distribution is continuous. From (9) - (13), we have

$$\mathrm{V}\left\{\tilde{\boldsymbol{\beta}}_L^{\mathrm{UNI}}\middle|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\right\} \geq \frac{\sigma^2 P_\eta}{k}\left\{\mathrm{E}(\mathbf{x}\mathbf{x}^{\mathrm{T}})\right\}^{-1}, \tag{14}$$

$$\mathrm{V}\left\{\tilde{\boldsymbol{\beta}}_L^{\mathrm{LEV}}\middle|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\right\} \geq \frac{(p+1)\sigma^2 P_\eta}{k}\left(E[\mathbf{x}\mathbf{x}^{\mathrm{T}}\{\mathrm{E}(\mathbf{x}\mathbf{x}^{\mathrm{T}})\}^{-1}\mathbf{x}\mathbf{x}^{\mathrm{T}}]\right)^{-1}, \tag{15}$$

$$\mathrm{V}\left\{\tilde{\boldsymbol{\beta}}_L^{\mathrm{SLEV}}\middle|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\right\} \geq \frac{\sigma^2 P_\eta}{k}\left(\frac{\alpha}{p+1}E[\mathbf{x}\mathbf{x}^{\mathrm{T}}\{\mathrm{E}(\mathbf{x}\mathbf{x}^{\mathrm{T}})\}^{-1}\mathbf{x}\mathbf{x}^{\mathrm{T}}]+(1-\alpha)\mathrm{E}(\mathbf{x}\mathbf{x}^{\mathrm{T}})\right)^{-1}, \tag{16}$$

almost surely as $n \to \infty$. From (14), (15) and (16), one sees that covariance matrices of these commonly used subsampling-based estimators are bounded from below in the Loewner ordering by finite quantities that are at the order of $1/k$. These quantities do not go to 0 as the full data sample size n goes to ∞.

3 The D-optimality criterion and an IBOSS algorithm

In this section, we study the commonly used D-optimality criterion and develop IBOSS algorithms based on theoretical characterizations of IBOSS subdata under this criterion.

In our framework, for given full data of size n, the D-optimality criterion suggests the selection of subdata of size k so that

$$\boldsymbol{\delta}_{\mathrm{D}}^{opt} = \arg\max_{\boldsymbol{\delta}}\left|\sum_{i=1}^{n}\delta_i\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}\right|, \quad \sum_{i=1}^{n}\delta_i = k.$$

Obtaining an exact solution is computationally far too expensive. In working towards an approximate solution, we first derive an upper bound for $|\mathbf{M}(\boldsymbol{\delta})|$ which, while only attainable for very special cases, will guide our later algorithm.

Theorem 2 (D-optimality)**.** *For subdata of size k represented by $\boldsymbol{\delta}$,*

$$|\mathbf{M}(\boldsymbol{\delta})| \leq \frac{k^{p+1}}{4^p\sigma^{2(p+1)}}\prod_{j=1}^{p}(z_{(n)j}-z_{(1)j})^2, \tag{17}$$

where $z_{(n)j} = \max\{z_{1j}, z_{2j}, ..., z_{nj}\}$ and $z_{(1)j} = \min\{z_{1j}, z_{2j}, ..., z_{nj}\}$ are the nth and first order statistics of $z_{1j}, z_{2j}, ..., z_{nj}$. If the subdata consists of the 2^p points $(a_1, \ldots, a_p)^{\mathrm{T}}$ where $a_j = z_{(n)j}$ or $z_{(1)j}$, $j = 1, 2, ..., p$, each occurring equally often, then equality holds in (17)*.*

Remark 3. Often k is much smaller than 2^p, so that subdata with equality in Theorem 2 will not exist. However, just as for Hadamard's determinant bound (Hadamard, 1893),

the result suggests to collect subdata with extreme covariate values, both small and large, occurring with the same frequency. This agrees with the common statistical knowledge that larger variation in covariates is more informative and results in better parameter estimation.

The following algorithm is motivated by the result in Theorem 2.

Algorithm 1 (Algorithm motivated by D-optimality)**.** *Suppose that $r = k/(2p)$ is an integer. Using a partition-based selection algorithm (Martínez, 2004), perform the following steps:*

(1) For z_{i1}, $1 \leq i \leq n$, include r data points with the r smallest z_{i1} values and r data points with the r largest z_{i1} values;

(2) For $j = 2, ..., p$, exclude data points that were previously selected, and from the remainder select r data points with the smallest z_{ij} values and r data points with the largest z_{ij} values.

(3) Return $\hat{\boldsymbol{\beta}}^{\mathrm{D}} = \{(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{X}_{\mathrm{D}}^*\}^{-1}(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{y}_{\mathrm{D}}^*$ and the estimated covariance matrix $\hat{\sigma}_{\mathrm{D}}^2\{(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{X}_{\mathrm{D}}^*\}^{-1}$, where $\mathbf{X}_{\mathrm{D}}^* = (\mathbf{1}, \mathbf{Z}_{\mathrm{D}}^*)$, $\mathbf{Z}_{\mathrm{D}}^*$ is the covariate matrix of the subdata selected in the previous steps, $\mathbf{y}_{\mathrm{D}}^*$ is the response vector of the subdata and $\hat{\sigma}_{\mathrm{D}}^2 = \left\|\mathbf{y}_{\mathrm{D}}^* - \mathbf{X}_{\mathrm{D}}^*\hat{\boldsymbol{\beta}}^{\mathrm{D}}\right\|^2/(k-p-1)$.

Remark 4. For each covariate, a partition-based selection algorithm has an average time complexity of $O(n)$ to find the rth largest or smallest value (Musser, 1997; Martínez, 2004). Thus the time to obtain the subdata is $O(np)$. Using the subdata, the computing time for $\hat{\boldsymbol{\beta}}^{\mathrm{D}}$ and $\hat{\sigma}_{\mathrm{D}}^2$ is $O(kp^2+p^3)$ and $O(kp)$, respectively. Thus, the time complexity of Algorithm 1 is $O(np + kp^2 + p^3 + kp) = O(np + kp^2)$. For the scenario that $n > kp$, this reduces to $O(np)$. This algorithm is faster than algorithmic leveraging, which has a computing time of $O(np \log n)$ (Drineas *et al.*, 2012).

Remark 5. Algorithm 1 gives the covariance matrix of the resultant estimator, which is very crucial for statistical inference. This is the exact covariance matrix of $\hat{\boldsymbol{\beta}}^{\mathrm{D}}$ if the variance of the error term, σ^2, is known. With an additional assumption of normality of ε_i, $\hat{\boldsymbol{\beta}}^{\mathrm{D}}$ has an exact normal distribution.

Remark 6. Algorithm 1 is naturally suited for distributed storage and processing facilities for parallel computing. One can simultaneously process each covariate and find the indexes

of its extreme values. These indexes can then be combined to obtain the subdata. While this approach may result in a subdata size that is smaller than k if there is duplication of indexes, the resultant estimator will still have the same convergence rate.

Remark 7. Algorithm 1 selects subdata according to extreme values of each covariate, which may include outliers. However, the selection rule is ancillary, and the resultant subdata follow the same underlying regression model as the full data. We can thus use outlier diagnostic methods to identify outliers in the subdata. If there are outliers in the full data, it is very likely that these data points will be identified as outliers in the subdata. On the other hand, if there are data points that are far from others but still follow the underlying model, then these data points actually contain more information about the model and should be used for parameter estimation.

Remark 8. The restriction that the subdata sample size k is chosen to make $r = k/(2p)$ an integer is mostly for convenience. In the case that $r = k/(2p)$ is not an integer, one can either adjust k by using the floor $\lfloor r \rfloor$ or the ceiling $\lceil r \rceil$, or use a combination of $\lfloor r \rfloor$ and $\lceil r \rceil$ to keep the subdata sample size as k.

The following theorem gives some insight on the quality of using Algorithm 1 to approximate the upper bound of $|\mathbf{M}(\boldsymbol{\delta})|$ in Theorem 2.

Theorem 3. *Let $\mathbf{Z}_{\mathrm{D}}^*$ be the covariate matrix for the subdata of size $k = 2pr$ selected using Algorithm 1 and $\mathbf{X}_{\mathrm{D}}^* = (\mathbf{1}, \mathbf{Z}_{\mathrm{D}}^*)$. The determinant $|(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{X}_{\mathrm{D}}^*|$ satisfies*

$$\frac{|(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{X}_{\mathrm{D}}^*|}{\frac{k^{p+1}}{4^p}\prod_{j=1}^{p}(z_{(n)j} - z_{(1)j})^2} \geq \frac{\lambda_{\min}^p(\mathbf{R})}{p^p}\prod_{j=1}^{p}\left(\frac{z_{(n-r+1)j} - z_{(r)j}}{z_{(n)j} - z_{(1)j}}\right)^2, \tag{18}$$

where $\lambda_{\min}(\mathbf{R})$ is the smallest eigenvalue of $\mathbf{R}$, the sample correlation matrix of $\mathbf{Z}_{\mathrm{D}}^*$.

From this theorem, it is seen that although Algorithm 1 may not achieve the unachievable upper bound in Theorem 2, it may achieve the same order. For example, if p is fixed and $\liminf_{n\to\infty} \lambda_{\min}(\mathbf{R}) > 0$, then under reasonable assumptions, the lower bound in Equation (18) will not converge to 0 as $n \to \infty$. This means that $|(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{X}_{\mathrm{D}}^*|$ is of the same order as the upper bound for $|\mathbf{M}(\boldsymbol{\delta})|$ in Theorem 2, even though the latter is typically not attainable.

4 Properties of parameter estimator

In this section, we investigate the theoretical properties of the D-optimality motivated IBOSS algorithm, and provide both finite sample assessment and asymptotic results. These results are provided to evaluate the performance of the proposed method and show more insights about the IBOSS approach. The application of the D-optimality motivated IBOSS algorithm does not depend on asymptotic properties of the approach.

Since $\hat{\boldsymbol{\beta}}^{\mathrm{D}}$ is unbiased for $\boldsymbol{\beta}$, we focus on its variance. The next theorem gives bounds on variances of estimators of the intercept and slope parameters from the D-optimality motivated algorithm.

Theorem 4. *If $\lambda_{\min}(\mathbf{R}) > 0$, then, the following results hold for the estimator, $\hat{\boldsymbol{\beta}}^{\mathrm{D}}$, obtained from Algorithm 1:*

$$\mathrm{V}(\hat{\beta}_0^{\mathrm{D}}|\mathbf{Z}) \geq \frac{\sigma^2}{k}, \tag{19}$$

$$\frac{4\sigma^2}{k\lambda_{\max}(\mathbf{R})(z_{(n)j} - z_{(1)j})^2} \leq \mathrm{V}(\hat{\beta}_j^{\mathrm{D}}|\mathbf{Z}) \leq \frac{4p\sigma^2}{k\lambda_{\min}(\mathbf{R})(z_{(n-r+1)j} - z_{(r)j})^2}, \quad j = 1, ..., p. \tag{20}$$

Theorem 4 describes finite sample properties of the proposed estimator and does not require any quantity to go to ∞. It shows that the variance of the intercept estimator is bounded from below by a term proportional to the inverse subdata size. This is similar to the results for existing subsampling methods. However, for the slope estimator, the variance is bounded from above by a term that is proportional to $\frac{p}{k(z_{(n-r+1)j} - z_{(r)j})^2}$, which may converge to 0 as n increases even when the subdata size k is fixed. We present this asymptotic result in the following theorem.

Theorem 5. *Assume that covariate distributions are in the domain of attraction of the generalized extreme value distribution, and $\liminf_{n\to\infty} \lambda_{\min}(\mathbf{R}) > 0$. For large enough n, the following results hold for the estimator, $\hat{\boldsymbol{\beta}}^{\mathrm{D}}$, obtained from Algorithm 1:*

$$\mathrm{V}(\hat{\beta}_j^{\mathrm{D}}|\mathbf{Z}) = O_P\left\{\frac{p}{k(z_{(n-r+1)j} - z_{(r)j})^2}\right\}, \quad j = 1, ..., p. \tag{21}$$

Furthermore,

$$\mathrm{V}(\hat{\beta}_j^{\mathrm{D}}|\mathbf{Z}) \asymp_P \frac{p}{k(z_{(n)j} - z_{(1)j})^2}, \quad j = 1, ..., p, \tag{22}$$

if one of the following conditions holds: 1) r is fixed; 2) the support of F_j is bounded, $r \to \infty$, and $r/n \to 0$, where F_j is the marginal distribution function of the jth component of $\mathbf{z}$; 3) the upper endpoint for the support of F_j is ∞ and the lower endpoint for the support of F_j is finite, and $r \to \infty$ slow enough such that

$$\frac{r}{n[1 - F_j\{(1-\epsilon)F_j^{-1}(1-n^{-1})\}]} \to 0, \tag{23}$$

for all $\epsilon > 0$; 4) the upper endpoint for the support of F_j is finite and the lower endpoint for the support of F_j is $-\infty$, and $r \to \infty$ slow enough such that

$$\frac{r}{nF_j\{(1-\epsilon)F_j^{-1}(n^{-1})\}} \to 0, \tag{24}$$

for all $\epsilon > 0$; 5) the upper endpoint and the lower endpoint for the support of F_j are ∞ and $-\infty$, respectively, and (23) *and* (24) *hold.*

Equation (21) gives a general result on the variance of a slope estimator. It holds for any values of n, r and p, so that it can also be used to obtain asymptotic results when one or more of n, r and p go to infinity. The expression shows that if $p/(z_{(n-r+1)j} - z_{(r)j})^2 = o_P(1)$, then the convergence of the variance would be faster than $1/k$, the typical convergence rate for a subsampling method (Ma *et al.*, 2015; Wang *et al.*, 2017). Note that the results are derived from the upper bound in (20), and thus the real convergence of the variance can be faster than $\frac{p}{k(z_{(n-r+1)j} - z_{(r)j})^2}$.

For the condition in (23), it can be satisfied by many commonly seen distributions, such as exponential distribution, double exponential distribution, lognormal distribution, normal distribution, and gamma distribution (Hall, 1979). For different distributions, the required rate at which $r \to \infty$ is different. For example, if F_j is a normal distribution function, then (23) holds if and only if $\log r / \log\log n \to 0$; if F_j is an exponential distribution function, then (23) holds if and only if $\log r / \log n \to 0$. The condition in (24) is the same as that in (23) if one take $\mathbf{z} = -\mathbf{z}$.

For the result in (22), it can be shown from the proof that

$$\mathrm{V}(\hat{\beta}_j^{\mathrm{D}} | \mathbf{Z}) \asymp_P \frac{p}{k\left\{F_j^{-1}(1-n^{-1}) - F_j^{-1}(n^{-1})\right\}^2}, \quad j = 1, ..., p.$$

What we find more interesting is the fact that, when k is fixed, from Theorems 2.8.1 and 2.8.2 in Galambos (1987), $z_{(n-r+1)j} - z_{(r)j}$ goes to infinity with the same rate as that

of $z_{(n)j} - z_{(1)j}$. Thus the order of the variance of a slope estimator is the inverse of the squared full data sample range for the corresponding covariate. If the sample range goes to ∞ as $n \to \infty$, then the variance converges to 0 even when the subdata size k is fixed. This suggests that subdata may preserve information at a scale related to the full data size. We will return to this for specific cases with more details. In the remainder of this section, we focus on the case that both p and k are fixed.

That the variance $\mathrm{V}(\hat{\beta}_0^{\mathrm{D}}|\mathbf{Z})$ does not go to 0 for a fixed subdata size k is not a concern if inference for the slope parameters is of primary interest, as is often the case. However, if the focus is on building a predictive model, the intercept needs to be estimated more precisely. This can be done by using the full data means, $\bar{y}$ and $\bar{\mathbf{z}}$, say. After obtaining the slope estimator $\hat{\boldsymbol{\beta}}_1^{\mathrm{D}}$, compute the following adjusted estimator of the intercept

$$\hat{\beta}_0^{Da} = \bar{y} - \bar{\mathbf{z}}^{\mathrm{T}} \hat{\boldsymbol{\beta}}_1^{\mathrm{D}}. \tag{25}$$

The estimator $\hat{\beta}_0^{Da}$ has a convergence rate similar to that of the slope parameter estimators, because $\hat{\beta}_0^{Da} - \beta_0 = (\hat{\beta}_0^{\mathrm{full}} - \beta_0) + \bar{\mathbf{z}}^{\mathrm{T}}(\hat{\boldsymbol{\beta}}_1^{\mathrm{full}} - \boldsymbol{\beta}_1) - \bar{\mathbf{z}}^{\mathrm{T}}(\hat{\boldsymbol{\beta}}_1^{\mathrm{D}} - \boldsymbol{\beta}_1)$ and the last term is the dominating term if $\mathrm{E}(\mathbf{z}) \neq \mathbf{0}$. The rate may be faster than that of the slope parameter estimators if $\mathrm{E}(\mathbf{z}) = \mathbf{0}$. The additional computing time for this approach is $O(np)$, but the estimation efficiency for β_0 will be substantially improved. We will demonstrate this numerically in Section 5.

Whereas Theorem 5 provides a general result for the variance of individual parameter estimators, more can be said for special cases. The next theorem studies the structure of the covariance matrix for estimators based on Algorithm 1 under various assumptions.

Theorem 6. *Let* $\boldsymbol{\mu} = (\mu_1, ..., \mu_p)^{\mathrm{T}}$ *and let* $\boldsymbol{\Sigma} = \boldsymbol{\Phi}\boldsymbol{\rho}\boldsymbol{\Phi}$ *be a full rank covariance matrix, where* $\boldsymbol{\Phi} = \mathrm{diag}(\sigma_1, ..., \sigma_p)$ *is a diagonal matrix of standard deviations and* $\boldsymbol{\rho}$ *is a correlation matrix. Assume that* $\mathbf{z}_i$*'s,* $i = 1, ..., n$*, are i.i.d. with a distribution specified below. The following results hold for* $\hat{\boldsymbol{\beta}}^{\mathrm{D}}$*, the estimator from Algorithm 1.*
(i) For multivariate normal covariates, i.e., $\mathbf{z}_i \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$*,*

$$\mathrm{V}(\mathbf{A}_n \hat{\boldsymbol{\beta}}^{\mathrm{D}}|\mathbf{Z}) = \frac{\sigma^2}{2k} \begin{bmatrix} 2 & \mathbf{0} \\ \mathbf{0} & p(\boldsymbol{\Phi}\boldsymbol{\rho}^2\boldsymbol{\Phi})^{-1} \end{bmatrix} + O_P\left(\frac{1}{\sqrt{\log n}}\right), \tag{26}$$

where $\mathbf{A}_n = \text{diag}\big(1, \sqrt{\log n}, ..., \sqrt{\log n}\big)$.

(ii) For multivariate lognormal covariates, i.e., $\mathbf{z}_i \sim \text{LN}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$,

$$\text{V}(\mathbf{A}_n \hat{\boldsymbol{\beta}}^{\text{D}} | \mathbf{Z}) = \frac{2\sigma^2}{k} \begin{bmatrix} 1 & -\mathbf{u}^{\text{T}} \\ -\mathbf{u} & p\boldsymbol{\Lambda} + \mathbf{u}\mathbf{u}^{\text{T}}, \end{bmatrix} + o_P(1) \tag{27}$$

where $\mathbf{A}_n = \text{diag}\Big\{1, \exp\big(\sigma_1\sqrt{2\log n}\big), ..., \exp\big(\sigma_p\sqrt{2\log n}\big)\Big\}$, $\mathbf{u} = (e^{-\mu_1}, ..., e^{-\mu_p})^{\text{T}}$ *and* $\boldsymbol{\Lambda} = \text{diag}(e^{-2\mu_1}, ..., e^{-2\mu_p})$.

