Statistica Sinica **34** (2024), 911-932
doi:https://doi.org/10.5705/ss.202022.0020

A PERTURBATION SUBSAMPLING FOR LARGE SCALE DATA

Yujing Yao and Zhezhen Jin*

Columbia University

Abstract: When analyzing large-scale data, subsampling methods and divide-and-conquer procedures are appealing, because they ease the computational burden, while preserving the validity of inferences. Here, sampling may occur with or without replacement. In this paper, we propose a perturbation subsampling approach based on independent and identically distributed stochastic weights for analyzing large-scale data. We justify the method based on optimizing convex objective functions by establishing the asymptotic consistency and normality of the resulting estimators. This method simultaneously provides consistent point and variance estimators. We demonstrate the finite-sample performance of the proposed method using simulation studies and two real-data analyses.

Key words and phrases: Convex objective function, distributed computing, optimization, perturbation, subsampling.

1. Introduction

The computation for a full-sample analysis of a large-scale data set is often infeasible. One solution is to use parallel computing to process subsets of the full data, and then combining the results from subsets (McDonald, Hall and Mann (2010); Boyd et al. (2011); Terenin, Simpson and Draper (2015); Jordan, Lee and Yang (2019)). An alternative approach is to use subsampling, where the analysis is based on a selected fraction of the complete data set (Drineas, Mahoney and Muthukrishnan (2006); Mahoney (2011); Dhillon et al. (2013); Ma, Mahoney and Yu (2015); Quiroz et al. (2019)).

Optimal subsampling methods have been studied for various models, based on sampling with or without replacement (Wang, Zhu and Ma (2018); Ai et al. (2021); Keret and Gorfine (2020); Zuo et al. (2021); Wang and Ma (2021)). These optimal subsampling methods require calculating data-dependent nonuniform sampling probabilities for all data at once, which may require significant computational resources. In general, it is nontrivial to justify the asymptotic properties of the estimators resulting from such sampling strategies. The underlying multinomial distribution of the sampling with replacement or the multivariate hypergeometric distribution of the sampling without replacement leads to nonindependence of the subsample, with negative correlations. In other

*Corresponding author.

words, although the subsample observations might be conditionally independent, they are correlated unconditionally. In addition, current results, including those related to optimal subsampling and Poisson subsampling (Särndal, Swensson and Wretman (2003); Wang (2019); Yu et al. (2020); Wang and Ma (2021)), quantify the difference between the subsample estimator and the full data estimator using the conditional distribution and conditional variance. However, theoretical and methodological discussions on the difference between the subsample estimator and the true value are limited.

In this paper, we develop a perturbation subsampling method for statistical inference of large-scale data by optimizing convex objective functions. The proposed method depends on stochastic weights, generated by two steps: the first step draws a subsample using Bernoulli sampling, and the second step generates random perturbation weights for the subsample with a known probability distribution in order to approximate the objective function of the full data. Repeating the perturbation is feasible for a distributed computing framework and provides an empirical distribution for statistical inference. The rest of the paper is organized as follows. In Section 2, we introduce the proposed perturbation subsampling method for analyzing large-scale data. Section 3 presents our theoretical results. Section 4 examines the performance of the proposed approaches using simulated data sets. In Section 5, we analyze real data sets and conclude with a discussion in Section 6. Proofs of the theoretical results are provided in the Supplementary Material.

2. A Perturbation Subsampling for Large-Scale Data

2.1. Optimization of convex objective function

Suppose that Y is the response variable and $\boldsymbol{X}$ is a d-dimensional vector of covariates. The relationship between Y and $\boldsymbol{X}$ can be characterized by a d-dimensional unknown parameter $\boldsymbol{\beta}_0 \in \mathbb{R}^d$, where

$$\boldsymbol{\beta}_0 = \underset{\boldsymbol{\beta}\in\mathbb{R}^d}{\operatorname{argmin}} \, Ef(\boldsymbol{\beta}, \boldsymbol{X}, Y), \tag{2.1}$$

with $f(\boldsymbol{\beta}, \boldsymbol{x}, y)$ being a continuous, convex objective function with respect to $\boldsymbol{\beta}$. Throughout this paper, we assume that $Ef(\boldsymbol{\beta}, \boldsymbol{X}, Y)$ exists and is finite. Based on the independent and identically distributed (i.i.d.) sample $(y_i, \boldsymbol{x}_i)$, for $i = 1, \ldots, n$, we can obtain the estimator of $\boldsymbol{\beta}_0$ by minimizing the empirical analog of the convex objective function:

$$\hat{\boldsymbol{\beta}}_n = \underset{\boldsymbol{\beta}\in\mathbb{R}^d}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} f(\boldsymbol{\beta}, \boldsymbol{x}_i, y_i). \tag{2.2}$$

In general, the resulting estimator is an M-estimator (Niemiro (1992)). It is the maximum likelihood estimator (MLE) when the function $f(\boldsymbol{\beta}, \boldsymbol{x}_i, y_i)$ is the minus of the log-likelihood of $(y_i, \boldsymbol{x}_i)$, and is the L_p-norm estimator if $f(\boldsymbol{\beta}, \boldsymbol{x}_i, y_i) = |y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}|^p$, for a known $p > 1$ and $i = 1, \ldots, n$.

2.2. Perturbation subsampling

We introduce a perturbation subsampling method that reduces the sample size from n to r_n using Bernoulli sampling and approximates the objective function of the full data using a perturbation with independently generated stochastic weights.

The procedure generates two different i.i.d. completely known nonnegative random variables. Here, r_n is the desired reduced sample size ($r_n < n$), and q_n is the ratio between the desired subset size and the full sample size. In the first step, we achieve subsampling by generating n i.i.d. Bernoulli random variables with probability q_n. Specifically, if the generated ith Bernoulli random variable takes the value one, then the ith sample is selected, otherwise, the ith sample is not selected. The size of the resulting subsample is approximately r_n, because it is the expectation of the sum of n copies of Bernoulli random variables. In the second step, we perform a perturbation using nonnegative stochastic weights, generated independently from a known probability distribution with mean $1/q_n$, to approximate the objective function of the full data; see Algorithm 1.

Algorithm 1: A perturbation subsampling algorithm.