For the distributions in Theorem 6, $\text{V}(\hat{\beta}_0^{\text{D}} | \mathbf{Z})$ is proportional to $1/k$ and never converges to 0 with a fixed k. Based on Theorem 6, $\text{V}(\hat{\boldsymbol{\beta}}_1^{\text{D}} | \mathbf{Z})$ converges to 0 at different rates for different distributions. When $\mathbf{z}$ has a normal distribution, the convergence rate of components of $\text{V}(\hat{\boldsymbol{\beta}}_1^{\text{D}} | \mathbf{Z})$ is $1/\log n$. When $\mathbf{z}$ has a lognormal distribution, the component $\text{V}_{j_1 j_2}(\hat{\boldsymbol{\beta}}_1^{\text{D}} | \mathbf{Z})$ has a convergence rate $\exp\big\{-(\sigma_{j_1} + \sigma_{j_2})\sqrt{2\log n}\big\}$, $j_1, j_2 = 1, ..., p$. In comparison, for most popular subsampling-based methods, from the results in Section 2.1, for the normal and lognormal distributions, variances of the slope parameter estimators never converge to 0 because they are bounded from below by terms that are proportional to $1/k$.

For a subsampling-based method, if the covariate distribution is sufficiently heavy-tailed, then some components of the lower bound in Theorem 1 can go to 0. However, even then the convergence rate is much slower than that for the IBOSS approach, which may then produce an estimator with a convergence rate close to that of the full data estimator. For example, Table 1 summarizes the orders of variances for parameter estimators when the only covariate z in a simple linear regression model has a t distribution with degrees of freedom ν. Three approaches are compared: the D-optimality motivated IBOSS approach (D-OPT), the UNI approach and the full data approach (FULL).

For β_0, neither subdata approach produces a variance that goes to 0. For β_1, the D-OPT IBOSS approach results in a variance that goes to 0 at a rate of $n^{-2/\nu}$. When $\nu \leq 2$, the variance of the estimator based on the full data goes to 0 at a rate that is slower than $n^{-(2/\nu+\alpha)}$ for any $\alpha > 0$, so that the D-OPT IBOSS approach reaches a rate that is very close to that of the full data. For the UNI approach, the lower bound of the variance goes to 0 at a much slower rate. Note that the convergence to 0 does not contradict the conclusion in (14), which assumes that the $\mathbf{x}_i$ are i.i.d. with finite second moment.

Table 1: Orders of variances and orders of lower bounds of variances when the covariate has a t_ν distribution. The orders are in probability.

Methods	Covariates are t_ν		
	β_0	β_1	
		$\nu > 2$	$\nu \leq 2$
D-OPT	$1/k$	$1/(kn^{2/\nu})$	$1/(kn^{2/\nu})$
UNI	$1/k$	$1/k$	slower than $1/\{kn^{(2/\nu-1+\alpha)}\}$ for any $\alpha > 0$
FULL	$1/n$	$1/n$	slower than $1/\{kn^{(2/\nu+\alpha)}\}$ for any $\alpha > 0$

5 Numerical experiments

Using simulated and real data, we will now evaluate the performance of the IBOSS method.

5.1 Simulation studies

Data are generated from the linear model (1) with the true value of $\boldsymbol{\beta}$ being a 51 dimensional vector of unity and $\sigma^2 = 9$. An intercept is included so $p = 50$. Let $\boldsymbol{\Sigma}$ be a covariance matrix with $\Sigma_{ij} = 0.5^{I(i \neq j)}$, for $i, j = 1, ..., 50$, where $I()$ is the indicator function. Covariates $\mathbf{z}_i$'s are generated according to the following scenarios.

Case 1. $\mathbf{z}_i$'s have a multivariate normal distribution, i.e., $\mathbf{z}_i \sim N(\mathbf{0}, \boldsymbol{\Sigma})$.

Case 2. $\mathbf{z}_i$'s have a multivariate lognormal distribution, i.e., $\mathbf{z}_i \sim \text{LN}(\mathbf{0}, \boldsymbol{\Sigma})$.

Case 3. $\mathbf{z}_i$'s have a multivariate t distribution with degrees of freedom $\nu = 2$, i.e., $\mathbf{z}_i \sim t_2(\mathbf{0}, \boldsymbol{\Sigma})$.

Case 4. $\mathbf{z}_i$'s have a mixture distribution of four different distributions, $N(\mathbf{1}, \boldsymbol{\Sigma})$, $t_2(\mathbf{1}, \boldsymbol{\Sigma})$, $t_3(\mathbf{1}, \boldsymbol{\Sigma})$, $\text{U}[\mathbf{0}, \mathbf{2}]$ and $\text{LN}(\mathbf{0}, \boldsymbol{\Sigma})$ with equal proportions, where $\text{U}[\mathbf{0}, \mathbf{2}]$ means its components are independent uniform distributions between 0 and 2.

Case 5. $\mathbf{z}_i$'s consist of multivariate normal random variables with interactions and quadratic terms. To be specific, denote $\mathbf{v} = (v_1, ..., v_{20})^{\text{T}} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_{20\times 20})$, where $\boldsymbol{\Sigma}_{20\times 20}$ is the 20 by 20 upper diagonal sub-matrix of $\boldsymbol{\Sigma}$. Let $\mathbf{z} = (\mathbf{v}^{\text{T}}, v_1\mathbf{v}^{\text{T}}, v_2 v_{11}, v_2 v_{12}, ..., v_2 v_{20})^{\text{T}}$ and $\mathbf{z}_i$'s are generated from the distribution of $\mathbf{z}$.

The simulation is repeated $S = 1000$ times and empirical mean squared errors (MSE) are calculated using $\text{MSE}_{\beta_0} = S^{-1}\sum_{s=1}^{S}(\hat{\beta}_0^{(s)} - \beta_0)^2$ and $\text{MSE}_{\boldsymbol{\beta}_1} = S^{-1}\sum_{s=1}^{S}\|\hat{\boldsymbol{\beta}}_1^{(s)} - \boldsymbol{\beta}_1\|^2$ for intercept and slope estimators from different approaches, where $\hat{\beta}_0^{(s)}$, and $\hat{\boldsymbol{\beta}}_1^{(s)}$ are estimates in the sth repetition. We compare four different approaches: D-OPT, the D-optimality motivated IBOSS algorithm described in Algorithm 1 (black solid line —○—), UNI (green short dotted line ···+···), LEV (blue dashed line ·-×-·), and FULL, the full data approach (aqua long dashed line –◇–). To get the best performance of the LEV method in parameter estimation, exact statistical leverage scores are used to calculate the subsampling probabilities. Note that for linear regression, the divide-and-conquer method produces results that are identical to these from the FULL (Lin and Xie, 2011; Schifano *et al.*, 2016), while the computational cost is not lower than the FULL. Thus the comparisons between the IBOSS approach and the full data approach reflect the relative performance of the IBOSS and the divide-and-conquer method in the context of linear regression.

For full data sizes $n = 5\times10^3$, 10^4, 10^5 and 10^6 and fixed subdata size $k = 10^3$, Figures 1 and 2 present plots of the $\log_{10}$ of the MSEs against $\log_{10}(n)$. Figure 1 gives the $\log_{10}$ of the MSEs for estimating the slope parameter $\boldsymbol{\beta}_1$ using different methods. As seen in the plots, the D-OPT IBOSS method uniformly dominates the subsampling-based methods UNI and LEV, and its advantage is more significant if the tail of the covariate distribution is heavier. More importantly, the MSEs from the D-OPT IBOSS method for estimating $\boldsymbol{\beta}_1$ decrease as the full data sample size n increases, even though the subdata size is fixed at $k = 10^3$. For the normal covariate distribution in Figure 1(a), the decrease in the MSE for the D-OPT IBOSS estimator is not as evident because, as shown in Theorem 6, the convergence rate of variances for this case is as slow as $p/k/\log n$. To show the relative performance of the D-OPT IBOSS approach compared to that of the subsampling-based approaches, in the right panel of Figure 1(a), we scale all the MSEs so that the MSEs for the UNI method are one. From this figure, the MSEs for the D-OPT IBOSS approach are about 80% of those for the subsampling-based approaches.

Unlike the IBOSS method, the random subsampling-based methods yield MSEs that show very little change with increasing n except for the t_2 and mixture covariate distributions. This agrees with the conclusion in Theorem 1 that the covariance matrix for a

random subsampling-based estimator is bounded from below by a matrix that depends only on the subdata size k if the fourth moment of the covariate distribution is finite. For the t_2 and mixture covariate distributions, the second moments of the covariate distributions are not finite and we see that the MSEs decrease as n becomes larger. However, the convergence rates are much slower than for the IBOSS method.

For the full data approach, where all data points are used, the MSEs decrease as the size n increases. It is noteworthy that the performance of the D-OPT IBOSS method can be comparable to that of using the full data for estimating $\boldsymbol{\beta}_1$. For example, as shown in Figure 1 (d) for the mixture of distributions, an analysis using the D-OPT IBOSS method with subdata size $k = 10^3$ from full data of size $n = 10^6$ outperforms a full data analysis with data of size $n = 10^5$; the MSE from the full data analysis is 2.4 times as large as the MSE from using subdata of size $k = 10^3$.

Figure 2 gives results for estimating the intercept parameter β_0. In general, the D-OPT IBOSS method is superior to other subdata based methods, but its MSE does not decrease as the full data size increases. This agrees with the result in Theorem 4. We also calculate the MSE of the adjusted estimator in (25), $\hat{\beta}_0^{Da} = \bar{y} - \bar{\mathbf{z}}^{\mathrm{T}}\hat{\boldsymbol{\beta}}_1^{\mathrm{D}}$, which is labeled with D-OPTa (red dashed line - ⊖ -) in Figure 2. It is seen that the relative performances of $\hat{\beta}_0^{Da}$ for Cases 2, 4 and 5 are similar to those of the slope estimator, which agrees with the asymptotic properties discussed below (25) in Section 4. For Cases 1 and 3, the results are very interesting in that $\hat{\beta}_0^{Da}$ performs as good as the full data approach, which means that the convergence rate of $\hat{\beta}_0^{Da}$ is much faster than that of the slope estimator. This seems surprising, but it agrees with the asymptotic properties discussed below (25) in Section 4. For these two cases, $\mathrm{E}(\mathbf{z}) = \mathbf{0}$ and this is the reason why the convergence rate can be faster than that of the slope parameter estimators.

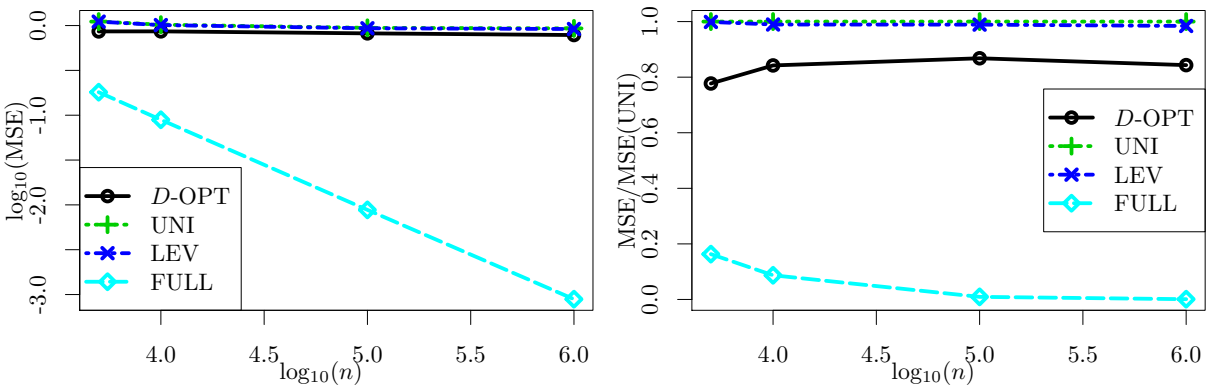
To see the effect of the subdata size k for estimating the slope parameter $\boldsymbol{\beta}_1$, Figure 3 presents plots of the $\log_{10}$ of the MSEs against the subdata size k, with choices $k = 200$, 400, 500, 10^3, 2×10^3, 3×10^3 and 5×10^3, for fixed full data size $n = 10^6$. The MSE for using the full data is a constant with respect to k and is plotted for comparison. Clearly, all subdata-based methods improve as the subdata size k increases, with the D-OPT IBOSS method again being the best performer. For example, when $n = 10^6$ for the mixture

covariate distribution, the analysis based on the D-OPT IBOSS method with $k = 200$ is about 10 times as accurate as that of the LEV method with $k = 5 \times 10^3$ as measured by the MSE value.

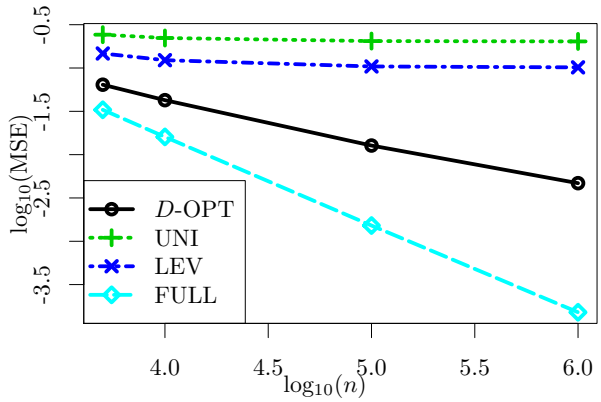
To evaluate the performance of the D-OPT IBOSS approach for statistical inference, we calculate the empirical coverage probabilities and average lengths of the 95% confidence intervals from this method. Results for the full data analysis are also computed for comparison. Figure 4 gives results for the normal and mixture covariate distributions. The estimated parameter is the first slope parameter β_1. Confidence intervals are constructed using $\hat{\beta}_1^{(s)} \pm Z_{0.975} SE_1^{(s)}$, where $\hat{\beta}_1^{(s)}$ and $SE_1^{(s)}$ are the estimate and its standard error of β_1 in the sth repetition, and $Z_{0.975}$ is the 97.5th percentile of the standard normal distribution. It is seen that all empirical coverage levels are close to the nominal level of 0.95, which shows that the inference based on IBOSS subdata is valid. We do not compare this to subsampling-based approaches because we are not aware of theoretically justified methods for constructing confidence intervals under these approaches.

Results on computational efficiency of the D-OPT IBOSS approach are presented in Table 2, which shows CPU times (in seconds) for different combinations of the full data size n and the number of covariates p for a fixed subdata size of $k = 10^3$ and normal distribution for the $\mathbf{z}_i$'s. The R programming language (R Core Team, 2015) is used to implement each method. For the IBOSS approach, it requires a partition-based partial sort algorithm which is not available in R, so the standard C++ function "*nth_element*" (Stroustrup, 1986), is called from R for partial sorting. In order to get good performance in terms of CPU times for the LEV method, the leverage scores are approximated using the fast algorithm in Drineas *et al.* (2012). The CPU times for using the full data are also presented for comparison. All computations are carried out on a desktop running Windows 10 with an Intel I7 processor and 16GB memory.

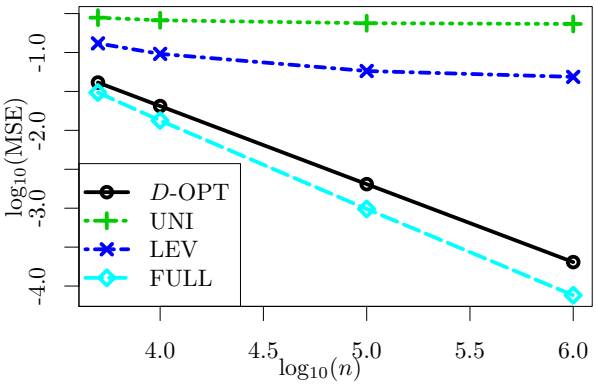
It is seen from Table 2 that the D-OPT IBOSS method compares favorably to the LEV method, both being more efficient than the full data method. Results for other cases are similar and thus are omitted.



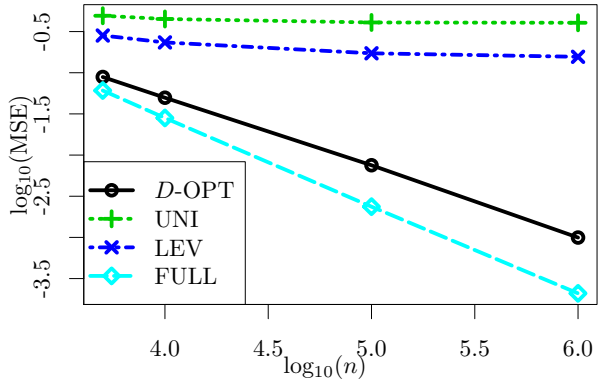
(a) Case 1: $\mathbf{z}_i$'s are normal.



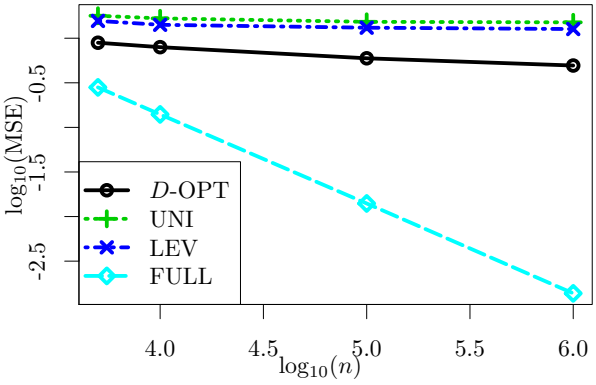
(b) Case 2: $\mathbf{z}_i$'s are lognormal.



(c) Case 3: $\mathbf{z}_i$'s are t_2.

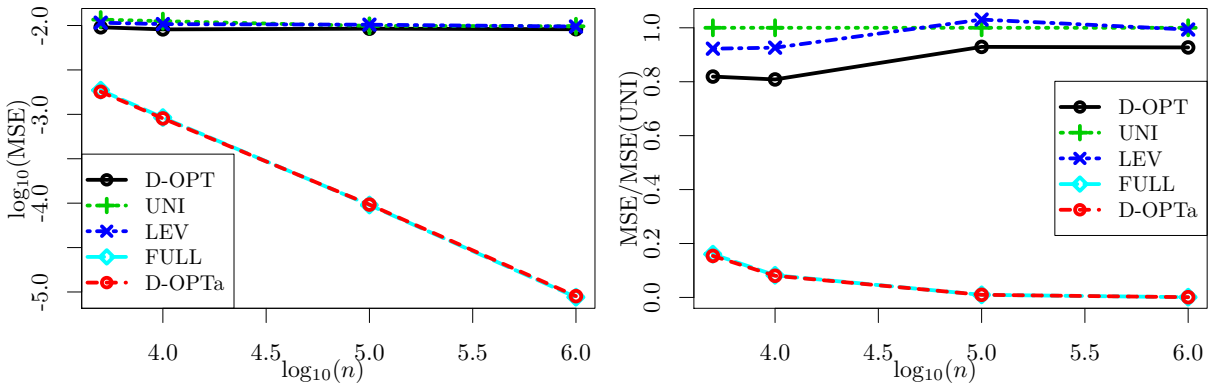


(d) Case 4: $\mathbf{z}_i$'s are a mixture.