1 **Subset**: Generate n i.i.d. random variables $\{U_{n,i}\}_{i=1}^n$, where $U_{n,i} \sim \texttt{Bernoulli}(q_n)$, $q_n = r_n/n$.

2 **Stochastic weighting**: Generate n i.i.d. nonnegative random variables $\{V_{n,i}\}_{i=1}^n$ from a completely known probability distribution with $EV_{n,i} = 1/q_n$.

3 **Estimation**: Minimize the perturbed objective function to obtain an estimator

$$\tilde{\boldsymbol{\beta}}_n = \underset{\boldsymbol{\beta} \in \mathbb{R}^d}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} W_{n,i} f(\boldsymbol{\beta}, \boldsymbol{x}_i, y_i), \text{ where } W_{n,i} = U_{n,i} V_{n,i}. \tag{2.3}$$

Remark 1. The first step is based on Bernoulli sampling in survey sampling (Särndal, Swensson and Wretman (2003)), and has been used in subsampling algorithms such as the pilot subsampling step in Algorithm 2 of Yu et al. (2020). The second step is novel in terms of subsampling algorithms. Here, we use a stochastic weight generated from a known probability distribution to approximate the objective function of the full data, rather than rescale with fixed and data-dependent weights in the subsampling, as in Ma, Mahoney and Yu (2015) and Wang, Zhu and Ma (2018). More importantly, this step can be implemented repeatedly to estimate the variance of the perturbation subsampling estimator.

Remark 2. In practice, the second step needs to generate $V_{n,i}$ only for those $U_{n,i} = 1$. However, for a theoretical justification, it is convenient to assume that we generate $V_{n,i}$ for all $i = 1, \ldots, n$. The same remark holds for Algorithm 2 and 3.

Remark 3. Several common probability distributions can be used in the second step. Examples include the following:

1. Continuous distributions:
   (a) Gamma distribution, for example, $V_n \sim \texttt{Gamma}(1/q_n, 1)$, $V_n \sim \texttt{Exponential}(1/q_n)$;
   (b) Scaled Beta distribution, for example, $V_n \sim 3/q_n\texttt{Beta}(1, 2)$, $V_n \sim \texttt{Uniform}(0, 2/q_n)$;
   (c) Half normal distribution, for example, $V_n \sim \texttt{Half-Normal}(0, q_n^2\pi/2)$.
2. Discrete distributions:
   (a) Geometric distribution, for example, $V_n \sim \texttt{Geometric}(q_n)$;
   (b) Negative binomial distribution, for example, $V_n \sim \texttt{Negative Binomial}$ $(1/q_n, 1/2)$;
   (c) Poisson distribution, for example, $V_n \sim \texttt{Poisson}(1/q_n)$.

Different choices of probability distribution satisfy the expectation requirement of perturbation subsampling, but with different variances. Specifically, if the variance of V_n is b_n^2, then $\text{var}(W_n) = 1/q_n - 1 + b_n^2 q_n$. If $b_n = 0$, then Algorithm 1 is the classic Bernoulli sampling. With the requirement $b_n^2 > 0$, the stochastic weighting can be used repeatedly to estimate the variance of the perturbation subsampling estimator for statistical inference, as in Jin, Ying and Wei (2001). The procedure is summarized in Algorithm 2.

2.3. Repeated perturbation subsampling for large-scale data

Repeatedly using Algorithms 1 and 2 yields a collection of subsampling estimators that can be used for statistical inference. A more general algorithm for repeated perturbation subsampling is summarized in Algorithm 3, which involves both repeated Bernoulli subsampling and repeated stochastic weighting for each Bernoulli subsampling. This algorithm can be implemented under parallel or distributed computational architectures, with the data distributed as subsets across the machines.

Remark 4. In Algorithm 2 and Algorithm 3, we can estimate the conditional variance of the proposed estimator from the repeated subsampling estimates. We can estimate the unconditional variance using an adjustment by a factor that

Algorithm 2: A perturbation subsampling algorithm for variance estimation.

1 **Subset**: Generate n i.i.d. random variables $\{U_{n,i}\}_{i=1}^n$, where $U_{n,i} \sim \texttt{Bernoulli}(q_n)$.

2 **for** *fixed* k, $k = 1, \ldots, m$ *with prespecified number of perturbations* $m(> 1)$ **do**

3 **Stochastic weighting**: Generate n i.i.d. nonnegative random variables $\{V_{n,k,i}\}_{i=1}^n$ from the completely known probability distribution with $E(V_{n,k,i}) = 1/q_n$ and $\text{var}(V_{n,k,i}) = b_n^2$.

4 **Estimation**: Minimize the perturbed objective function to obtain an estimator $\tilde{\boldsymbol{\beta}}_{n,k}$ such that

$$\tilde{\boldsymbol{\beta}}_{n,k} = \underset{\boldsymbol{\beta}\in\mathbb{R}^d}{\text{argmin}} \frac{1}{n}\sum_{i=1}^n W_{n,k,i} f(\boldsymbol{\beta}, \boldsymbol{x}_i, y_i), \text{ where } W_{n,k,i} = U_{n,i}V_{n,k,i}. \quad (2.4)$$

5 **end**

6 **Variance estimation**: The conditional variance of $\tilde{\boldsymbol{\beta}}_n$ in (2.3) can be estimated by

$$\tilde{\text{var}}(\tilde{\boldsymbol{\beta}}_n|\mathcal{D}_n) = \frac{1}{b_n q_n (m-1)} \sum_{k=1}^m \left(\tilde{\boldsymbol{\beta}}_{n,k} - \frac{1}{m}\sum_{k=1}^m \tilde{\boldsymbol{\beta}}_{n,k}\right)\left(\tilde{\boldsymbol{\beta}}_{n,k} - \frac{1}{m}\sum_{k=1}^m \tilde{\boldsymbol{\beta}}_{n,k}\right)^T, \quad (2.5)$$

where $\mathcal{D}_n = \{\boldsymbol{x}_i, y_i\}_{i=1}^n$, and the unconditional variance can be estimated by

$$\tilde{\text{var}}(\tilde{\boldsymbol{\beta}}_n) = \left(\frac{r_n}{n a_n} + 1\right)\tilde{\text{var}}(\tilde{\boldsymbol{\beta}}_n|\mathcal{D}_n), \text{ where } a_n = 1 - q_n + b_n^2 q_n^2. \quad (2.6)$$

involves the subsample size (r_n), the number of parallel processes (M), and the variance of the stochastic weight (b_n^2).

3. Theoretical Results

In this section, we study the theoretical properties of the estimators obtained from Algorithm 1 and Algorithm 3. It is easy to see that Algorithm 2 is a special case of Algorithm 3. Note that the stochastic weights $W_{n,i}$ in the two algorithms are independent of the data $\mathcal{D}_n = \{y_i, \boldsymbol{x}_i\}_{i=1}^n$. Conditional on the data, only the stochastic weights are random. Unconditionally, the randomness in the resulting estimators involves both the stochastic weights and the data. We use Pr^*, E^*, and var^* to denote the conditional probability, conditional expectation, and conditional variance given the data, respectively. We write $||\cdot||$ for the Frobenius norm for a matrix or the Euclidean norm for a vector. We assume the following regularity conditions.

Assumption 1. *The parameter space of* $\boldsymbol{\beta}$ *is compact in* $\mathbb{R}^d$. *The* $\boldsymbol{\beta}_0$ *satisfying* (2.1) *is an interior point of the parameter space and is unique.*

Algorithm 3: A repeated perturbation subsampling algorithm.

Prespecify the number of parallels $M(>1)$:
for *fixed* l, $l = 1, \ldots, M$ **do**
  Subset: Generate n i.i.d. r.v. $\{U_{n,i,l}\}_{i=1}^{n}$, where $U_{n,i,l} \sim \texttt{Bernoulli}(q_n)$.
  for *fixed* k, $k = 1, \ldots, m$ *with prespecified number of perturbations* $m(>1)$ **do**
    Stochastic weighting: Generate n i.i.d. nonnegative r.v. $\{V_{n,i,l,k}\}_{i=1}^{n}$ with $E(V_{n,i,l,k}) = 1/q_n$ and $\text{var}(V_{n,i,l,k}) = b_n^2$.
    Point estimation: Minimize the perturbed objective function such that

$$\tilde{\boldsymbol{\beta}}_{n,l,k} = \underset{\boldsymbol{\beta}\in\mathbb{R}^d}{\text{argmin}} \frac{1}{n}\sum_{i=1}^{n} W_{n,i,l,k} f(\boldsymbol{\beta}, \boldsymbol{x}_i, y_i), \text{ where } W_{n,i,l,k} = U_{n,i,l} V_{n,i,l,k}. \tag{2.7}$$

  end
  Variance estimation: The estimate of $\text{var}(\tilde{\boldsymbol{\beta}}_{n,l})$ can be obtained as in (2.6).
end
Combination: Obtain the final estimator as

$$\tilde{\boldsymbol{\beta}}_n^{(M)} = \left(\sum_{l=1}^{M} \tilde{\text{var}}(\tilde{\boldsymbol{\beta}}_{n,l})^{-1}\right)^{-1} \sum_{l=1}^{M} \tilde{\text{var}}(\tilde{\boldsymbol{\beta}}_{n,l})^{-1} \tilde{\boldsymbol{\beta}}_{n,l}. \tag{2.8}$$

The estimates of the conditional and unconditional variance of $\tilde{\boldsymbol{\beta}}_n^{(M)}$ are

$$\tilde{\text{var}}(\tilde{\boldsymbol{\beta}}_n^{(M)}|\mathcal{D}_n) = \frac{\sum_{l=1}^{M} \tilde{e}_l \tilde{e}_l^T}{M-1} \sum_{l=1}^{M} \left(\tilde{\boldsymbol{\beta}}_{n,l} - \frac{1}{M}\sum_{l=1}^{M} \tilde{\boldsymbol{\beta}}_{n,l}\right)\left(\tilde{\boldsymbol{\beta}}_{n,l} - \frac{1}{M}\sum_{l=1}^{M} \tilde{\boldsymbol{\beta}}_{n,l}\right)^T$$

$$\text{and} \quad \tilde{\text{var}}(\tilde{\boldsymbol{\beta}}_n^{(M)}) = \left(\frac{r_n(\sum_{l=1}^{M} \tilde{e}_l \tilde{e}_l^T)^{-1}}{n a_n} + \boldsymbol{I}_{d\times d}\right) \tilde{\text{var}}(\tilde{\boldsymbol{\beta}}_n^{(M)}|\mathcal{D}_n), \tag{2.9}$$

where $\tilde{e}_l = (\sum_{l=1}^{M} \tilde{\text{var}}(\tilde{\boldsymbol{\beta}}_{n,l})^{-1})^{-1} \tilde{\text{var}}(\tilde{\boldsymbol{\beta}}_{n,l})^{-1}$ and $a_n = 1 - q_n + b_n^2 q_n^2$.

Assumption 2. *The first and second gradients of the convex objective function* $f(\boldsymbol{\beta}, \boldsymbol{x}, y)$ *with respect to* $\boldsymbol{\beta}$ *in a neighborhood of* $\boldsymbol{\beta}_0$ *exist and are finite. The gradients are denoted by* $\dot{f}$ *and* $\ddot{f}$ *respectively, given by* $\dot{f}(\boldsymbol{\beta}_0, \boldsymbol{x}, y) = (\partial f(\boldsymbol{\beta}, \boldsymbol{x}, y)/\partial\boldsymbol{\beta})|_{\boldsymbol{\beta}=\boldsymbol{\beta}_0}$ *and* $\ddot{f}(\boldsymbol{\beta}_0, \boldsymbol{x}, y) = (\partial^2 f(\boldsymbol{\beta}, \boldsymbol{x}, y)/\partial\boldsymbol{\beta}\partial\boldsymbol{\beta}^T)|_{\boldsymbol{\beta}=\boldsymbol{\beta}_0}$. *We further assume that the matrix* $E(\ddot{f}(\boldsymbol{\beta}_0, \boldsymbol{X}, Y))$ *is positive definite.*

Assumption 3. *Stochastic weights* $U_n \sim \texttt{Bern}(q_n)$, $E(V_n) = 1/q_n$, *and there exists* $\alpha > 0$ *such that* $\limsup_{n\to\infty} q_n^{2+\alpha} E V_n^{2+\alpha} < \infty$.