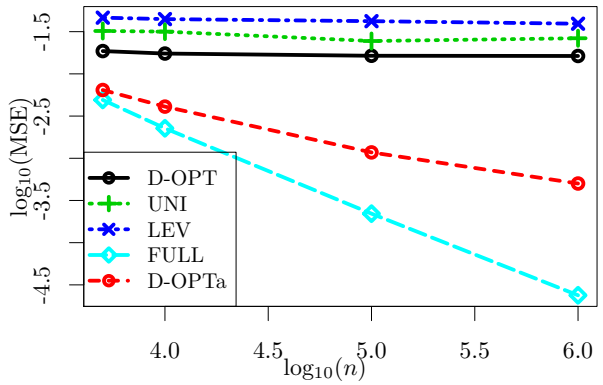


(e) Case 5: $\mathbf{z}_i$'s include interaction terms.

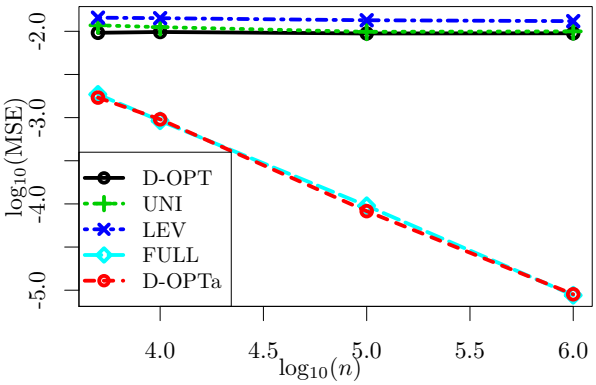
Figure 1: MSEs for estimating the slope parameter for five different distributions for the covariates $\mathbf{z}_i$. The subdata size k is fixed at $k = 1000$ and the full data size n changes. Logarithm with base 10 is taken of n and MSEs for better presentation of the figures except for the right panel of (a) in which MSEs are scaled so that MSEs for the UNI method are 1.



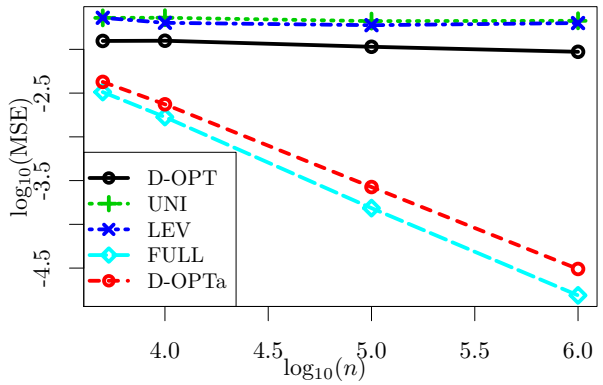
(a) Case 1: $\mathbf{z}_i$'s are normal.



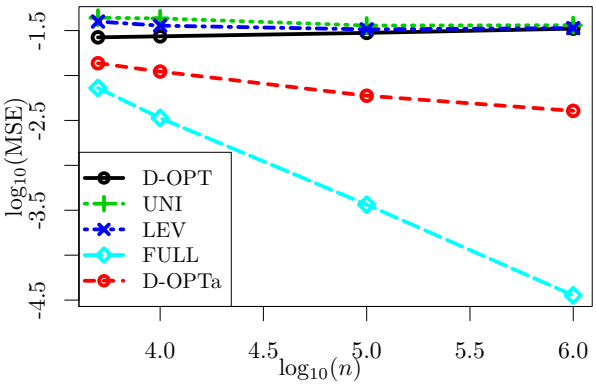
(b) Case 2: $\mathbf{z}_i$'s are lognormal.



(c) Case 3: $\mathbf{z}_i$'s are t_2.

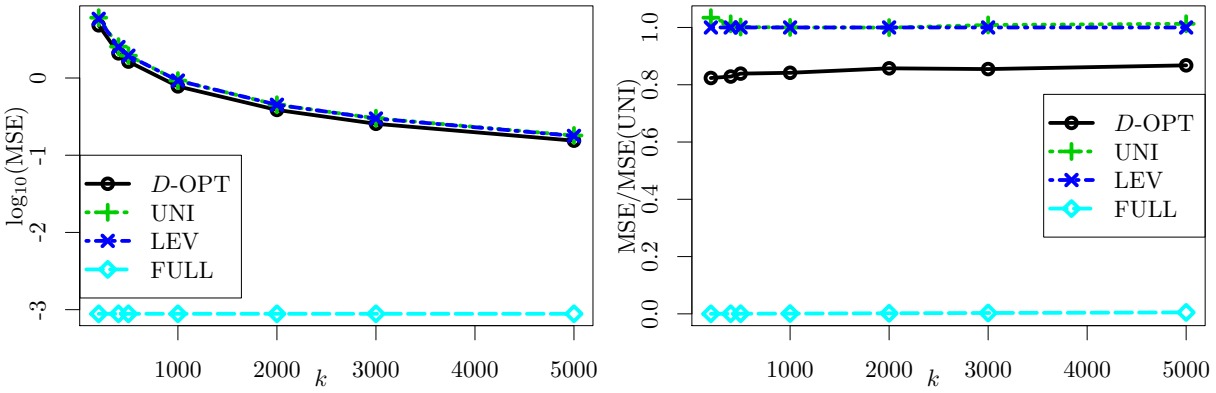


(d) Case 4: $\mathbf{z}_i$'s are a mixture.

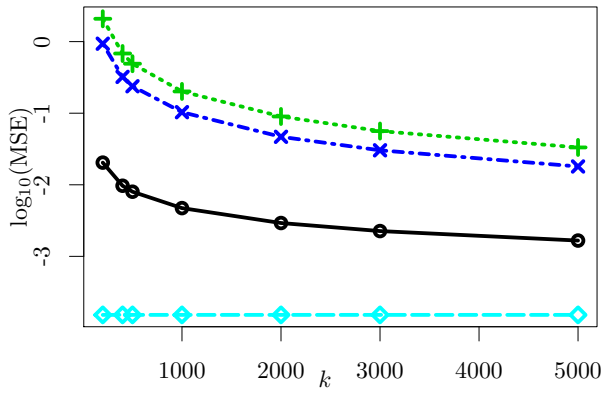


(e) Case 5: $\mathbf{z}_i$'s include interaction terms.

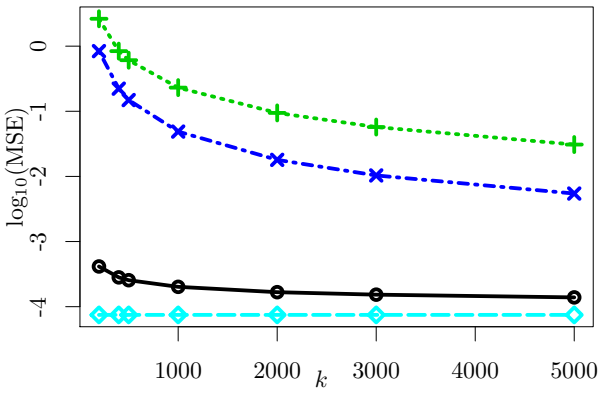
Figure 2: MSEs for estimating the intercept parameter for five different distributions for the covariates $\mathbf{z}_i$. The subdata size k is fixed at $k = 1000$ and the full data size n changes. Logarithm with base 10 is taken of n and MSEs for better presentation of the figures except for the right panel of (a) in which MSEs are scaled so that MSEs for the UNI method are 1.



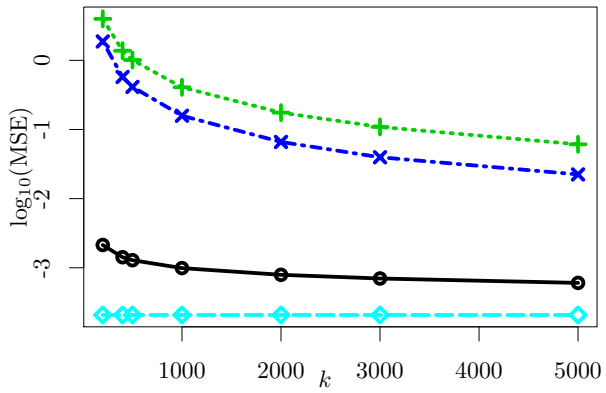
(a) Case 1: $\mathbf{z}_i$'s are normal.



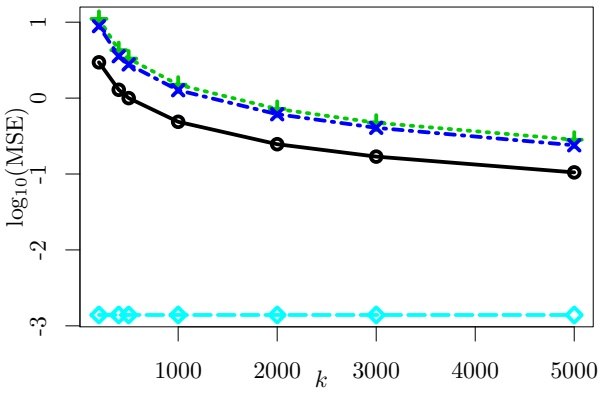
(b) Case 2: $\mathbf{z}_i$'s are lognormal.



(c) Case 3: $\mathbf{z}_i$'s are t_2.

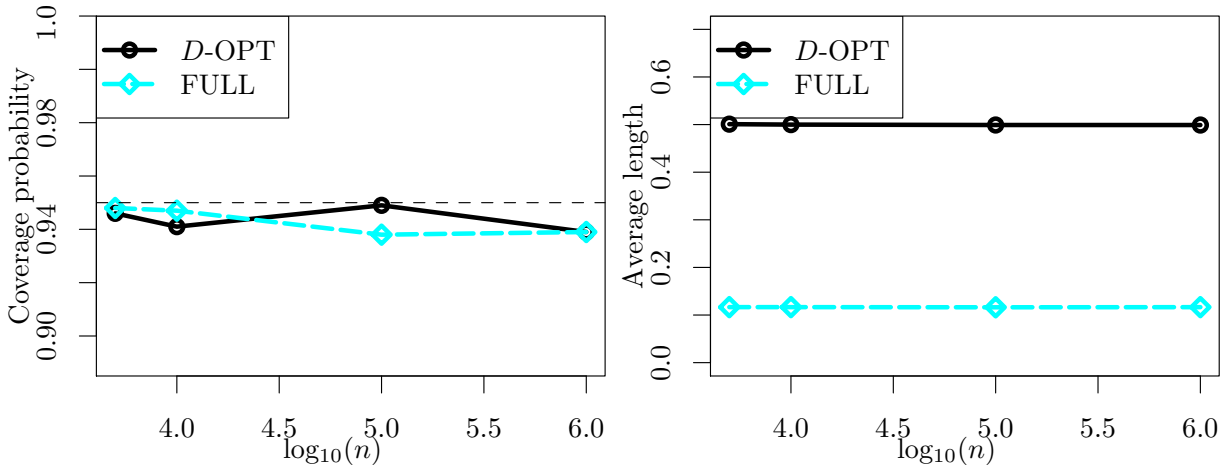


(d) Case 4: $\mathbf{z}_i$'s are a mixture.

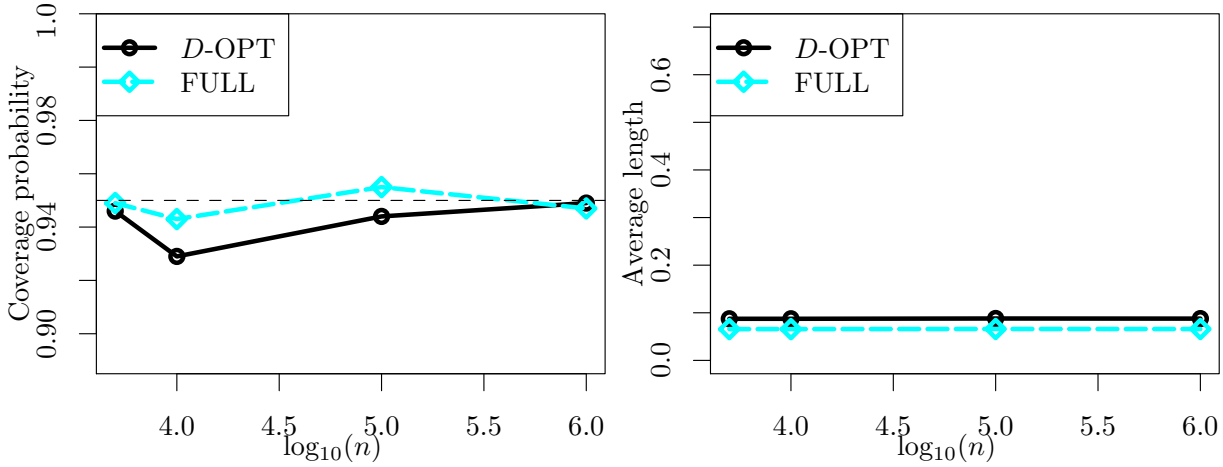


(e) Case 5: $\mathbf{z}_i$'s include interaction terms.

Figure 3: MSEs for estimating the slope parameter for five different distributions for the covariates $\mathbf{z}_i$. The full data size is fixed at $n = 10^6$ and the subdata size k changes. Logarithm with base 10 is taken of MSEs for better presentation of the figures except for the right panel of (a) in which MSEs are scaled so that MSEs for the UNI method are 1.



(a) Case 1: $\mathbf{z}_i$'s are normal.



(b) Case 4: $\mathbf{z}_i$'s are a mixture.

Figure 4: Empirical coverage probabilities and average lengths of 95% confidence intervals from the D-OPT IBOSS method and the full data method. The gray horizontal dashed line in the left panel is the intended coverage probability 0.95. The subdata size is fixed at $k = 10^3$.

Table 2: CPU times for different combinations of n and p with a fixed $k = 10^3$.

(a) CPU times for different n with $p = 500$

n	D-OPT	UNI	LEV	FULL
5×10^3	1.19	0.33	0.88	1.44
5×10^4	1.36	0.29	2.20	13.39
5×10^5	8.89	0.31	21.23	132.04

(b) CPU times for different p with $n = 5 \times 10^5$

p	D-OPT	UNI	LEV	FULL
10	0.19	0.00	1.94	0.21
100	1.74	0.02	4.66	6.55
500	9.30	0.31	21.94	132.47

5.2 Real data

In this section, we evaluate the performance of the proposed IBOSS approach on two real data examples.

5.2.1 Example 1: food intakes data

The first example is a data set obtained from the *Continuing Survey of Food Intakes by Individuals* (CSFII) that was published by the Human Nutrition Research Center, U.S. Department of Agriculture, Beltsville, Maryland (CSFII Reports No. 85-4 and No. 86-3). Part of the data set has been used in (Thompson *et al.*, 1992). It contains dietary intake and related information for $n = 1,827$ individuals, such as the intakes of calorie, fat, protein, and carbohydrate, as well as body mass index, age, etc. The size of this data set is not too big, and we can compare the IBOSS method to the full analysis. With this size of the data, we are also able to plot the full data in order to compare its pattern with that of the subdata selected by the IBOSS method. Interest is in examining the effects of the average intake levels of fat (z_1), protein (z_2), carbohydrate (carb, z_3), as well as body mass index (BMI, z_4) and age (z_5) on calorie intake, y. Thus $p = 5$. We fit the model

$$y = \beta_0 + \beta_1 z_1 + \beta_2 z_2 + \beta_3 z_3 + \beta_4 z_4 + \beta_5 z_5 + \varepsilon,$$

using both the D-OPT IBOSS method with $k = 10p = 50$ and the full data. Results are summarized in Table 3. The D-OPT IBOSS estimates for the slope parameters are not very different from those from the full data, and the signs of the estimates from the IBOSS method and from the full data are consistent. The standard errors for the IBOSS method, while larger than for the full data, are reasonably good in view of the small subdata size. The estimates for the intercept parameter show a larger difference, and the standard error for the IBOSS method is large. This agrees with the theoretical result that the intercept cannot be estimated precisely without a large subdata size. The D-OPT IBOSS method identifies the significant effects of fat, protein and carbohydrate intake levels on calorie intake. Based on the full data, the effect of BMI is near the boundary of significance at the 5% level, and is not identified as significant by the D-OPT IBOSS method.

Figure 5 gives scatter plots of calorie intake against each covariate for the full data of $n = 1,827$ with the D-OPT subdata of $k = 50$ marked. It is seen that the relationship

Table 3: Estimation results for the CSFII data. For the D-OPT IBOSS method, the subdata size is $k = 10p = 50$.

Parameter	D-OPT		FULL	
	Estimate	Std. Error	Estimate	Std. Error
Intercept	33.545	46.833	45.489	11.883
Age	-0.496	1.015	-0.200	0.234
BMI	-0.153	0.343	-0.521	0.224
Fat	8.459	0.405	9.302	0.115
Protein	5.080	0.386	4.254	0.127
Carb	3.761	0.106	3.710	0.035

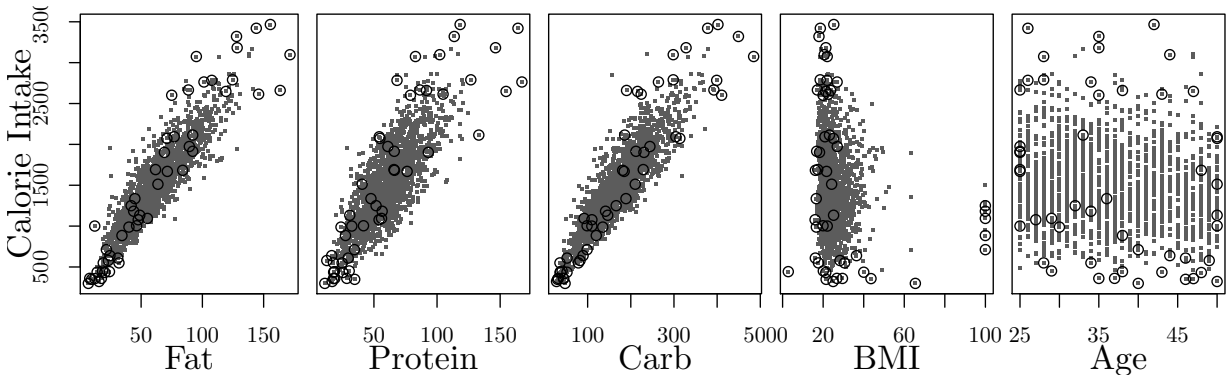


Figure 5: Scatter plots of calorie intake against each covariate for the full CSFII data (grey dots). The D-OPT subdata is labeled by ∘.

between the response and each covariate is similar for the subdata and the full data, especially for covariates fat, protein and carbohydrate. Also, there do not seem to be any extreme outliers in this data set.