Assumption 1 is required to guarantee the consistency of the minimizer of the convex objective function. Assumption 2 guarantees $E(||\dot{f}(\boldsymbol{\beta}, \boldsymbol{X}, Y)||^2) < \infty$ for each $\boldsymbol{\beta}$ in a neighborhood of $\boldsymbol{\beta}_0$, thus implying the asymptotic normality of the full-data estimator (Theorem 4 of Niemiro (1992)). Assumption 3 is a requirement on the stochastic weights. It is equivalent to there existing $\alpha > 0$

such that $\limsup_{n\to\infty} q_n^{1+\alpha} E W_n^{2+\alpha} < \infty$, because U_n and V_n are independent, $E(W_n^\alpha) = E(U_n^\alpha)E(V_n^\alpha)$, and $E(U_n^\alpha) = q_n$, for any $\alpha > 0$.

Our first theorem establishes the consistency and asymptotic normality of the estimator obtained from Algorithm 1, conditional on the full data.

Theorem 1. *Under Assumptions* 1-3*, the estimator obtained from Algorithm* 1 *satisfies*

$$Pr^*(||\tilde{\boldsymbol{\beta}}_n - \hat{\boldsymbol{\beta}}_n|| \geq \epsilon) \to 0, \text{ as } r_n \to \infty \text{ and } n \to \infty, \tag{3.1}$$

for any $\epsilon > 0$*, and conditional on full data,*

$$(\boldsymbol{C}_{1n}^{-1}\boldsymbol{M}_{1n}\boldsymbol{C}_{1n}^{-1})^{-1/2}\sqrt{\frac{r_n}{a_n}}(\tilde{\boldsymbol{\beta}}_n - \hat{\boldsymbol{\beta}}_n) \xrightarrow{D} N_d(\mathbf{0}, \boldsymbol{I}_{d\times d}), \text{ as } r_n \to \infty, \text{ and } n \to \infty, \tag{3.2}$$

where $\boldsymbol{M}_{1n} = (1/n)\sum_{i=1}^n \dot{f}(\hat{\boldsymbol{\beta}}_n, \boldsymbol{x_i}, y_i)\dot{f}(\hat{\boldsymbol{\beta}}_n, \boldsymbol{x_i}, y_i)^T$, $\boldsymbol{C}_{1n} = (1/n)\sum_{i=1}^n \ddot{f}(\hat{\boldsymbol{\beta}}_n, \boldsymbol{x_i}, y_i)$*, and* $a_n = 1 - q_n + b_n^2 q_n^2$.

The next theorem establishes the unconditional consistency and asymptotic normality of the estimator obtained from Algorithm 1.

Theorem 2. *Under Assumptions* 1-3*, the estimator obtained from Algorithm* 1 *satisfies*

$$Pr(||\tilde{\boldsymbol{\beta}}_n - \boldsymbol{\beta}_0|| \geq \epsilon) \to 0, \text{ as } r_n \to \infty \text{ and } n \to \infty, \tag{3.3}$$

for any $\epsilon > 0$*, and*

$$(\boldsymbol{C}_2^{-1}\boldsymbol{M}_2\boldsymbol{C}_2^{-1})^{-1/2}\sqrt{\frac{r_n}{d_n}}(\tilde{\boldsymbol{\beta}}_n - \boldsymbol{\beta}_0) \xrightarrow{D} N_d(\mathbf{0}, \boldsymbol{I}_{d\times d}), \text{ as } r_n \to \infty \text{ and } n \to \infty, \tag{3.4}$$

where $\boldsymbol{M}_2 = E(\dot{f}(\boldsymbol{\beta}_0, \boldsymbol{X}, Y)\dot{f}(\boldsymbol{\beta}_0, \boldsymbol{X}, Y)^T)$, $\boldsymbol{C}_2 = E(\ddot{f}(\boldsymbol{\beta}_0, \boldsymbol{X}, Y))$*, and* $d_n = 1 + b_n^2 q_n^2$.

Remark 5. The estimate of the unconditional variance of $\tilde{\boldsymbol{\beta}}_n$ can be approximated by the ratio of the adjusting factors, that is, $(r_n/a_n)/(r_n/d_n) = r_n/(na_n) + 1$ times the corresponding estimate of the conditional variance of $\tilde{\boldsymbol{\beta}}_n$, which can be obtained using (2.5) from Algorithm 2, in practice.

Algorithm 2 is a special case of Algorithm 3, with $M = 1$. Our next corollary shows the asymptotic conditional normality of the proposed estimator in Algorithm 3.

Corollary 1. *Suppose that all conditions in Theorem* 1 *hold. Then the estimator given by* (2.8) *obtained from Algorithm* 3 *is* $\sqrt{r_n M}$*-consistent to the full-data estimator conditional on the data, that is,*

$$\left(a_n M \sum_{l=1}^M e_l e_l^T \boldsymbol{C}_{1n}^{-1}\boldsymbol{M}_{1n}\boldsymbol{C}_{1n}^{-1}\right)^{-1/2} \sqrt{r_n M}(\tilde{\boldsymbol{\beta}}_n^{(M)} - \hat{\boldsymbol{\beta}}_n) \xrightarrow{D} N_d(\mathbf{0}, \boldsymbol{I}_{d\times d}),$$

$$\text{as} \quad r_n \to \infty \quad \text{and} \quad n \to \infty, \tag{3.5}$$

where $e_l = (\sum_{l=1}^{M} var(\tilde{\boldsymbol{\beta}}_{n,l})^{-1})^{-1} var(\tilde{\boldsymbol{\beta}}_{n,l})^{-1}$, *and* $a_n = 1 - q_n + b_n^2 q_n^2$.

The next corollary shows the asymptotic unconditional normality of the proposed estimator in Algorithm 3.

Corollary 2. *Suppose that all conditions in Theorem* 2 *hold. Then the estimator given by* (2.8) *obtained from Algorithm* 3 *is* $\sqrt{n}$*-consistent to the true value unconditionally when* $r_n M \geq n$*:*

$$(h_n \boldsymbol{C}_2^{-1} \boldsymbol{M}_2 \boldsymbol{C}_2^{-1})^{-1/2} \sqrt{n} (\tilde{\boldsymbol{\beta}}_n^{(M)} - \boldsymbol{\beta}_0) \xrightarrow{D} N_d(\mathbf{0}, \boldsymbol{I}_{d\times d}), \text{ as } r_n \to \infty \text{ and } n \to \infty, \tag{3.6}$$

where $e_l = (\sum_{l=1}^{M} var(\tilde{\boldsymbol{\beta}}_{n,l})^{-1})^{-1} var(\tilde{\boldsymbol{\beta}}_{n,l})^{-1}$, *and* $h_n = \boldsymbol{I}_{d\times d} + (na_n/r_n) \sum_{l=1}^{M} e_l e_l^T$.

Remark 6. The proposed estimator in Algorithm 3 is $\sqrt{r_n M}$-consistent when $r_n M < n$, and is $\sqrt{n}$-consistent when $r_n M > n$. The estimate of the unconditional variance of $\tilde{\boldsymbol{\beta}}_n^{(M)}$ can be approximated by the ratio of the adjusting factors, that is, $r_n(\sum_{l=1}^{M} e_l e_l^T)^{-1}/(na_n) + \boldsymbol{I}_{d\times d}$ times the corresponding estimate of the conditional variance of $\tilde{\boldsymbol{\beta}}_n^{(M)}$. The conditional variance can be obtained using (2.9) from Algorithm 3, in practice.

Remark 7. When $r_n M > n$, the estimator from the full data is still more efficient than the subsample estimator with perturbations. Under regularity conditions, we can derive the asymptotic distribution of the estimator from the full data, as follows:

$$(\boldsymbol{C}_2^{-1} \boldsymbol{M}_2 \boldsymbol{C}_2^{-1})^{-1/2} \sqrt{n} (\hat{\boldsymbol{\beta}}_n - \boldsymbol{\beta}_0) \xrightarrow{D} N_d(\mathbf{0}, \boldsymbol{I}_{d\times d}). \tag{3.7}$$

The unconditional variance of the repeated perturbation estimator is approximately $h_n(> 1)$ times the variance of the estimator from the full data. Note that different choices of stochastic weight V_n yield different h_n.

4. Simulation Study

In this section, we present simulation studies that evaluate the finite-sample performance of our proposed approaches. Three models are considered: a linear regression model, logistic regression model, and probit regression model. The design matrix $\boldsymbol{X}_{n\times d}$ of the regression models is generated from the following distributions:

1. Mean-zero normal data from a multivariate normal distribution $N(\mathbf{0}, \boldsymbol{\Sigma})$, with $\Sigma_{i,j} = 0.5^{|i-j|}$.

2. Mean-nonzero normal data from a multivariate normal distribution $N(\mathbf{1}, \boldsymbol{\Sigma})$, with $\Sigma_{i,j} = 0.5^{|i-j|}$.

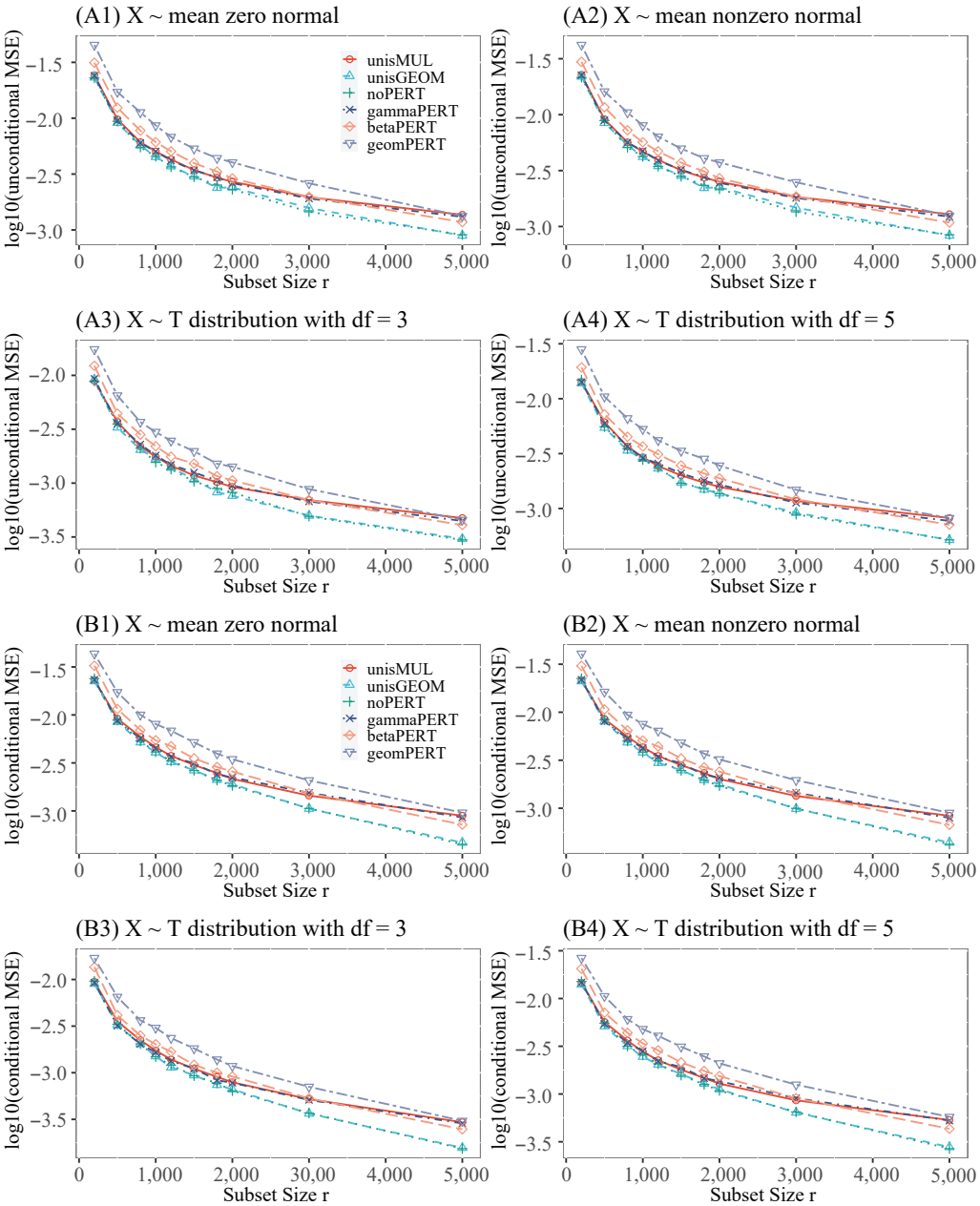

Figure 1. Empirical MSEs for a linear regression model, based on $N = 1{,}000$ simulations.