To compare the IBOSS performance to that of the subsampling approaches, we compute the MSE for the vector of slope parameters for each method by using one thousand bootstrap samples. Each bootstrap sample is a random sample of size n from the full data using uniform sampling with replacement. For a bootstrap sample, we implement each subdata method to obtain the subdata estimate or implement the full data approach to obtain the full data estimate. The bootstrap MSEs are the empirical MSEs corresponding

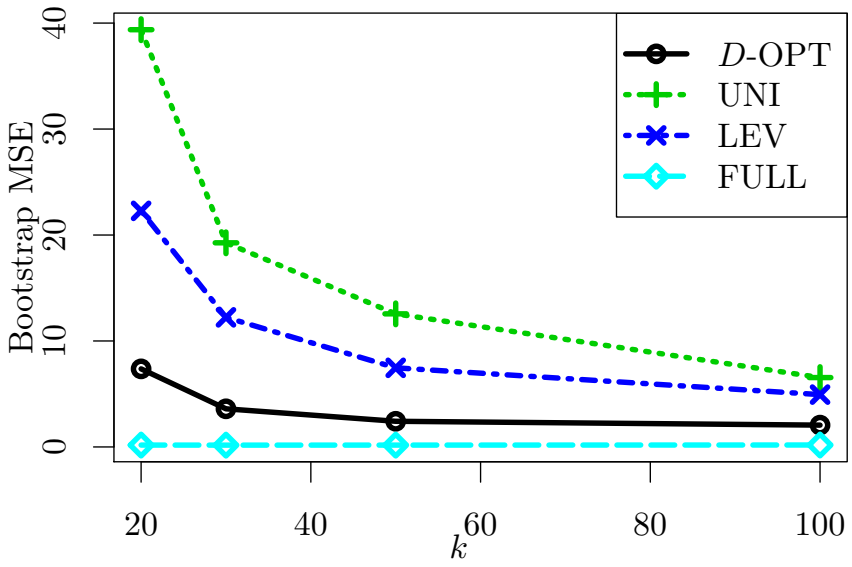


Figure 6: MSEs for estimating slope parameters for the CSFII data. They are computed from 1000 bootstrap samples.

to the 1,000 estimates. We do this for $k = 4p$, $6p$, $10p$ and $20p$. Figure 6 shows that the D-OPT IBOSS method dominates random subsampling-based methods. The full data approach is shown for comparison.

5.2.2 Example 2: chemical sensors data

In this example, we consider chemical sensors data collected to develop and test strategies to solve a wide variety of tasks, e.g., to develop algorithms for continuously monitoring or improving response time of sensory systems (Fonollosa *et al.*, 2015). The data were collected at the ChemoSignals Laboratory in the BioCircuits Institute, University of California San Diego. It contains the readings of 16 chemical sensors exposed to the mixture of Ethylene and CO at varying concentrations in air. Each measurement was constructed by the continuous acquisition of the sixteen-sensor array signals for a duration of about 12 hours without interruption. The concentration transitions were set at random times and to random concentration levels. Further information about the data set can be found in Fonollosa *et al.* (2015).

For illustration, we use the reading from the last sensor as the response and readings from other sensors as covariates. Since trace concentrations often have a lognormal distri-

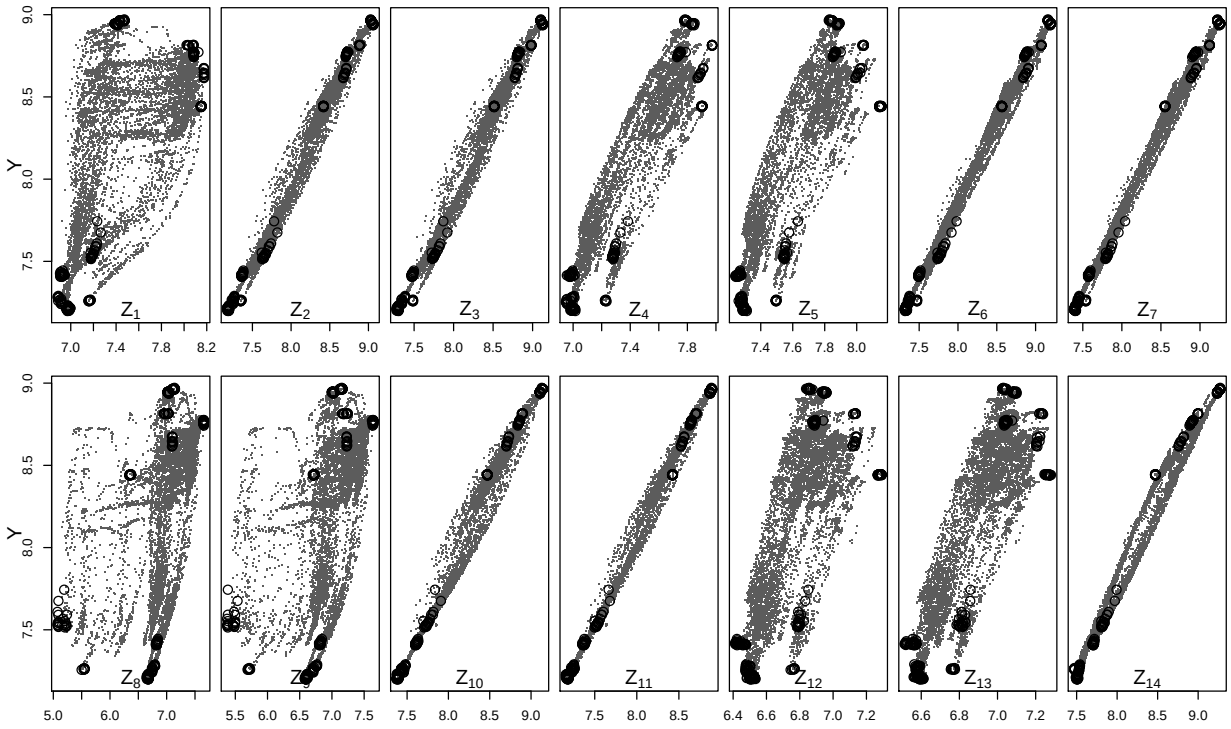


Figure 7: Scatter plots of a size $10,000$ simple random sample of the chemical sensors data (grey dots). The D-OPT subdata of size $k = 280$ is plotted as $\circ$.

bution (Goodson, 2011), we take a log-transformation of the sensors readings. Readings from the second sensor are not used in the analysis because about 20% of the values are negative for reasons unknown to us. Thus, there are $p = 14$ covariates in this example. In addition, we exclude the first 20,000 data points corresponding to less than 4 minutes of system run-in time. Thus, the full data used contain $n = 4,188,261$ data points.

Figure 7 gives scatter plots of the response variable against each covariate for a simple random sample of size $10,000$, with D-OPT subdata of $k = 280$ overlaid. Due to the size of the data, we cannot plot the full data in Figure 7. However, a simple random sample with a large sample size should be able to represent the overall pattern of the full data. It is seen that a linear model seems appropriate for the log-transformed readings and the relationship between the response and each covariate is similar for the subdata and the full data.

We also use bootstrap to calculate the MSEs of different estimators for estimating the slope parameters. As for the first example, we considered $k = 4p$, $6p$, $10p$ and $20p$ as

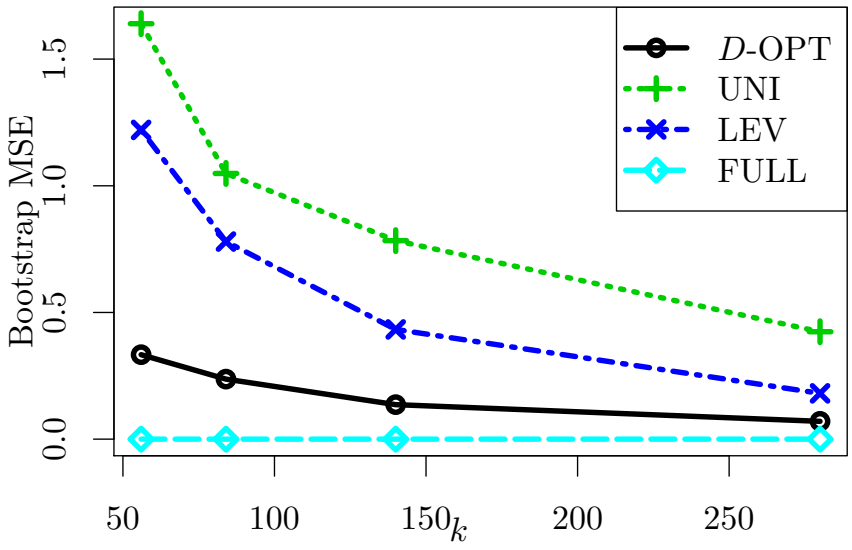


Figure 8: MSEs from 100 bootstrap samples for estimating slope parameters for the chemical sensors data.

subdata size for each method. Results computed from 100 bootstrap samples are plotted in Figure 8. The performance of the D-OPT IBOSS method lies between the full data approach and the subsampling based methods.

6 Concluding remarks

In this paper, we have developed a subdata selection method, IBOSS, in the context of big data linear regression problems. Using the framework for the IBOSS method, we have analyzed existing subsampling-based methods and derived a lower bound for covariance matrices of the resultant estimators. For the IBOSS method, we focused on D-optimality. After a theoretical characterization of the IBOSS subdata under D-optimality, we developed a computationally efficient algorithm to approximate the optimal subdata. Theoretical properties of the D-OPT IBOSS method have been examined in detail through asymptotic analysis, and its performance has been demonstrated by using simulated and real data.

There are important and unsolved questions that require future study. For example, while we only considered the D-optimality criterion, there are other optimality criteria with meaningful statistical interpretations and different inferential purposes. This includes

A-optimality, which seeks to minimize the average variance of estimators of regression coefficients, and c-optimality, which minimizes the variance of the best estimator of a pre-specified function of the model parameters. These optimality criteria may also be useful to develop efficient IBOSS methods.

Identifying informative subdata is important for extracting useful information from big data and more research is needed. We hope that this work will stimulate additional research in the direction suggested in this paper.

A Appendix

A.1 Proof of Theorem 1

We will use the following convexity result (cf. Nordström, 2011) in the proof of Theorem 1.

Lemma 1. *For any positive definite matrices* $\mathbf{B}_1$ *and* $\mathbf{B}_2$ *of the same dimension,*

$$\{\alpha\mathbf{B}_1 + (1-\alpha)\mathbf{B}_2\}^{-1} \leq \alpha\mathbf{B}_1^{-1} + (1-\alpha)\mathbf{B}_2^{-1} \tag{28}$$

in the Loewner ordering, where $0 \leq \alpha \leq 1$.

Proof of Theorem 1. The unbiasedness can be verified by direct calculation,

$$\mathrm{E}\{\tilde{\boldsymbol{\beta}}_L|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\} = \mathrm{E}_{\boldsymbol{\eta}_L}[\mathrm{E}_{\mathbf{y}}\{\tilde{\boldsymbol{\beta}}_L|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\}] = \mathrm{E}_{\boldsymbol{\eta}_L}(\boldsymbol{\beta}) = \boldsymbol{\beta}.$$

Let $\mathbf{W} = \mathrm{diag}(w_1\eta_{L1}, ..., w_n\eta_{Ln})$. The variance-covariance matrix of the sampling-based estimators can be written as

$$\begin{aligned}
\mathrm{V}\{\tilde{\boldsymbol{\beta}}_L|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\} =& \mathrm{E}_{\boldsymbol{\eta}_L}[\mathrm{V}_{\mathbf{y}}\{\tilde{\boldsymbol{\beta}}_L|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\}] + \mathrm{V}_{\boldsymbol{\eta}_L}[\mathrm{E}_{\mathbf{y}}\{\tilde{\boldsymbol{\beta}}_L|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\}] \\
=& \sigma^2\mathrm{E}_{\boldsymbol{\eta}_L}\left\{\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{X}\right)^{-1}\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}^2\mathbf{X}\right)\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{X}\right)^{-1}\right\} + \mathrm{V}_{\boldsymbol{\eta}_L}(\boldsymbol{\beta}) \\
=& \sigma^2\mathrm{E}_{\boldsymbol{\eta}_L}\left[\left\{\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{X}\right)\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}^2\mathbf{X}\right)^{-1}\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{X}\right)\right\}^{-1}\right] \\
\geq& \sigma^2\left[\mathrm{E}_{\boldsymbol{\eta}_L}\left\{\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{X}\right)\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}^2\mathbf{X}\right)^{-1}\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{X}\right)\right\}\right]^{-1}.
\end{aligned} \tag{29}$$

The last inequality is due to Lemma 1. Notice that $\mathbf{W}\mathbf{X}\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}^2\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{W} = \mathrm{pr}(\mathbf{W}\mathbf{X})$,

the orthogonal projection matrix onto the column space of $\mathbf{WX}$. Define

$$\mathbf{B}_{WX} = \begin{bmatrix} w_1\eta_{L1}\mathbf{x}_1^{\mathrm{T}} & & \\ & \ddots & \\ & & w_n\eta_{Ln}\mathbf{x}_n^{\mathrm{T}} \end{bmatrix}.$$

Notice that the column-space of $\mathbf{WX} = (w_1\eta_{L1}\mathbf{x}_1, ..., w_n\eta_{Ln}\mathbf{x}_n)^{\mathrm{T}}$ is contained in the column-space of $\mathbf{B}_{WX}$. Hence we have $\mathrm{pr}(\mathbf{WX}) \leq \mathrm{pr}(\mathbf{B}_{WX})$ in the Loewner ordering, i.e.,

$$\mathbf{WX}\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}^2\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{W} \leq \begin{bmatrix} \mathbf{x}_1^{\mathrm{T}}\left(\mathbf{x}_1\mathbf{x}_1^{\mathrm{T}}\right)^{-}\mathbf{x}_1 I(\eta_{L1}>0) & & \\ & \ddots & \\ & & \mathbf{x}_n^{\mathrm{T}}\left(\mathbf{x}_n\mathbf{x}_n^{\mathrm{T}}\right)^{-}\mathbf{x}_n I(\eta_{Ln}>0) \end{bmatrix}.$$

where $I()$ is the indicator function. From this result, it can be shown that

$$\mathbf{X}^{\mathrm{T}}\mathbf{WX}\left(\mathbf{X}^{\mathrm{T}}\mathbf{W}^2\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{WX} \leq \sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}I(\eta_{Li}>0). \tag{30}$$

For sampling with replacement,

$$P(\eta_{Li}>0|\mathbf{Z}) = 1-(1-\pi_i)^k = \pi_i\sum_{i=1}^{k}(1-\pi_i)^{i-1} \leq k\pi_i.$$

For sampling without replacement,

$$P(\eta_{Li}>0|\mathbf{Z}) = P(\eta_{Li}=1|\mathbf{Z}) = k\pi_i.$$

Thus, in either case, $P(\eta_{Li}>0|\mathbf{Z}) \leq k\pi_i$. Therefore,

$$P\{\eta_{Li}>0|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\} = \frac{P\{\eta_{Li}>0, I_\Delta(\boldsymbol{\eta}_L)=1|\mathbf{Z}\}}{P\{I_\Delta(\boldsymbol{\eta}_L)=1|\mathbf{Z}\}} \leq \frac{P(\eta_{Li}>0|\mathbf{Z})}{P\{I_\Delta(\boldsymbol{\eta}_L)=1|\mathbf{Z}\}} \leq \frac{k\pi_i}{P\{I_\Delta(\boldsymbol{\eta}_L)=1|\mathbf{Z}\}}. \tag{31}$$

Combining (29), (30) and (31), we have

$$\begin{aligned} \mathrm{V}\{\tilde{\boldsymbol{\beta}}_L|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\} &\geq \sigma^2\left[\mathrm{E}_{\boldsymbol{\eta}_L}\left\{\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}I(\eta_{Li}>0)\right\}\right]^{-1} \\ &= \sigma^2\left[\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}P\{\eta_{Li}>0|\mathbf{Z}, I_\Delta(\boldsymbol{\eta}_L)=1\}\right]^{-1} \\ &\geq \frac{\sigma^2 P\{I_\Delta(\boldsymbol{\eta}_L)=1|\mathbf{Z}\}}{k}\left\{\sum_{i=1}^{n}\pi_i\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}\right\}^{-1}. \end{aligned}$$

□

A.2 Proof of Theorem 2

Proof. Let $\breve{z}_{ij} = \{2z_{ij} - (z_{(n)j} + z_{(1)j})\}/(z_{(n)j} - z_{(1)j})$. Then we have,

$$\sum_{i=1}^{n} \delta_i \mathbf{x}_i \mathbf{x}_i^{\mathrm{T}} = k\mathbf{B}_3^{-1} \breve{\mathbf{M}}(\boldsymbol{\delta})(\mathbf{B}_3^{\mathrm{T}})^{-1}, \tag{32}$$

where

$$\breve{\mathbf{M}}(\boldsymbol{\delta}) = \begin{bmatrix} 1 & k^{-1}\sum_{i=1}^{n} \delta_i \breve{z}_{i1} & \dots & k^{-1}\sum_{i=1}^{n} \delta_i \breve{z}_{id} \\ k^{-1}\sum_{i=1}^{n} \delta_i \breve{z}_{i1} & k^{-1}\sum_{i=1}^{n} \delta_i \breve{z}_{i1}^2 & \dots & k^{-1}\sum_{i=1}^{n} \delta_i \breve{z}_{i1}\breve{z}_{ip} \\ \vdots & \vdots & \ddots & \vdots \\ k^{-1}\sum_{i=1}^{n} \delta_i \breve{z}_{ip} & k^{-1}\sum_{i=1}^{n} \delta_i \breve{z}_{i1}\breve{z}_{ip} & \dots & k^{-1}\sum_{i=1}^{n} \delta_i \breve{z}_{ip}^2 \end{bmatrix},$$

and

$$\mathbf{B}_3 = \begin{bmatrix} 1 & & & \\ -\frac{z_{(n)1}+z_{(1)1}}{z_{(n)1}-z_{(1)1}} & \frac{2}{z_{(n)1}-z_{(1)1}} & & \\ \vdots & & \ddots & \\ -\frac{z_{(n)p}+z_{(1)p}}{z_{(n)p}-z_{(1)p}} & & & \frac{2}{z_{(n)p}-z_{(1)p}} \end{bmatrix} \tag{33}$$

Note that $\breve{z}_{ij} \in [-1, 1]$ for all $i = 1, ..., n$ and $j = 1, ..., p$, which implies $k^{-1}\sum_{i=1}^{n} \delta_i \breve{z}_{ij}^2 \leq 1$ for all $1 \leq j \leq p$. Thus,

$$|\breve{\mathbf{M}}(\boldsymbol{\delta})| = \prod_{j=0}^{p} \lambda_j \leq \left(\frac{\sum_{j=0}^{p} \lambda_j}{p+1}\right)^{p+1} = \left(\frac{1 + \sum_{j=1}^{p} k^{-1}\sum_{i=1}^{n} \delta_i \breve{z}_{ij}^2}{p+1}\right)^{p+1} \leq 1, \tag{34}$$

where λ_j, $j = 0, 1, ..., p$ are eigenvalues of $\breve{\mathbf{M}}(\boldsymbol{\delta})$. From (32), (33) and (34),

$$\left|\sum_{i=1}^{n} \delta_i \mathbf{x}_i \mathbf{x}_i^{\mathrm{T}}\right| = k^{p+1}|\mathbf{B}_3|^{-2}|\breve{\mathbf{M}}(\boldsymbol{\delta})| \leq k^{p+1}\left|\prod_{j=1}^{p} \frac{2}{z_{(n)j} - z_{(1)j}}\right|^{-2} = \frac{k^{p+1}}{4^p}\prod_{j=1}^{p}(z_{(n)j} - z_{(1)j})^2.$$

If the subdata consists of the 2^p points $(a_1, \dots, a_p)^{\mathrm{T}}$ where $a_j = z_{(n)j}$ or $z_{(1)j}$, $j = 1, 2, ..., p$, each occurring equally often, then the $\boldsymbol{\delta}^{opt}$ corresponding to this subdata satisfies $\breve{\mathbf{M}}(\boldsymbol{\delta}) = \mathbf{I}$. This $\boldsymbol{\delta}^{opt}$ attains equality in (34) and corresponds therefore to D-optimal subdata. □