3. T3 data from a multivariate t-distribution $\mathbf{t}_3(\mathbf{0}, \boldsymbol{\Sigma})$, with $\Sigma_{i,j} = 0.5^{|i-j|}$.

4. T5 data from a multivariate t-distribution $\mathbf{t}_5(\mathbf{0}, \boldsymbol{\Sigma})$, with $\Sigma_{i,j} = 0.5^{|i-j|}$.

The true regression coefficients are set to be a 3×1 vector of ones ($d = 3$). A total of $N = 1{,}000$ data sets are generated, with a sample size of $n = 10{,}000$

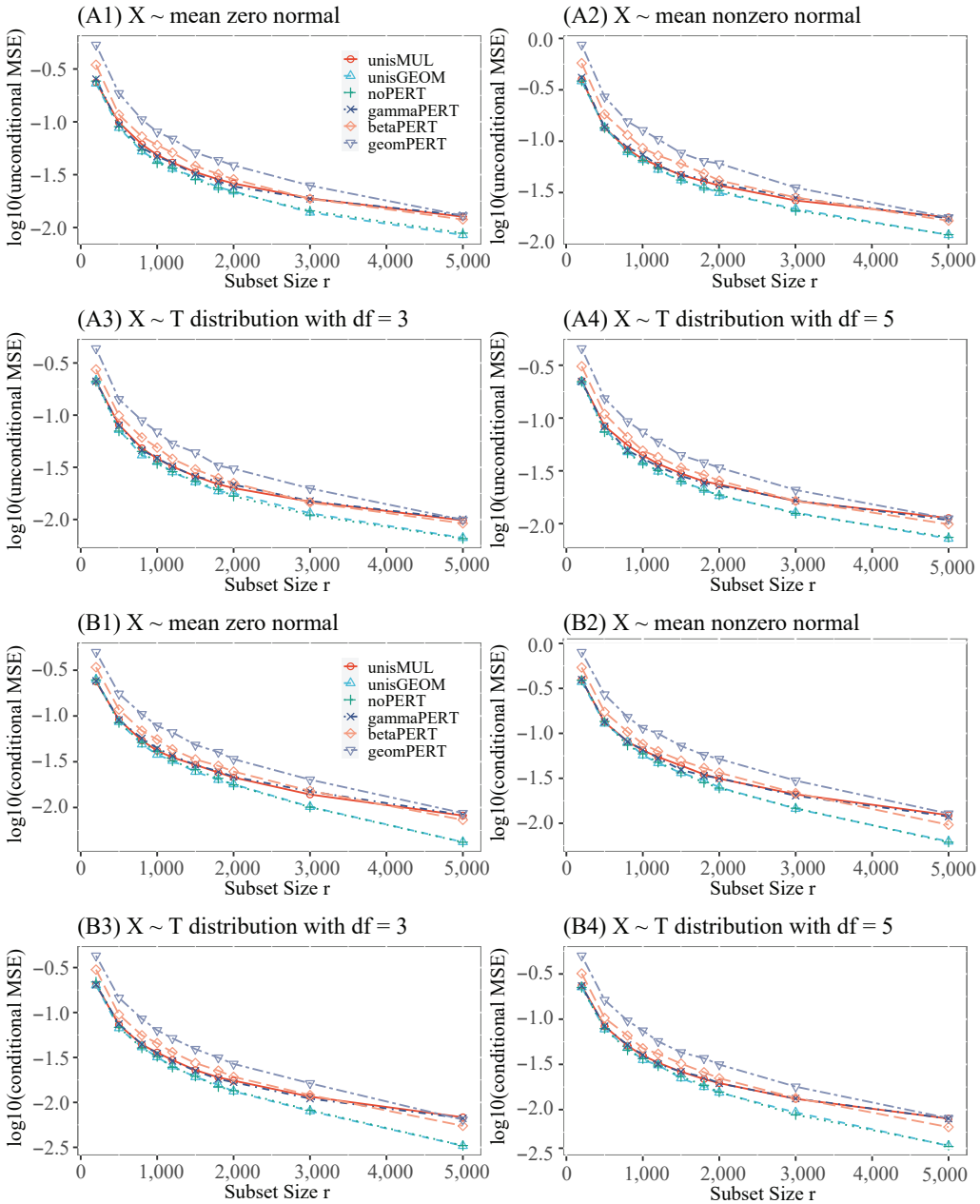

Figure 2. Empirical MSEs for a logistic regression model, based on $N = 1{,}000$ simulations.

for each data set. The target sample size r_n of the subsampling is set to $200, 500, 800, 1000, 1200, 1500, 1800, 2000, 3000$ to 5000.

The empirical mean squared error (MSE) is used to evaluate the proposed procedures and to compare different estimators. The unconditional MSE is

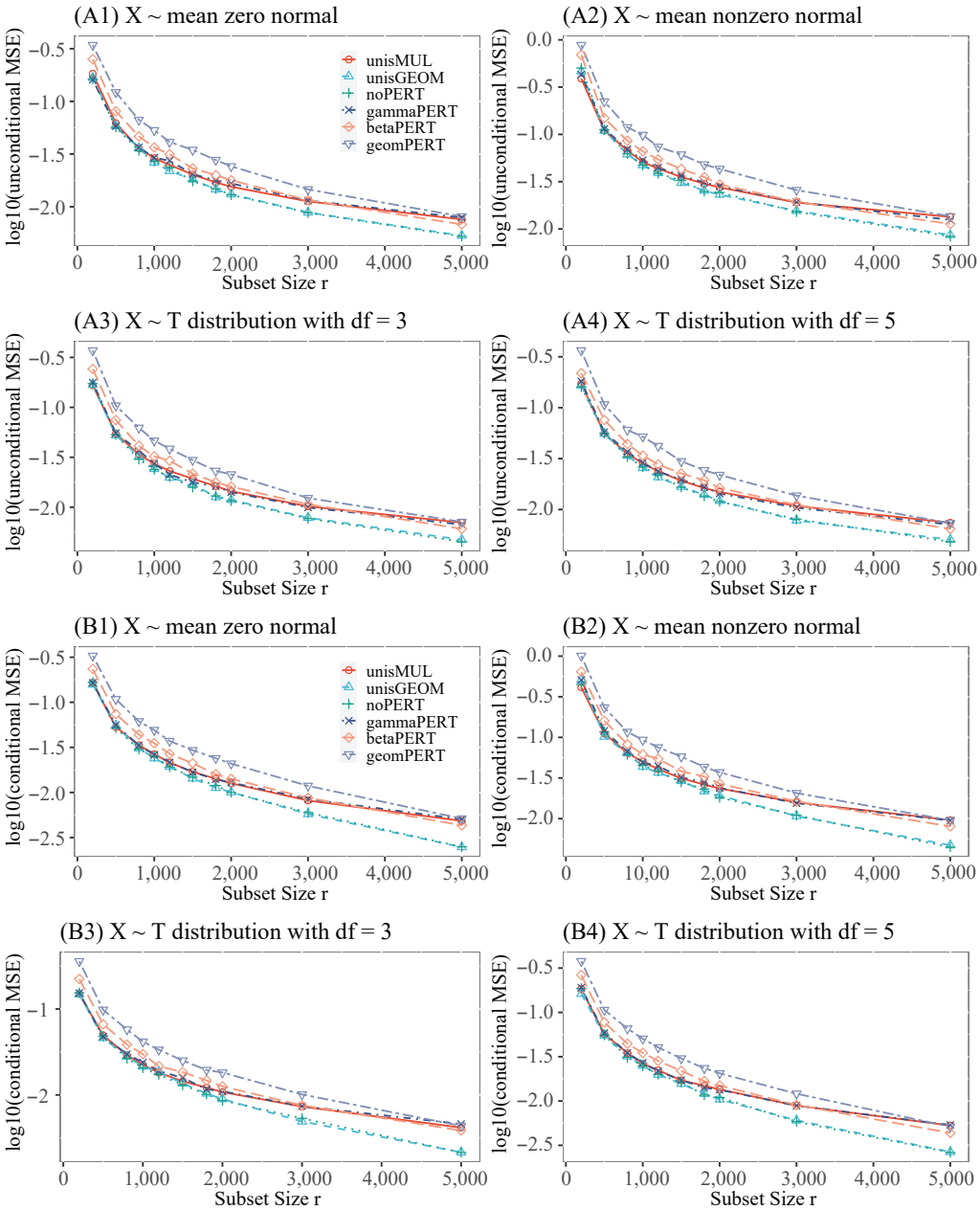

Figure 3. Empirical MSEs for a probit regression model, based on $N = 1{,}000$ simulations.

defined as $(1/N)\sum_{i=1}^{N}||\tilde{\boldsymbol{\beta}}_{n,i} - \boldsymbol{\beta}_0||^2$. The conditional MSE is defined as $(1/N)$ $\sum_{i=1}^{N}||\tilde{\boldsymbol{\beta}}_{n,i} - \hat{\boldsymbol{\beta}}_n||^2$. We use the empirical coverage of the confidence intervals to examine the proposed variance estimator. The computing time, including the subsampling and the estimation, is used to assess the computational efficiency. All methods are implemented in the R programming language (R Core Team

Table 1. Empirical coverage probabilities of β_{10} based on the proposed unconditional variance estimation method from Algorithm 2, with different subset sizes, for a linear regression model with *m=500, **m=1000, based on $N = 1000$ simulations.

Design matrix	Size r	gammaPERT*	betaPERT*	geomPERT*	gammaPERT**	betaPERT**	geomPERT**
Mean zero	1,000	0.945	0.915	0.850	0.947	0.897	0.846
normal	1,200	0.947	0.912	0.868	0.950	0.924	0.850
	1,500	0.956	0.942	0.872	0.943	0.941	0.863
	1,800	0.950	0.935	0.877	0.951	0.922	0.867
	2,000	0.949	0.940	0.865	0.958	0.932	0.871
	3,000	0.939	0.943	0.885	0.946	0.928	0.889
Mean nonzero	1,000	0.949	0.926	0.849	0.950	0.913	0.843
normal	1,200	0.949	0.920	0.871	0.955	0.917	0.836
	1,500	0.962	0.944	0.873	0.951	0.939	0.867
	1,800	0.954	0.927	0.875	0.951	0.927	0.879
	2,000	0.948	0.935	0.872	0.951	0.936	0.874
	3,000	0.941	0.945	0.886	0.943	0.937	0.891
T3	1,000	0.942	0.922	0.837	0.950	0.925	0.854
	1,200	0.952	0.919	0.868	0.925	0.926	0.869
	1,500	0.950	0.916	0.871	0.939	0.934	0.866
	1,800	0.951	0.927	0.868	0.948	0.936	0.881
	2,000	0.945	0.925	0.872	0.950	0.937	0.877
	3,000	0.944	0.944	0.910	0.936	0.936	0.907
T5	1,000	0.948	0.923	0.859	0.944	0.936	0.847
	1,200	0.958	0.923	0.860	0.932	0.935	0.841
	1,500	0.963	0.926	0.867	0.952	0.919	0.865
	1,800	0.950	0.926	0.875	0.946	0.918	0.886
	2,000	0.944	0.938	0.864	0.952	0.921	0.896
	3,000	0.953	0.945	0.905	0.954	0.947	0.912

(2013)). The computations are run on HPC, a Linux-based (CentOS 7.6.1810) computer cluster in the Department of Systems Biology at Columbia University.