A.3 Proof of Theorem 3

Proof. As before, for $i = 1, ..., n$, $j = 1, ..., p$, let $z_{(i)j}$ be the ith order statistic for $z_{1j}, ..., z_{nj}$. For $l \neq j$, let $z_j^{(i)l}$ be the concomitant of $z_{(i)l}$ for z_j, i.e., if $z_{(i)l} = z_{sl}$ then $z_j^{(i)l} = z_{sj}$,

$i = 1, ..., n$. For the subdata obtained from Algorithm 1, let $\bar{z}_j^*$ and $\text{var}(z_j^*)$ be the sample mean and sample variance for covariate z_j. From Algorithm 1, the values z_j, $j = 1, ..., p$, in the subdata consist of $z_{(m)j}$, and $z_j^{(m)l}$, $l = 1, ...j-1, j+1, ..., p$, $m = 1, ..., r$, $n-r+1, ..., n$. Note that the subdata may not contain exactly the r smallest and r largest values for each covariate since some data points may be removed in processing each covariate. However, since r is fixed when n goes to infinity, this will not affect the final result. Therefore, for easy of presentation, we abuse the notation and write the range of values of m as $1, ..., r$, $n-r+1, ..., n$. The information matrix based on the subdata can be written as

$$(\mathbf{X}_{\text{D}}^*)^{\text{T}}\mathbf{X}_{\text{D}}^* = \mathbf{B}_4^{-1}\begin{bmatrix} k & \mathbf{0}^{\text{T}} \\ \mathbf{0} & (k-1)\mathbf{R} \end{bmatrix}(\mathbf{B}_4^{\text{T}})^{-1}, \tag{35}$$

where

$$\mathbf{B}_4 = \begin{bmatrix} 1 & & & \\ -\frac{\bar{z}_1^*}{\sqrt{\text{var}(z_1^*)}} & \frac{1}{\sqrt{\text{var}(z_1^*)}} & & \\ \vdots & & \ddots & \\ -\frac{\bar{z}_p^*}{\sqrt{\text{var}(z_p^*)}} & & & \frac{1}{\sqrt{\text{var}(z_p^*)}} \end{bmatrix}. \tag{36}$$

From (35) and (36),

$$|(\mathbf{X}_{\text{D}}^*)^{\text{T}}\mathbf{X}_{\text{D}}^*| = k|(k-1)\mathbf{R}|\prod_{j=1}^{p}\text{var}(z_j^*) \geq k(k-1)^p\lambda_{\min}^p(\mathbf{R})\prod_{j=1}^{p}\text{var}(z_j^*). \tag{37}$$

For each sample variance,

$$\begin{aligned}
(k-1)\text{var}(z_j^*) =& \sum_{i=1}^{k}\left(z_{ij}^* - \bar{z}_j^*\right)^2 \\
=& \left(\sum_{i=1}^{r} + \sum_{i=n-r+1}^{n}\right)\left(z_{(i)j} - \bar{z}_j^*\right)^2 + \sum_{l\neq j}\left(\sum_{i=1}^{r} + \sum_{i=n-r+1}^{n}\right)\left(z_j^{(i)l} - \bar{z}_j^*\right)^2 \\
\geq& \left(\sum_{i=1}^{r} + \sum_{i=n-r+1}^{n}\right)\left(z_{(i)j} - \bar{z}_j^{**}\right)^2 \\
=& \sum_{i=1}^{r}\left(z_{(i)j} - \bar{z}_j^{*l}\right)^2 + \sum_{i=n-r+1}^{n}\left(z_{(i)j} - \bar{z}_j^{*u}\right)^2 + \frac{r}{2}\left(\bar{z}_j^{*u} - \bar{z}_j^{*l}\right)^2 \\
\geq& \frac{r}{2}\left(\bar{z}_j^{*u} - \bar{z}_j^{*l}\right)^2
\end{aligned}$$

$$\geq \frac{r}{2}\left(z_{(n-r+1)j} - z_{(r)j}\right)^2 \tag{38}$$

where $\bar{z}_j^{**} = \left(\sum_{i=1}^{r} + \sum_{i=n-r+1}^{n}\right) z_{(i)j}/(2r)$, $\bar{z}_j^{*l} = \sum_{i=1}^{r} z_{(i)j}/r$, and $\bar{z}_j^{*u} = \sum_{i=n-r+1}^{n} z_{(i)j}/r$. From (38),

$$\mathrm{var}(z_j^*) \geq \frac{r(z_{(n)j} - z_{(1)j})^2}{2(k-1)} \left(\frac{z_{(n-r+1)j} - z_{(r)j}}{z_{(n)j} - z_{(1)j}}\right)^2. \tag{39}$$

Thus,

$$\begin{aligned}
|(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{X}_{\mathrm{D}}^*| \geq & k(k-1)^p \lambda_{\min}^p(\mathbf{R}) \prod_{j=1}^{p} \frac{r(z_{(n)j} - z_{(1)j})^2}{2(k-1)} \left(\frac{z_{(n-r+1)j} - z_{(r)j}}{z_{(n)j} - z_{(1)j}}\right)^2 \\
= & \frac{r^p}{2^p} k \lambda_{\min}^p(\mathbf{R}) \prod_{j=1}^{p} (z_{(n)j} - z_{(1)j})^2 \times \prod_{j=1}^{p} \left(\frac{z_{(n-r+1)j} - z_{(r)j}}{z_{(n)j} - z_{(1)j}}\right)^2.
\end{aligned}$$

This shows that

$$\frac{|(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{X}_{\mathrm{D}}^*|}{\frac{k^{p+1}}{4^p}\prod_{j=1}^{p}(z_{(n)j} - z_{(1)j})^2} \geq \frac{\lambda_{\min}^p(\mathbf{R})}{p^p} \times \prod_{j=1}^{p} \left(\frac{z_{(n-r+1)j} - z_{(r)j}}{z_{(n)j} - z_{(1)j}}\right)^2.$$

□

A.4 Proof of Theorem 4

Proof. From (35) and (36),

$$\mathrm{V}(\hat{\boldsymbol{\beta}}^{\mathrm{D}}|\mathbf{Z}) = \sigma^2\{(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{X}_{\mathrm{D}}^*\}^{-1} = \sigma^2 \mathbf{B}_4^{\mathrm{T}} \begin{bmatrix} \frac{1}{k} & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0} & \frac{1}{k-1}\mathbf{R}^{-1} \end{bmatrix} \mathbf{B}_4.$$

Thus

$$\mathrm{V}(\hat{\beta}_0^{\mathrm{D}}|\mathbf{Z}) = \sigma^2 \left(\frac{1}{k} + \frac{1}{k-1}\mathbf{u}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{u}\right), \tag{40}$$

and

$$\mathrm{V}(\hat{\beta}_j^{\mathrm{D}}|\mathbf{Z}) = \frac{\sigma^2}{k-1}\frac{(\mathbf{R}^{-1})_{jj}}{\mathrm{var}(z_j^*)}, \tag{41}$$

where $\mathbf{u} = \left\{ -\bar{z}_1^*/\sqrt{\mathrm{var}(z_1^*)}, ..., -\bar{z}_p^*/\sqrt{\mathrm{var}(z_p^*)} \right\}^{\mathrm{T}}$ and $(\mathbf{R}^{-1})_{jj}$ is the jth diagonal element of $\mathbf{R}^{-1}$.

From (40), $\mathrm{V}(\hat{\beta}_0^{\mathrm{D}}|\mathbf{Z}) \geq \sigma^2/k$ because $\mathbf{u}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{u} \geq 0$.

Denote the spectral decomposition of $\mathbf{R}$ as $\mathbf{R} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{\mathrm{T}}$. Since $\mathbf{\Lambda}^{-1} \leq \lambda_{\min}^{-1}(\mathbf{R})\mathbf{I}_p$, $\mathbf{R}^{-1} = \mathbf{V}\mathbf{\Lambda}^{-1}\mathbf{V}^{\mathrm{T}} \leq \mathbf{V}\lambda_{\min}^{-1}(\mathbf{R})\mathbf{I}_p\mathbf{V}^{\mathrm{T}} = \lambda_{\min}^{-1}(\mathbf{R})\mathbf{I}_p^{\mathrm{T}}$. Thus $\mathbf{R}_{jj}^{-1} \leq \lambda_{\min}^{-1}(\mathbf{R})$ for all j. From this fact, and (41) and (39), we have

$$\mathrm{V}(\hat{\beta}_j^{\mathrm{D}}|\mathbf{Z}) = \frac{\sigma^2}{k-1}\frac{(\mathbf{R}^{-1})_{jj}}{\mathrm{var}(z_j^*)} \leq \frac{4p\sigma^2}{k\lambda_{\min}(\mathbf{R})(z_{(n-r+1)j} - z_{(r)j})^2}. \tag{42}$$

Similarly, we have

$$\mathrm{V}(\hat{\beta}_j^{\mathrm{D}}|\mathbf{Z}) = \frac{\sigma^2}{k-1}\frac{(\mathbf{R}^{-1})_{jj}}{\mathrm{var}(z_j^*)} \geq \frac{4\sigma^2}{k\lambda_{\max}(\mathbf{R})(z_{(n)j} - z_{(1)j})^2}. \tag{43}$$

Here we utilize the following inequality

$$\mathrm{var}(z_j^*) \leq \frac{1}{k-1}\sum_{i=1}^{k}\left(z_{ij}^* - \frac{z_{(n)j} + z_{(1)j}}{2}\right)^2 \leq \frac{k}{4(k-1)}\left(z_{(n)j} - z_{(1)j}\right)^2, \tag{44}$$

where the last inequality is due to the fact $|z_{ij}^* - \frac{z_{(n)j}+z_{(1)j}}{2}| \leq \frac{z_{(n)j}-z_{(1)j}}{2}$ for all $i = 1, \ldots, k$.

□

A.5 Proof of Theorem 5

Proof. For (21), it is a direct result from (20).

For (22), we consider the five cases in the following. For the first case that r is fixed, from results in Theorems 2.8.1 and 2.8.2 in Galambos (1987), we have that

$$\frac{z_{(n-r+1)j} - z_{(r)j}}{z_{(n)j} - z_{(1)j}} = O_P(1) \quad \text{and} \quad \frac{z_{(n)j} - z_{(1)j}}{z_{(n-r+1)j} - z_{(r)j}} = O_P(1). \tag{45}$$

Combining (21) and (45), (22) follows.

For the second case when $r \to \infty$, $r/n \to 0$, and the support of F_j is bounded, (45) can be easily verified.

For the third case when the upper endpoint for the support of F_j is ∞ and the lower endpoint for the support of F_j is finite, and $r \to \infty$ slow enough such that (23) holds, if we can show that $z_{(n-r+1)j}/z_{(n)j} = 1 + o_P(1)$, then the result in (22) follows. Let $b_{n,j} = F_j^{-1}(1 - n^{-1})$. From Hall (1979), we only need to show that $z_{(n-r+1)j}/b_{n,j} = 1 + o_P(1)$ in order to show that $z_{(n-r+1)j}/z_{(n)j} = 1 + o_P(1)$. For this, from the proof of Theorem 1 of Hall (1979), it suffices to show that

$$\left[\frac{1 - F_j(b_{n,j})}{1 - F_j\{(1-\epsilon)b_{n,j}\}}\right]^{-1/2}\left[1 - \frac{r\{1 - F_j(b_{n,j})\}}{1 - F_j\{(1-\epsilon)b_{n,j}\}}\right] \to \infty,$$

which holds by directly applying the assumption in (23) and the fact that $r \to \infty$.

For the fourth case, it can be proved by using an approach similar to the one used for the third case. It can also be proved by noting that $z_{(r)j} = -(-z)_{(n-r+1)j}$, $z_{(1)j} = -(-z)_{(n)j}$, and the fact that the condition in (24) on $\mathbf{z}$ becomes the condition in (23) on $-\mathbf{z}$.

For the fifth case, it can be proved by combining the results in the third case and the fourth case. □

A.6 Proof of Theorem 6

Let σ_j and $\rho_{j_1 j_2}$ be the jth diagonal element of $\boldsymbol{\Phi}$ and entry (j_1, j_2) of $\boldsymbol{\rho}$, respectively, for $j, j_1, j_2 = 1, ..., p$. As described in the proof of Theorem 3, from Algorithm 1, the values z_j, $j = 1, ..., p$, in the subdata consist of $z_{(i)j}$, and $z_j^{(i)l}$, $l = 1, ...j-1, j+1, ..., p$, $i = 1, ..., r$, $n-r+1, ..., n$, where $z_j^{(i)l}$ are the concomitants for z_j.

Let $\mathbf{v} = (\mathbf{Z}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{1}$ and $\boldsymbol{\Omega} = (\mathbf{Z}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{Z}_{\mathrm{D}}^*$. Then

$$(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{X}_{\mathrm{D}}^* = \begin{bmatrix} k & \mathbf{v}^{\mathrm{T}} \\ \mathbf{v} & \boldsymbol{\Omega} \end{bmatrix}. \tag{46}$$

The jth diagonal element of $\boldsymbol{\Omega}$ is

$$\Omega_{jj} = \left(\sum_{i=1}^{r} + \sum_{i=n-r+1}^{n} \right) z_{(i)j}^2 + \sum_{l \neq j} \left(\sum_{i=1}^{r} + \sum_{i=n-r+1}^{n} \right) \left(z_j^{(i)l} \right)^2, \tag{47}$$

while entry (j_1, j_2), $j_1 \neq j_2$, is

$$\Omega_{j_1 j_2} = \left(\sum_{i=1}^{r} + \sum_{i=n-r+1}^{n} \right) \left(z_{(i)j_1} z_{j_2}^{(i)j_1} + z_{(i)j_2} z_{j_1}^{(i)j_2} \right) + \sum_{l \neq j_1 j_2} \left(\sum_{i=1}^{r} + \sum_{i=n-r+1}^{n} \right) z_{j_1}^{(i)l} z_{j_2}^{(i)l}. \tag{48}$$

The jth element of $\mathbf{v}$ is

$$v_j = \left(\sum_{i=1}^{r} + \sum_{i=n-r+1}^{n} \right) z_{(i)j} + \sum_{l \neq j} \left(\sum_{i=1}^{r} + \sum_{i=n-r+1}^{n} \right) z_j^{(i)l}. \tag{49}$$

Now we consider the two specific distributions in Theorem 6 and prove the corresponding results in (26) and (27).

A.6.1 Proof of equation (26) in Theorem 6

Proof. When $\mathbf{z}_i \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, using the results in Example 2.8.1 of Galambos (1987), we obtain

$$\begin{aligned} z_{(i)j} &= \mu_j - \sigma_j\sqrt{2\log n} + o_P(1), \quad i = 1, ..., r, \\ z_{(i)j} &= \mu_j + \sigma_j\sqrt{2\log n} + o_P(1), \quad i = n-r+1, ..., n. \end{aligned} \tag{50}$$

Using an approach similar to Example 5.5.1 of Galambos (1987), we obtain

$$\begin{aligned} z_j^{(i)l} &= \mu_j - \rho_{lj}\sigma_j\sqrt{2\log n} + O_P(1), \quad i = 1, ..., r, \\ z_j^{(i)l} &= \mu_j + \rho_{lj}\sigma_j\sqrt{2\log n} + O_P(1), \quad i = n-r+1, ..., n. \end{aligned} \tag{51}$$

Using (50) and (51), from (47), (48) and (49), we obtain that

$$\Omega_{jj} = 4r\log n\sigma_j^2\sum_{l=1}^{p}\rho_{lj}^2 + O_P(\sqrt{\log n}), \tag{52}$$

$$\Omega_{j_1j_2} = 4r\log n\sigma_{j_1}\sigma_{j_2}\sum_{l=1}^{p}\rho_{lj_1}\rho_{lj_2} + O_P(\sqrt{\log n}) \tag{53}$$

$$v_j = O_P(1), \tag{54}$$

respectively. From (52), (53) and (54), we have

$$\boldsymbol{\Omega} = 4r\log n\boldsymbol{\Phi}\boldsymbol{\rho}^2\boldsymbol{\Phi} + O_P(\sqrt{\log n}) \quad \text{and} \quad \mathbf{v} = O_P(1). \tag{55}$$

The variance,

$$\mathrm{V}(\hat{\boldsymbol{\beta}}^{\mathrm{D}}|\mathbf{X}) = \sigma^2\begin{bmatrix} k & \mathbf{v}^{\mathrm{T}} \\ \mathbf{v} & \boldsymbol{\Omega} \end{bmatrix}^{-1} = \frac{\sigma^2}{c}\begin{bmatrix} 1 & -\mathbf{v}^{\mathrm{T}}\boldsymbol{\Omega}^{-1} \\ -\boldsymbol{\Omega}^{-1}\mathbf{v} & c\boldsymbol{\Omega}^{-1} + \boldsymbol{\Omega}^{-1}\mathbf{v}\mathbf{v}^{\mathrm{T}}\boldsymbol{\Omega}^{-1} \end{bmatrix}, \tag{56}$$

where $c = k - \mathbf{v}^{\mathrm{T}}\boldsymbol{\Omega}^{-1}\mathbf{v} = k + O_P(1/\log n)$ and the second equality is from (55). Note that from (55) $\boldsymbol{\Omega}^{-1} = O_P(1/\log n)$, so

$$\begin{aligned} \boldsymbol{\Omega}^{-1} - (4r\log n\boldsymbol{\Phi}\boldsymbol{\rho}^2\boldsymbol{\Phi})^{-1} &= \boldsymbol{\Omega}^{-1}(4r\log n\boldsymbol{\Phi}\boldsymbol{\rho}^2\boldsymbol{\Phi} - \boldsymbol{\Omega})(4r\log n\boldsymbol{\Phi}\boldsymbol{\rho}^2\boldsymbol{\Phi})^{-1} \\ &= O_P\left(\frac{1}{\log n}\right)O_P(\sqrt{\log n})O\left(\frac{1}{\log n}\right) = O_P\left\{\frac{1}{(\log n)^{3/2}}\right\}. \end{aligned}$$

Thus

$$\boldsymbol{\Omega}^{-1} = \frac{1}{4r\log n}(\boldsymbol{\Phi}\boldsymbol{\rho}^2\boldsymbol{\Phi})^{-1} + O_P\left\{\frac{1}{(\log n)^{3/2}}\right\}. \tag{57}$$

Combining (46), (56) and (57), and using that $k = 2rp$

$$\mathrm{V}(\hat{\boldsymbol{\beta}}^{\mathrm{D}}|\mathbf{X}) = \sigma^2 \begin{bmatrix} \frac{1}{k} + O_P\left(\frac{1}{\log n}\right) & O_P\left(\frac{1}{\log n}\right) \\ O_P\left(\frac{1}{\log n}\right) & \frac{1}{4r\log n}(\boldsymbol{\Phi}\boldsymbol{\rho}^2\boldsymbol{\Phi})^{-1} + O_P\left\{\frac{1}{(\log n)^{3/2}}\right\} \end{bmatrix}.$$

□

A.6.2 Proof of equation (27) in Theorem 6

Proof. When $\mathbf{z}_i \sim LN(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Let $z_{ij} = \exp(U_{ij})$ with $\mathbf{U}_i = (U_{i1}, ..., U_{ip})^{\mathrm{T}} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. From (50),

$$\begin{aligned} z_{(i)j} &= \exp(U_{(i)j}) = \exp(-\sigma_j\sqrt{2\log n})O_P(1) = o_P(1), \quad i = 1, ..., r, \\ z_{(i)j} &= \exp(U_{(i)j}) = \exp(\sigma_j\sqrt{2\log n})\{e^{\mu_j} + o_P(1)\}, \quad i = n-r+1, ..., n. \end{aligned} \tag{58}$$

Without loss of generality, assume that $\rho_{lj} \geq 0$, $l, j = 1, ..., p$. From (51),

$$\begin{aligned} z_j^{(i)l} &= \exp(U_j^{(i)l}) = \exp(-\rho_{lj}\sigma_j\sqrt{2\log n})O_P(1) = o_P(1), \quad i = 1, ..., r, \\ z_j^{(i)l} &= \exp(U_j^{(i)l}) = \exp\{\sigma_j\sqrt{2\log n} - (1-\rho_{lj})\sigma_j\sqrt{2\log n} + \mu_j + O_P(1)\} \\ &= \exp(\sigma_j\sqrt{2\log n})o_P(1), \quad i = n-r+1, ..., n. \end{aligned} \tag{59}$$

Using (58) and (59), from (47), (48) and (49), we obtain that

$$\Omega_{jj} = r\exp(2\sigma_j\sqrt{2\log n})\{e^{2\mu_j} + o_P(1)\}, \tag{60}$$

$$\Omega_{j_1j_2} = 2r\exp\left\{(\sigma_{j_1} + \sigma_{j_2})\sqrt{2\log n}\right\}o_P(1), \tag{61}$$

$$v_j = r\exp(\sigma_j\sqrt{2\log n})\{e^{\mu_j} + o_P(1)\}. \tag{62}$$

From (46), (60)-(62), for $\mathbf{A}_n = \mathrm{diag}\Big\{1, \exp\left(\sigma_1\sqrt{2\log n}\right), ..., \exp\left(\sigma_p\sqrt{2\log n}\right)\Big\}$,

$$\mathbf{A}_n^{-1}(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{X}_{\mathrm{D}}^*\mathbf{A}_n^{-1} = \mathbf{A}_n^{-1}\begin{bmatrix} k & \mathbf{v}^{\mathrm{T}} \\ \mathbf{v} & \boldsymbol{\Omega} \end{bmatrix}\mathbf{A}_n^{-1} = \begin{bmatrix} k & r\mathbf{v}_1^{\mathrm{T}} \\ r\mathbf{v}_1 & r\mathbf{B}_5, \end{bmatrix} + o_P(1) \tag{63}$$

where $\mathbf{v}_1 = (e^{\mu_1}, ..., e^{\mu_p})^{\mathrm{T}}$ and $\mathbf{B}_5 = \mathrm{diag}(e^{2\mu_1}, ..., e^{2\mu_p})$. From (63),

$$\begin{aligned} \mathrm{V}(\mathbf{A}_n\hat{\boldsymbol{\beta}}^{\mathrm{D}}|\mathbf{X}) = \sigma^2\mathbf{A}_n\{(\mathbf{X}_{\mathrm{D}}^*)^{\mathrm{T}}\mathbf{X}_{\mathrm{D}}^*\}^{-1}\mathbf{A}_n &= \sigma^2\begin{bmatrix} k & r\mathbf{v}_1^{\mathrm{T}} \\ r\mathbf{v}_1 & r\mathbf{B}_5, \end{bmatrix}^{-1} + o_P(1) \\ &= \frac{2\sigma^2}{k}\begin{bmatrix} 1 & -\mathbf{u}^{\mathrm{T}} \\ -\mathbf{u} & p\boldsymbol{\Lambda} + \mathbf{u}\mathbf{u}^{\mathrm{T}}, \end{bmatrix} + o_P(1). \end{aligned}$$

□

A.7 Proof of results in Table 1

When the covariate has a t distribution, from Theorem 4, for simple linear model, the variance of the estimator of β_1 using the D-OPT IBOSS approach is of the same order as $(z_{(n)1} - z_{(1)1})^{-2}$. From Theorems 2.1.2 and 2.9.2 of Galambos (1987), we obtain that $z_{(n)1} - z_{(1)1} \asymp_P n^{1/\nu}$. Thus, the variance is of the order $n^{-2/\nu}$.

For the full data approach, the variance of the estimator of β_1 is of the same order as $(\sum_{i=1}^n z_{i1}^2)^{-1}$. When z_1 has a t distribution with degrees of freedom $\nu > 2$, from Kolmogorov's strong law of large numbers (SLLN), $\sum_{i=1}^n z_{i1}^2 = O(n)$ almost surely. If $\nu \leq 2$, $\mathrm{E}[\{z_{i1}^2\}^{1/(2/\nu+\alpha)}] < \infty$ for any $\alpha > 0$. Thus, from Marcinkiewicz-Zygmund SLLN (Theorem 2 of Section 5.2 of Chow and Teicher, 2003), $\sum_{i=1}^n z_{ij}^2 = o(n^{2/\nu+\alpha})$ almost surely for any $\alpha > 0$. This shows that the order of $(\sum_{i=1}^n z_{i1}^2)^{-1}$ is slower than $n^{-(2/\nu+\alpha)}$ for any $\alpha > 0$.

For the UNI approach, the lower bound for the variance of the estimator of β_1 is of the same order as $n(\sum_{i=1}^n z_{i1}^2)^{-1}$, which is of order $O(1)$ when $\nu > 2$ and is slower than $n^{2/\nu-1+\alpha}$ for any $\alpha > 0$ when $\nu \leq 2$.

For the intercept β_0, the variance of the estimator is of the same order as the inverse of the sample size used in each method.

References

Candes, E. and Tao, T. (2007). The dantzig selector: statistical estimation when p is much larger than n. *The Annals of Statistics* 2313–2351.

Chow, Y. S. C. and Teicher, H. (2003). *Probability Theory: Independence, Interchangeability, Martingales*. Springer, New York.

Drineas, P., Magdon-Ismail, M., Mahoney, M., and Woodruff, D. (2012). Faster approximation of matrix coherence and statistical leverage. *Journal of Machine Learning Research* **13**, 3475–3506.

Drineas, P., Mahoney, M., Muthukrishnan, S., and Sarlos, T. (2011). Faster least squares approximation. *Numerische Mathematik* **117**, 219–249.

Drineas, P., Mahoney, M. W., and Muthukrishnan, S. (2006). Sampling algorithms for l_2 regression and applications. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, 1127–1136. Society for Industrial and Applied Mathematics.

Fan, J. and Lv, J. (2008). Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **70**, 5, 849–911.

Fonollosa, J., Sheik, S., Huerta, R., and Marco, S. (2015). Reservoir computing compensates slow response of chemosensor arrays exposed to fast varying gas concentrations in continuous monitoring. *Sensors and Actuators B: Chemical* **215**, 618–629.

Galambos, J. (1987). *The asymptotic theory of extreme order statistics*. Florida: Robert E. Krieger.

Goodson, D. Z. (2011). *Mathematical methods for physical and analytical chemistry*. John Wiley & Sons.

Hadamard, J. (1893). Résolution d'une question relative aux déterminants. *Bull. sci. math* **17**, 1, 240–246.

Hall, P. (1979). On the relative stability of large order statistics. In *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 86, 467–475. Cambridge Univ Press.

Kiefer, J. (1959). Optimum experimental designs. *Journal of the Royal Statistical Society. Series B (Methodological)* **21**, 2, 272–319.

Lin, N. and Xie, R. (2011). Aggregated estimating equation estimation. *Statistics and Its Interface* **4**, 73–83.

Ma, P., Mahoney, M., and Yu, B. (2014). A statistical perspective on algorithmic leveraging. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 91–99.

Ma, P., Mahoney, M., and Yu, B. (2015). A statistical perspective on algorithmic leveraging. *Journal of Machine Learning Research* **16**, 861–911.

Ma, P. and Sun, X. (2015). Leveraging for big data regression. *Wiley Interdisciplinary Reviews: Computational Statistics* **7**, 1, 70–76.

Martínez, C. (2004). On partial sorting. Tech. rep., 10th Seminar on the Analysis of Algorithms.

Meinshausen, N., Yu, B., *et al.* (2009). Lasso-type recovery of sparse representations for high-dimensional data. *The Annals of Statistics* **37**, 1, 246–270.

Musser, D. R. (1997). Introspective sorting and selection algorithms. *Software: Practice and Experience* **27**, 8, 983–993.

Nordström, K. (2011). Convexity of the inverse and Moore–Penrose inverse. *Linear Algebra and its Applications* **434**, 6, 1489–1512.

R Core Team (2015). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Schifano, E. D., Wu, J., Wang, C., Yan, J., and Chen, M.-H. (2016). Online updating of statistical inference in the big data setting. *Technometrics* **58**, 3, 393–403.

Stroustrup, B. (1986). *The C++ programming language*. Pearson Education India.

Thompson, E. E., Sowers, M., Frongillo, E., and Parpia, B. (1992). Sources of fiber and fat in diets of u.s. women aged 19 to 50: implications for nutrition education and policy. *American Journal of Public Health* **82**, 695–702.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)* 267–288.

Wang, H., Zhu, R., and Ma, P. (2017). Optimal subsampling for large sample logistic regression. *Journal of the American Statistical Association* **0**, ja, 0–0.

Supplementary Material for "Information-Based Optimal Subdata Selection for Big Data Linear Regression"

We present additional numerical results about the performance of the IBOSS method.

S.1 Predictive performance

In this section, we investigate the performance of IBOSS in predicting the mean response for a given setting of covariates. We focus on the mean squared prediction error (MSPE),

$$\text{MSPE} = E[\{E(y_{new}) - \hat{y}_{new}\}^2] = E[\{\mathbf{x}_{new}^{\mathrm{T}}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})\}^2]. \tag{S.1}$$

Note that the mean squared prediction error for predicting a future response is

$$E\{(y_{new} - \hat{y}_{new})^2\} = E[\{y_{new} - E(y_{new})\}^2] + E[\{E(y_{new}) - \hat{y}_{new}\}^2] = \sigma^2 + \text{MSPE}, \tag{S.2}$$

and the variance of y_{new}, σ^2, cannot be reduced by choosing a better subdata or a larger subdata sample size k. Thus it is reasonable to focus on the MSPE in (S.1) to evaluate the performance of IBOSS. For prediction, the estimation of β_0 is also important, so we use $\hat{\beta}_0^{Da} = \bar{y} - \bar{\mathbf{z}}^{\mathrm{T}}\hat{\boldsymbol{\beta}}_1^{\mathrm{D}}$ as indicated in the paper.

We use the same five cases considered in the paper to generate full data sets. In addition, we consider another case, Case 6, in which the covariates are from a multivariate t distribution with degrees of freedom $\nu = 1$. This is a case often used in evaluating the performance of the LEV method.

Case 6. $\mathbf{z}_i$'s have a multivariate t distribution with degrees of freedom $\nu = 1$, i.e., $\mathbf{z}_i \sim t_1(\mathbf{0}, \boldsymbol{\Sigma})$.

For each case, we implement different methods to obtain parameter estimates, and then generate a new sample of size 5,000 to calculate the MSPEs. The simulation is repeated 1,000 times and empirical MSPEs are calculated. Figure S.1 presents plots of the $\log_{10}$ of the MSPEs against $\log_{10}(n)$. For prediction, the relative performance of IBOSS compared

with other methods are similar to that of parameter estimation. That is, the D-OPT IBOSS method uniformly dominates the subsampling-based methods UNI and LEV, and its advantage is more significant if the tail of the covariate distribution is heavier. Specifically for Case 6, it is seen that the performance of D-OPT IBOSS is almost identical to that of the full data approach, and LEV significantly outperforms the UNI.

S.2 Column permutation

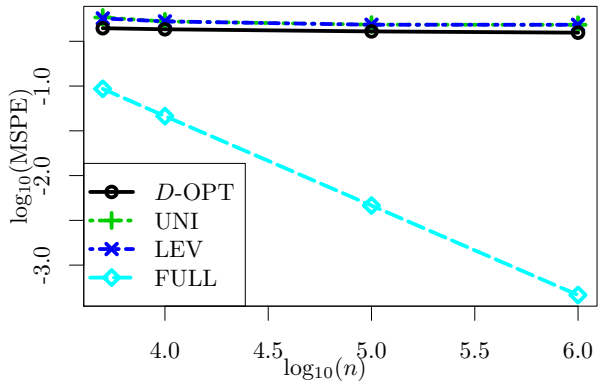
In this section, we provide numerical results accessing the effect of column permutation on the IBOSS method. To differentiate the effect of each column in the covariate matrix, we change the covariance matrix $\mathbf{\Sigma}$ such that $\Sigma_{ij} = 0.5^{|i-j|}$ if $i \neq j$, and $\Sigma_{ij} = 1 + 3(i-1)/p$ if $i = j$, $i, j = 1, ..., 50$. With this setup, the correlation structure for the covariates is unexchangeable and variances for different columns are different. Using this covariance matrix, we generate covariates $\mathbf{z}_i$'s according to Case 5 in Section 5.1 of the paper. The IBOSS method is applied with the original order of covariate columns as well as with a single random permutation of covariate columns. Results are presented in Figure S.2. It is seen that the performances of IBOSS for the two approaches are very similar. This agrees with the theoretical results.

S.3 Interaction model

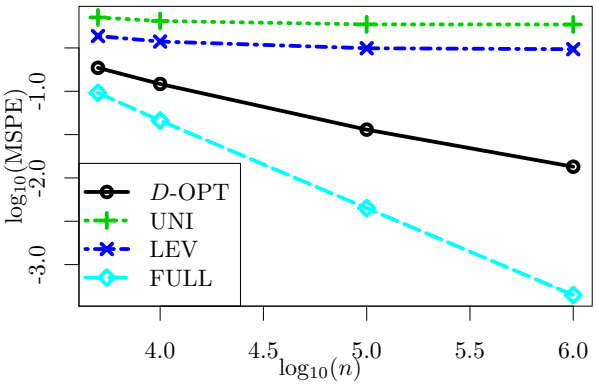
In this section, we consider a case that the true model contains all the main effects and all the pairwise interaction terms. However, only the main effects are used in selecting subdata. Data are generated from the following linear model,

$$y_i = \beta_0 + \sum_{j=1}^{10} z_{ij}\beta_j + \sum_{j_1 \neq j_2}^{10} z_{ij_1} z_{ij_2} \beta_{j_1 j_2} + \varepsilon_i, \quad i = 1, ..., n, \tag{S.3}$$

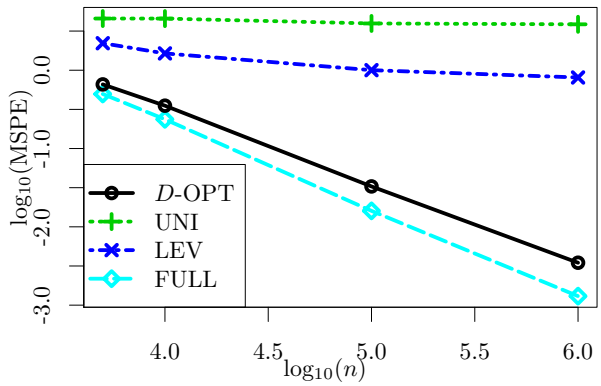
where the true value of regression coefficients are $\beta_j = \beta_{j_1 j_2} = 1$ for $j, j_1, j_2 = 1, ..., 10$, and ε_i's are i.i.d. $N(0, 9)$. Two different distributions are considered to generate covariates $\mathbf{z}_i$'s: one is a multivariate normal distribution $\mathbf{z}_i \sim N(\mathbf{0}, \mathbf{\Sigma}_{10\times 10})$ and the other is a multivariate lognormal distribution $\mathbf{z}_i \sim LN(\mathbf{0}, \mathbf{\Sigma}_{10\times 10})$, where $\mathbf{\Sigma}_{10\times 10}$ is a 10 by 10 covariance matrix



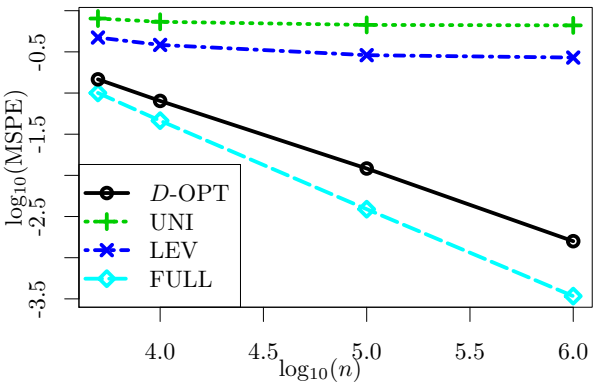
(a) Case 1: $\mathbf{z}_i$'s are normal.



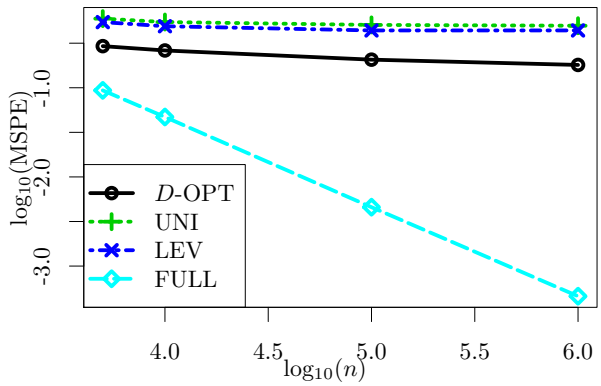
(b) Case 2: $\mathbf{z}_i$'s are lognormal.



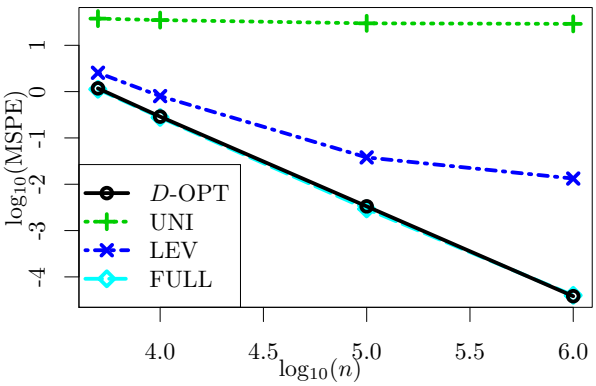
(c) Case 3: $\mathbf{z}_i$'s are t_2.



(d) Case 4: $\mathbf{z}_i$'s are a mixture.

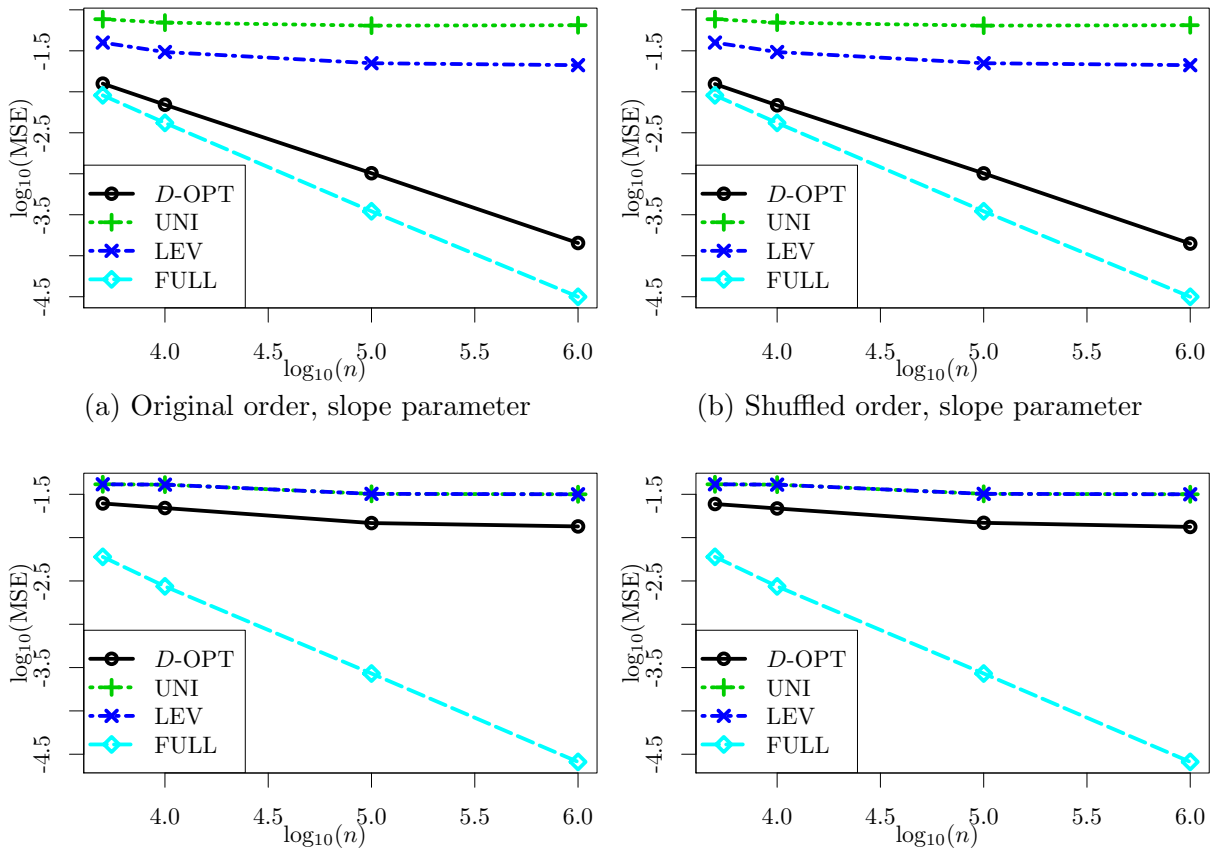


(e) Case 5: $\mathbf{z}_i$'s include interaction terms.



(f) Case 6: $\mathbf{z}_i$'s are t_1.

Figure S.1: MSPEs for predicting mean responses for six different distributions of the covariates $\mathbf{z}_i$. The subdata size k is fixed at $k = 1000$ and the full data size n changes. Logarithm with base 10 is taken of n and MSPEs for better presentation of the figures.



(a) Original order, slope parameter

(b) Shuffled order, slope parameter

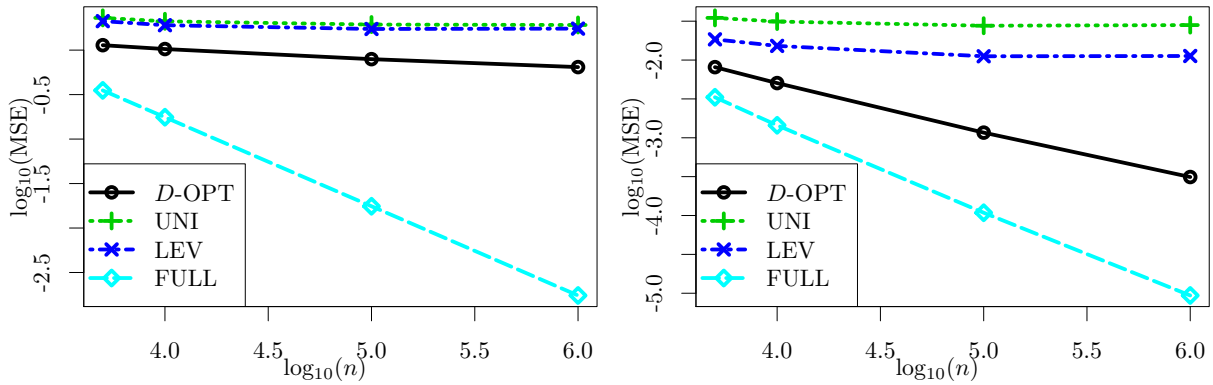
(c) Original order, intercept parameter

(d) Shuffled order, intercept parameter

Figure S.2: MSEs for estimating the slope parameter (top panel) and the intercept parameter (bottom panel) with different orders of the covariate columns. The left panel presents results with the original order of covariate columns and the right panel presents results with the randomly shuffled order of covariate columns. The subdata size k is fixed at $k = 1000$ and the full data size n changes. Logarithm with base 10 is taken of n and MSEs for better presentation of the figures.

with $\Sigma_{ij} = 0.5^{I(i\neq j)}$, for $i, j = 1, ..., 10$. In selecting subdata, only the main effects are used. The interaction terms are not used in subdata selection but are used in parameter estimation.

Figure S.3 presents the MSEs for estimating the slope parameters, which are calculated from 1000 iterations of the simulation. It is seen that IBOSS is still the most efficient method among subdata-based methods for both of the distributions.



(a) Case 1: $\mathbf{z}_i$'s are normal. (b) Case 2: $\mathbf{z}_i$'s are lognormal.

Figure S.3: MSEs for estimating the slope parameter for two different distributions of the covariates $\mathbf{z}_i$. The subdata size k is fixed at $k = 1000$ and the full data size n changes. Logarithm with base 10 is taken of n and MSEs for better presentation of the figures.

S.4 Nonlinear relationships

In this section, we consider the scenario that true relationships between the response and the covariates are nonlinear, and transformations cannot linearize the relationships, i.e., a finite-dimensional linear model cannot be correct. We consider the following two models

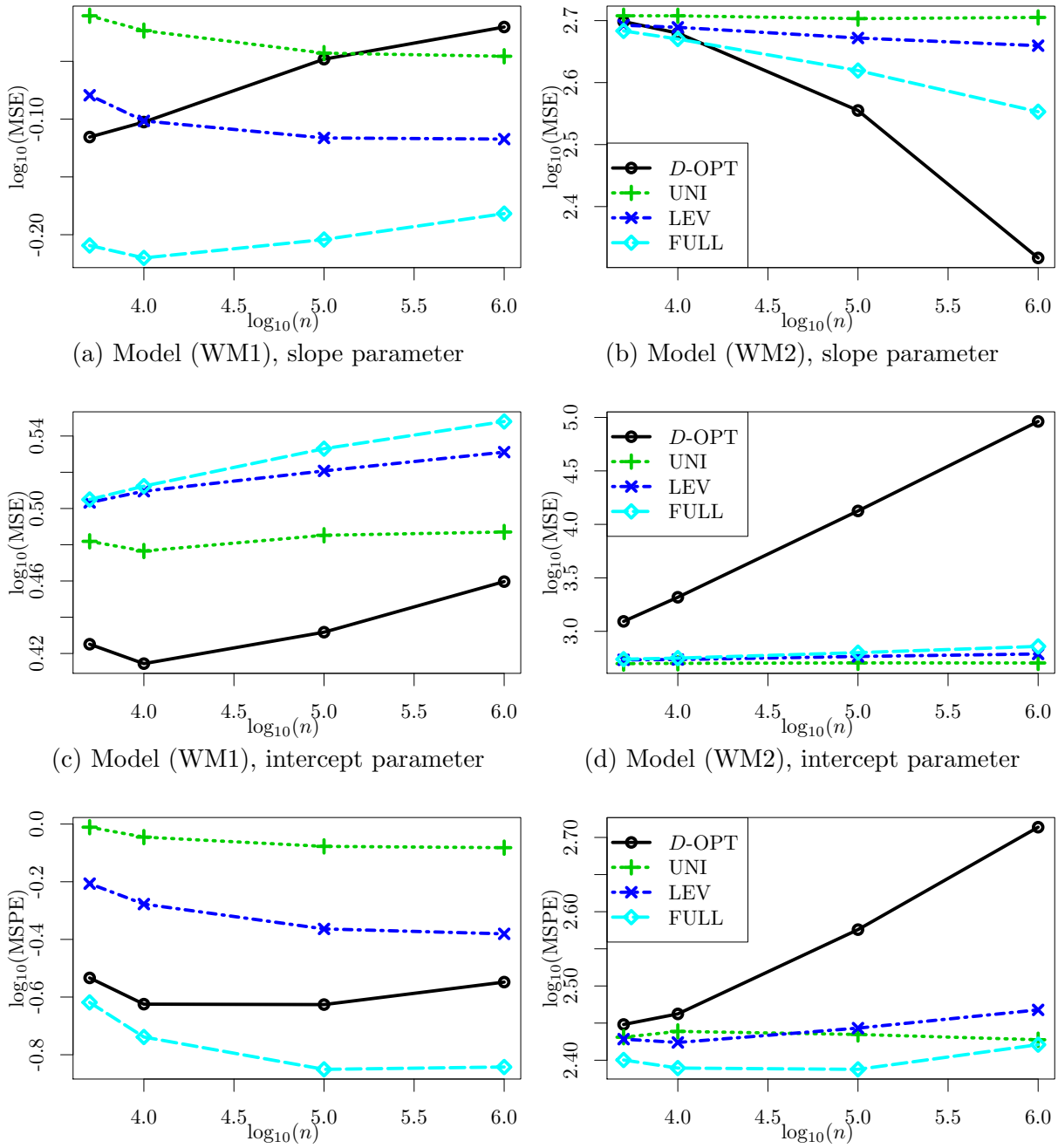
$$y_i = \beta_0 + \sum_{j=1}^{p-1} z_{ij}\beta_j + \frac{3e^{z_{ip}^{(t)}}}{1 + e^{z_{ip}^{(t)}}} + \varepsilon_i, \quad i = 1, ..., n, \tag{WM1}$$

$$y_i = \beta_0 + \sum_{j=1}^{p-1} z_{ij}\beta_j + 30 \log\left(1 + e^{z_{ip}^{(t)}}\right) + \varepsilon_i, \quad i = 1, ..., n, \tag{WM2}$$

where $z_{ip}^{(t)} = z_{ip} I(z_{ip} \leq 100) + 100 I(z_{ip} > 100)$. Covariates and parameter setups are the same as those of Case 4 for the mixture distribution. Although full data are generated

from nonlinear model (WM1) or (WM2), the linear main effects model is used for subdata selection and analysis.

Figure S.4 presents plots of the $\log_{10}$ of the MSEs of estimating the slope parameter and the intercept parameter against $\log_{10}(n)$, and plots of the $\log_{10}$ of the MSPEs of predicting the mean response. It is seen that, including the full data approach, no method dominates others and larger sample sizes do not necessarily mean more accurate results. When the underlying model is incorrect, the problem is very complicated and there is no simple answer to which method will produce satisfactory results. We present the numerical studies here to show that IBOSS does not always produce the worst results for this scenario, but we have no intention to state that the IBOSS works better than other methods.



(a) Model (WM1), slope parameter

(b) Model (WM2), slope parameter

(c) Model (WM1), intercept parameter

(d) Model (WM2), intercept parameter

(e) Model (WM1), prediction

(f) Model (WM2), prediction

Figure S.4: MSEs for estimating the slope parameter (top row), MSEs for estimating the intercept parameter (middle row), and MSPEs for predicting the mean response (bottom row) when true models are nonlinear. The left column is for model (WM1) and the right column is for model (WM2). The subdata size k is fixed at $k = 1000$ and the full data size n changes. Logarithm with base 10 is taken of n and MSEs for better presentation of the figures.

S.5 Accuracy-cost tradeoff of the IBOSS method

In this section, we provide additional results showing the accuracy-cost tradeoff of the IBOSS method. Full data of size $n = 5 \times 10^6$ are generated using the same setup of Case 1. The IBOSS method is implemented with subdata sample sizes of $k = 10^2, 10^3, 10^4, 10^5$ and 10^6, and the average CPU times and MSEs are calculated from 100 repetitions of the simulation. Results are reported in Figure S.5. It is seen that as the required CPU time increases, the MSE decreases, which indicates a clear tradeoff between computational cost and estimation accuracy for the IBOSS method. However, as the CPU time increases, the MSE can drop sharply. For example, when the CPU time increases from 6.4976 seconds (corresponding to $k = 10^2$) to 7.0839 seconds, the MSE decreases from 13.57091 to 0.00786855. Thus the IBOSS has the advantage to significantly increase the estimation accuracy with little increase in computational cost.

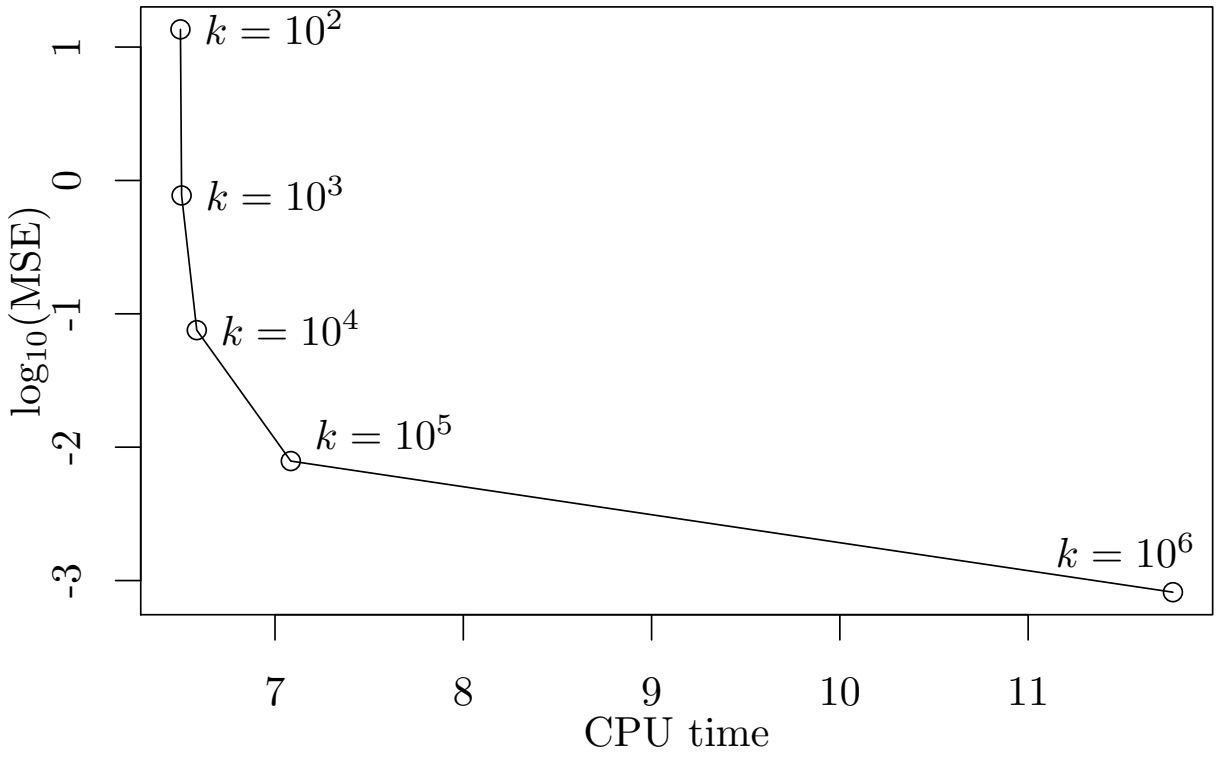
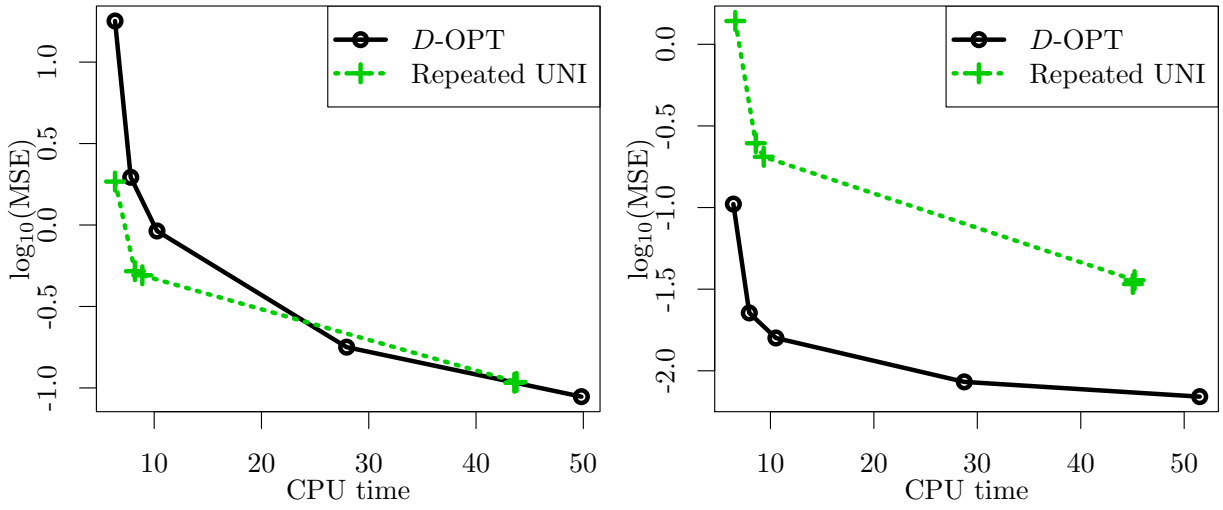


Figure S.5: Average CPU times and MSEs for different subdata sample size k when the covariates are from a multivariate normal distribution. The full data size is set to $n = 5 \times 10^6$ with a dimension $p = 50$.

We perform additional experiments to further investigate the accuracy-cost tradeoff of the IBOSS for both large n and large p, and draw comparisons with the performance of repeating the UNI method. Full data are generated with $n = 5 \times 10^5$ and $p = 500$, and subdata of sizes $k = 10^3, 5 \times 10^3, 10^4$, 5×10^4, and 10^5 are taken using the IBOSS method or the UNI method. For the UNI method, it is repeated multiple times so that it consumes similar CPU times to the IBOSS method, and the average of the estimates from all repetitions are used as the final estimate. Figure S.6 presents the results when the covariates are from the multivariate normal distribution (Case 1) and the mixture distribution (Case 4) described in Section 5 of the main paper. The average CPU times and MSEs for the slope parameters are calculated from 100 repetitions of the simulation. For Case 1 with multivariate normal covariates, the repeated UNI method may produce smaller MSEs compared with the IBOSS method using similar CPU times. However, the differences are not very significant compared with the advantage of the IBOSS method for Case 4, in which the covariate distribution has a heaver tail.



(a) Case 1: $\mathbf{z}_i$'s are normal.

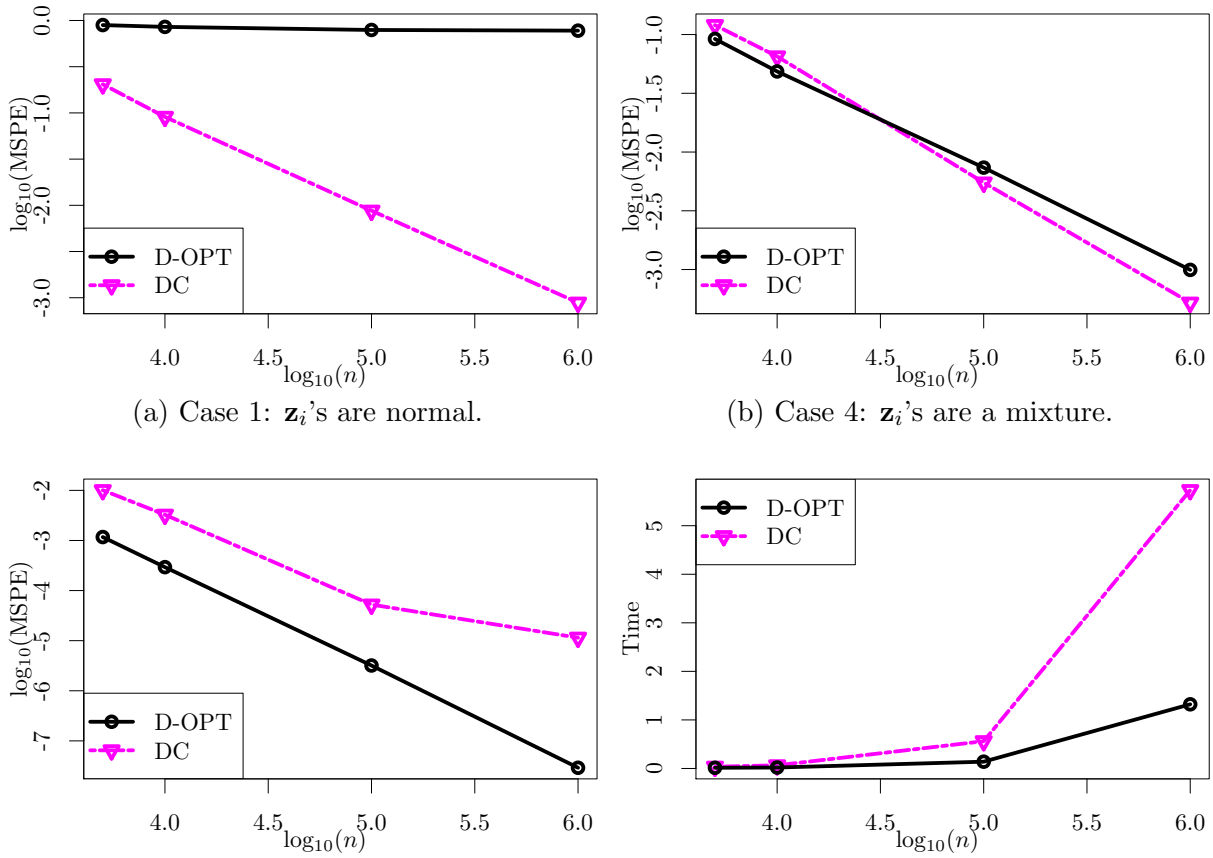
(b) Case 4: $\mathbf{z}_i$'s are a mixture.

Figure S.6: MSEs for different CPU times when the covariates are from a multivariate normal distribution (a) and a mixture distribution (b). The full data size is set to $n = 5 \times 10^5$ with dimension $p = 500$. Subdata sample size are $k = 10^3, 5 \times 10^3, 10^4$, 5×10^4, and 10^5.

S.6 Comparison with the divide-and-conquer method

In this section, we provide numerical results comparing the IBOSS method and the divide-and-conquer (DC) method proposed in Section 4.3 of Battey *et al.* (2015). The DC method divide the full data into S subdata sets (The notation k is used in Battey *et al.* (2015); we use S here because k is used to denote the subdata size.), and the ordinary least squares estimate, say $\hat{\boldsymbol{\beta}}_s$, is calculated for each subdata. The DC estimate is the average of $\hat{\boldsymbol{\beta}}_s$'s, i.e., $\bar{\boldsymbol{\beta}} = S^{-1}\sum_{s=1}^{S}\hat{\boldsymbol{\beta}}_s$. We choose $S = \lfloor n^{1/4} \rfloor$. In our implementation, if n/S is not an integer, the last subdata will have a sample size of $n - \lfloor n/S \rfloor * (S-1)$.

Figure S.7 gives the average CPU times and MSEs for the slope parameters with dimension $p = 50$ and different full data size n, with choices of $5 \times 10^3, 10^4, 10^5$, and 10^6. The average CPU times and MSEs are calculated from 100 repetitions of the simulation. It is seen that the relative performances of estimation efficiency between the IBOSS D-OPT method and the DC method depend on the covariate distribution. The DC method is better when covariates are normally distributed; the IBOSS D-OPT method and the DC method perform similarly when the covariate has a mixture distribution; the IBOSS D-OPT dominates the DC method when the covariate has a t_1 distribution. In terms of computational cost in Figure S.7 (d), the IBOSS D-OPT is more efficient than the DC method especially for large values of n. Note that the CPU times for either the DC method or the IBOSS D-OPT method do not depend on the covariate distribution.



(a) Case 1: $\mathbf{z}_i$'s are normal.

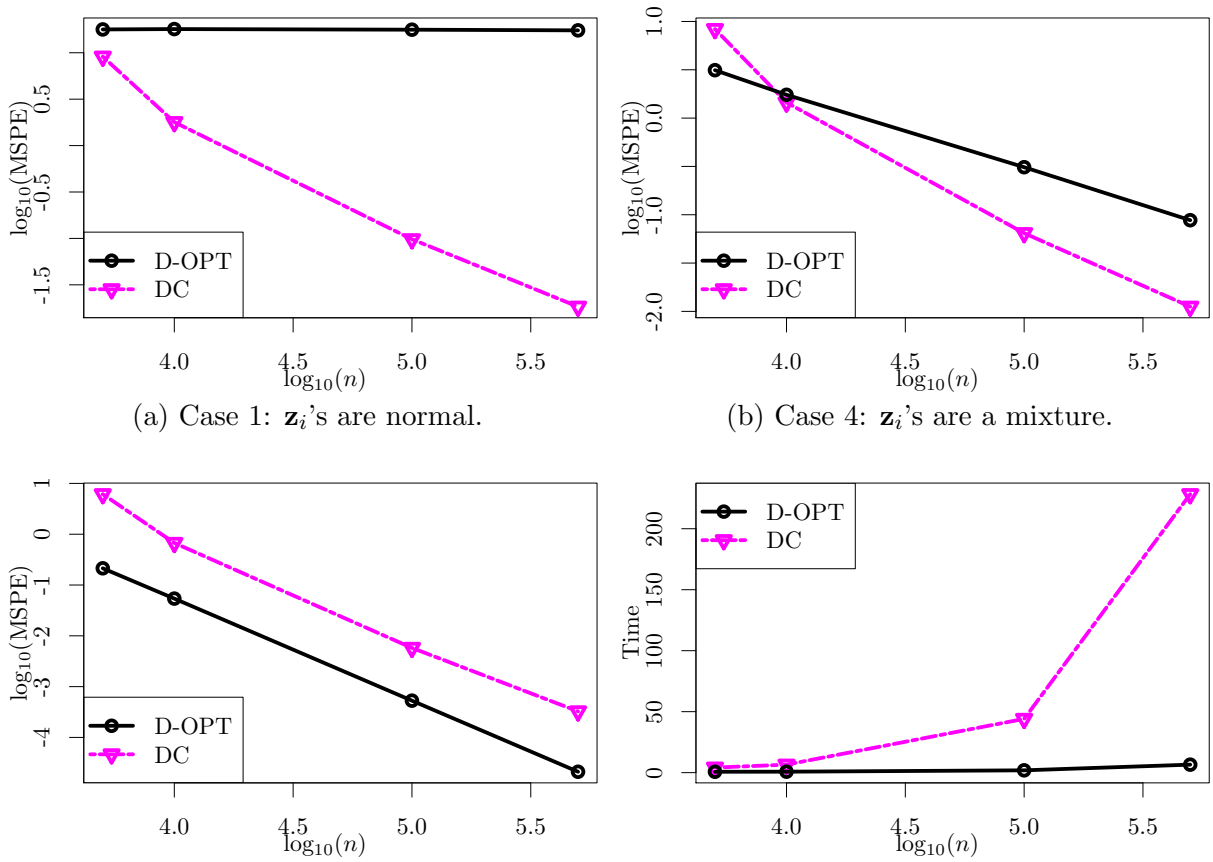
(b) Case 4: $\mathbf{z}_i$'s are a mixture.

(c) Case 6: $\mathbf{z}_i$'s are t_1.

(d) CPU times.

Figure S.7: MSEs and CPU times for estimating the slope parameter: (a)-(c) give results for MSEs and (d) gives results for CPU times. The subdata size k is fixed at $k = 1000$ and the full data size n changes with fix dimension $p = 50$. Logarithm with base 10 is taken of n and MSEs for better presentation of the figures.

To further compare the IBOSS D-OPT method and the DC method with a larger p, we increase the dimension to be $p = 500$. Figure S.8 gives the average CPU times and MSEs for the slope parameters. Full data are generated with sample sizes $n = 5 \times 10^3, 10^4, 10^5$, and 5×10^5. Subdata sample size for the IBOSS method is $k = 1000$. It is seen that the relative performances of estimation efficiency between the IBOSS D-OPT method and the DC method depend on the covariate distribution are similar to those with $p = 50$. In terms of computational cost in Figure S.8 (d), the advantage of the IBOSS D-OPT method is more significant compared with the DC method.



(a) Case 1: $\mathbf{z}_i$'s are normal.

(b) Case 4: $\mathbf{z}_i$'s are a mixture.

(c) Case 6: $\mathbf{z}_i$'s are t_1.

(d) CPU times.

Figure S.8: MSEs and CPU times for estimating the slope parameter: (a)-(c) give results for MSEs and (d) gives results for CPU times. The subdata size k is fixed at $k = 1000$ and the full data size n changes with fixed dimension $p = 500$. Logarithm with base 10 is taken of n and MSEs for better presentation of the figures.

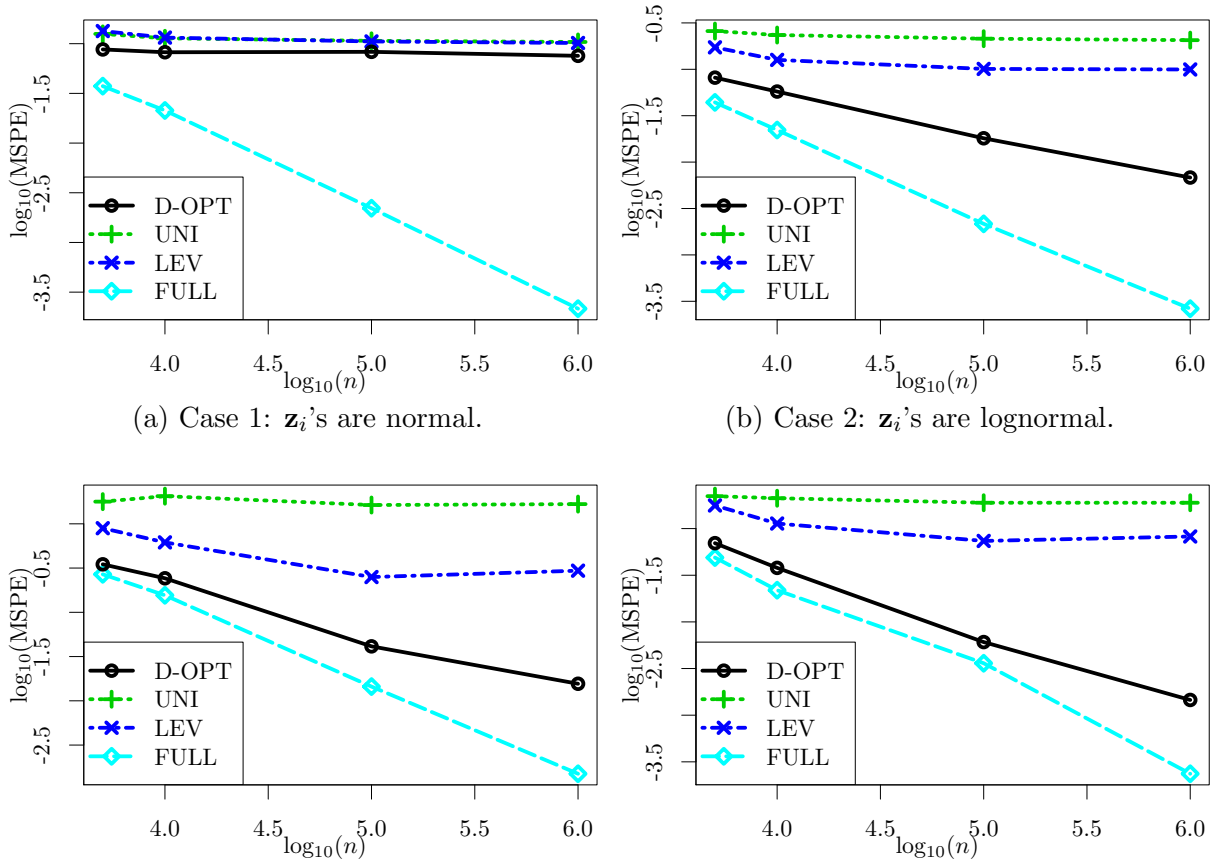
S.7 Performance of IBOSS with regularization method

In this section, we provide numerical results to evaluate the performance of the IBOSS method in application to regularization methods. We use the IBOSS method to select subdata, and then feed it to the elastic net regularization(Zou and Hastie, 2005) method. Full data with dimension $p = 60$ are generated for sample sizes n, with choices of $5 \times 10^3, 10^4, 10^5$, and 10^6. The intercept is set to $\beta_0 = 1$, while the slope parameter $\boldsymbol{\beta}_1$ has a sparse structure with the first 10 element being 0.1 and the rest 50 element being 0.

The elastic net method is implemented using the glmnet R package (Friedman *et al.*, 2010). Tuning parameters are selected using the cross validation method provided in the R package.

We calculate the MSPEs based on 100 repetitions of the simulation. In each repetition, we implement different methods to obtain a subdata set of $k = 1000$, apply the elastic net to the subdata set to estimate a model, and then use the model to calculate the MSPEs based on a new sample of size 5,000. Figure S.9 presents the results of the simulation. It is seen that the relative performance of IBOSS compared with other methods are similar to that of parameter estimation in the main paper. That is, the D-OPT IBOSS method uniformly dominates the subsampling-based methods UNI and LEV, and its advantage is more significant if the tail of the covariate distribution is heavier.

We also implement the ridge regression method. The results are similar so we omit them.



(a) Case 1: $\mathbf{z}_i$'s are normal.

(b) Case 2: $\mathbf{z}_i$'s are lognormal.

(c) Case 3: $\mathbf{z}_i$'s are t_2.

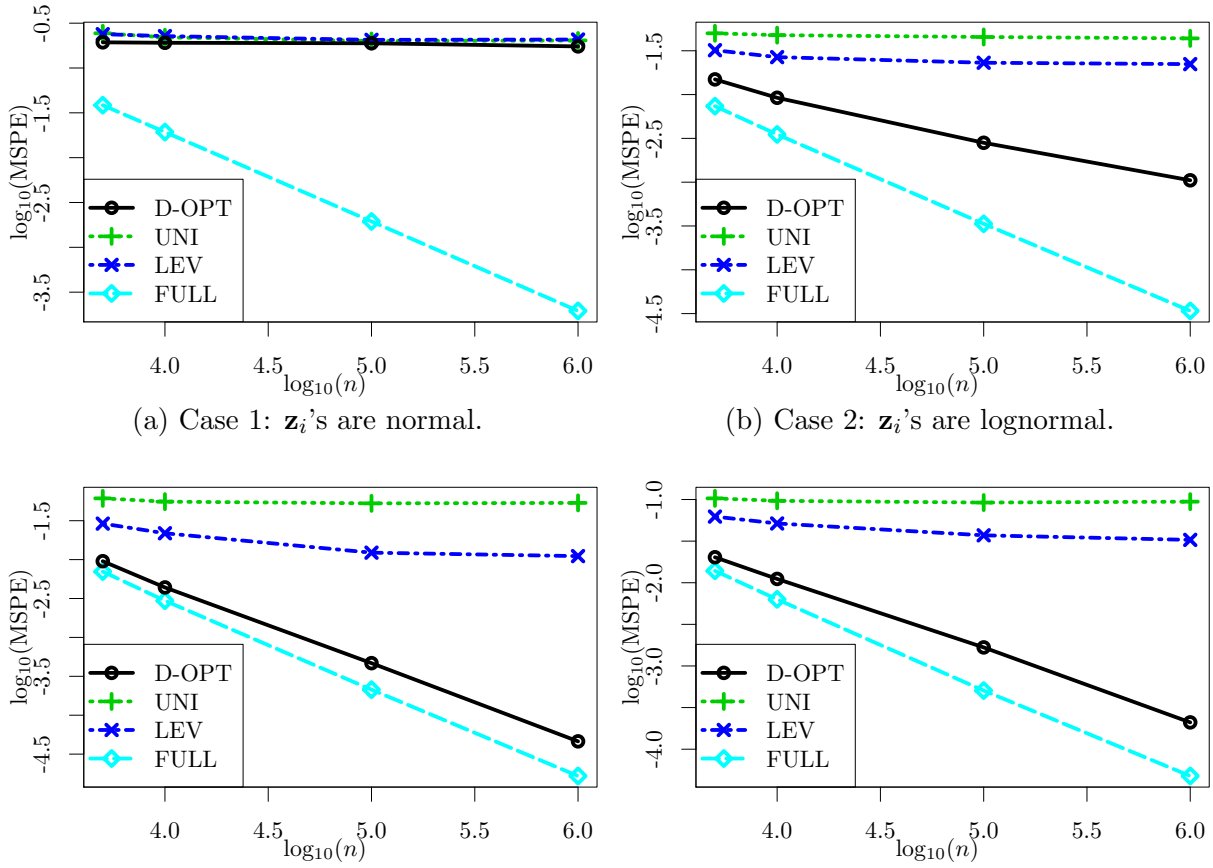
(d) Case 4: $\mathbf{z}_i$'s are a mixture.

Figure S.9: MSPEs for predicting mean responses using the elastic net method with the subdata of size $k = 1000$ selected from the full data. Logarithm with base 10 is taken of the full data sample size n and MSPEs for better presentation of the figures.

S.8 Unequal variance

In this section, we provide a simple numerical study to evaluate the performance of the IBOSS method when the error term in the linear model is heteroscedastic. We use same setup in the main paper to generate the full data except that the standard deviations of the error terms are different and are generated from the exponential distribution with rate parameter 1, i.e., the variance for each error term is randomly generated from a squared exponential random variable. Figure S.10 presents MSE for estimating the slope parameter. It is seen that the relative performance of IBOSS compared with other methods are similar

to that of parameter estimation in the main paper. That is, the D-OPT IBOSS method uniformly dominates the subsampling-based methods UNI and LEV, and its advantage is more significant if the tail of the covariate distribution is heavier. Note that when the error terms have unequal variances, transformations are often used to stabilize the variances or weighted least squares are often used instead of the ordinal least squares. These questions are beyond the scope of this paper and we will investigate them in another project.



(a) Case 1: $\mathbf{z}_i$'s are normal.

(b) Case 2: $\mathbf{z}_i$'s are lognormal.

(c) Case 3: $\mathbf{z}_i$'s are t_2.

(d) Case 4: $\mathbf{z}_i$'s are a mixture.

Figure S.10: MSEs for estimating the slope parameter when the error terms are heteroscedastic. The subdata size k is fixed at $k = 1000$ and the full data size n changes. Logarithm with base 10 is taken of n and MSEs for better presentation of the figures.

References

Battey, H., Fan, J., Liu, H., Lu, J., and Zhu, Z. (2015). Distributed estimation and inference with statistical guarantees. *arXiv preprint arXiv:1509.05457* .

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* **33**, 1, 1–22.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **67**, 2, 301–320.