The following procedures are considered to evaluate Algorithm 1:

1. Uniform subsampling estimator based on sampling with replacement (unisMUL);
2. Uniform subsampling estimator based on sampling without replacement (unisGEOM);
3. Bernoulli subsampling estimator (noPERT);
4. Perturbation subsampling estimator based on $\texttt{Gamma}(1/q_n, 1)$ distribution (gammaPERT);
5. Perturbation subsampling estimator based on $2/q_n\texttt{Beta}(1, 1)$ distribution (betaPERT);
6. Perturbation subsampling estimator based on $\texttt{Geometric}(q_n)$ distribution (geomPERT).

The results are shown in Figure 1-3. The patterns are similar among the three regression models. The unconditional MSEs (A1-A4) are larger than the

Table 2. Empirical coverage probabilities of β_{10} based on the proposed unconditional variance estimation method from Algorithm 2, with different subset sizes, for a logistic regression model with *m=500, **m=1000, based on $N = 1000$ simulations.

Design matrix	Size r	gammaPERT*	betaPERT*	geomPERT*	gammaPERT**	betaPERT**	geomPERT**
Mean zero	1,000	0.956	0.924	0.848	0.947	0.909	0.857
normal	1,200	0.941	0.932	0.864	0.954	0.924	0.844
	1,500	0.941	0.921	0.881	0.950	0.940	0.848
	1,800	0.953	0.925	0.872	0.941	0.922	0.872
	2,000	0.957	0.931	0.881	0.952	0.924	0.885
	3,000	0.939	0.936	0.884	0.943	0.934	0.908
Mean nonzero	1,000	0.946	0.932	0.847	0.950	0.931	0.845
normal	1,200	0.938	0.927	0.845	0.963	0.924	0.868
	1,500	0.948	0.930	0.862	0.957	0.941	0.877
	1,800	0.944	0.937	0.878	0.960	0.918	0.871
	2,000	0.961	0.935	0.885	0.951	0.912	0.869
	3,000	0.953	0.946	0.911	0.949	0.936	0.907
T3	1,000	0.942	0.924	0.837	0.945	0.931	0.854
	1,200	0.943	0.945	0.863	0.937	0.915	0.860
	1,500	0.952	0.944	0.863	0.949	0.921	0.874
	1,800	0.961	0.922	0.864	0.947	0.934	0.849
	2,000	0.955	0.947	0.859	0.941	0.936	0.867
	3,000	0.945	0.929	0.894	0.944	0.935	0.900
T5	1,000	0.950	0.922	0.852	0.943	0.932	0.838
	1,200	0.954	0.920	0.855	0.950	0.906	0.859
	1,500	0.961	0.934	0.844	0.961	0.926	0.861
	1,800	0.942	0.928	0.862	0.942	0.936	0.866
	2,000	0.957	0.939	0.882	0.945	0.919	0.880
	3,000	0.954	0.947	0.910	0.946	0.957	0.891

conditional MSE (B1-B4). Both MSEs decrease as the subset size increases under different design matrices for all methods. The perturbation subsampling estimators with Gamma and Beta distributions perform similarly to the sampling with replacement method. The sampling without replacement method and the Bernoulli subsampling estimator yield the lowest MSEs, and the perturbation subsampling estimator with a Geometric distribution has the largest MSE.

Table 1 shows the coverage probabilities based on Algorithm 2 with $m = 500$ and $m = 1{,}000$ for the linear regression model. The empirical coverage of the 95% confidence interval for β_{10} based on the perturbation with a Gamma distribution is close to the nominal level, and those based on perturbations with Beta and Geometric distributions are less than the nominal level. The empirical coverage remains stable when the perturbation number increases from $m = 500$ to $m = 1{,}000$. Table 2 shows similar results for the logistic regression.

The following procedures are considered to evaluate the performance of Algorithm 3:

1. Full-data estimator (Full);

2. Uniform subsampling estimator based on sampling with replacement (unisMUL);

Table 3. Computing time of repeated perturbation subsampling with $M = 20$ compared with that of other subsampling methods. The CPU time is the average of $N = 1{,}000$ simulations across all subsample sizes and four design matrices.

Method	unisMUL	A-optsMUL	L-optsMUL	gammaPERT	betaPERT	geomPERT
CPU time(s)	0.008	4.576	4.231	1.368	1.632	1.414

3. Two-step A-optimal subsampling estimator (A-optsMUL) (Ai et al. (2021));

4. Two-step L-optimal subsampling estimator (L-optsMUL) (Ai et al. (2021));

5. Repeated perturbation subsampling estimator based on $\texttt{Gamma}(1/q_n, 1)$ distribution (gammaPERT) with $M = 20, 50$;

6. Repeated perturbation subsampling estimator based on $2/q_n\texttt{Beta}(1, 1)$ distribution (betaPERT) with $M = 20, 50$;

7. Repeated perturbation subsampling estimator based on $\texttt{Geometric}(q_n)$ distribution (geomPERT) with $M = 20, 50$.

The A-optimal and L-optimal nonuniform sampling probabilities are obtained using the procedures in Ai et al. (2021). The sample size used to calculate the initial estimator for the optimal subsampling methods is set to 200. The number of perturbations m for each parallel process is set to 200. For the linear regression model, Figure 4 shows that the optimal subsampling exhibits greater estimation accuracy than that of the uniform subsampling method, especially when $\boldsymbol{X}_{n\times d}$ is generated from the t-distribution. The repeated perturbation subsampling estimators have much smaller MSEs. Increasing M from 20 to 50 yields smaller MSEs. The repeated perturbation subsampling methods also take less computational time than the optimal methods do (Table 3). Similar patterns are observed for the logistic regression model.

Table 4 shows the empirical coverage of the 95% confidence interval for β_{10} in a linear regression model based on Algorithm 3 with $M = 50, 200$. The empirical coverage is close to the nominal confidence level 0.95 for all three perturbation distributions considered. The empirical coverage remains stable when the perturbation number increases from $M = 50$ to $M = 200$. Table 5 shows the simulation results for the logistic regression.

The simulation studies show that perturbations with Gamma and Beta distributions yield better results than those with a Geometric distribution. Furthermore, the perturbation number m should be large if Algorithm 2 is used, but can be smaller if Algorithm 3 is used.

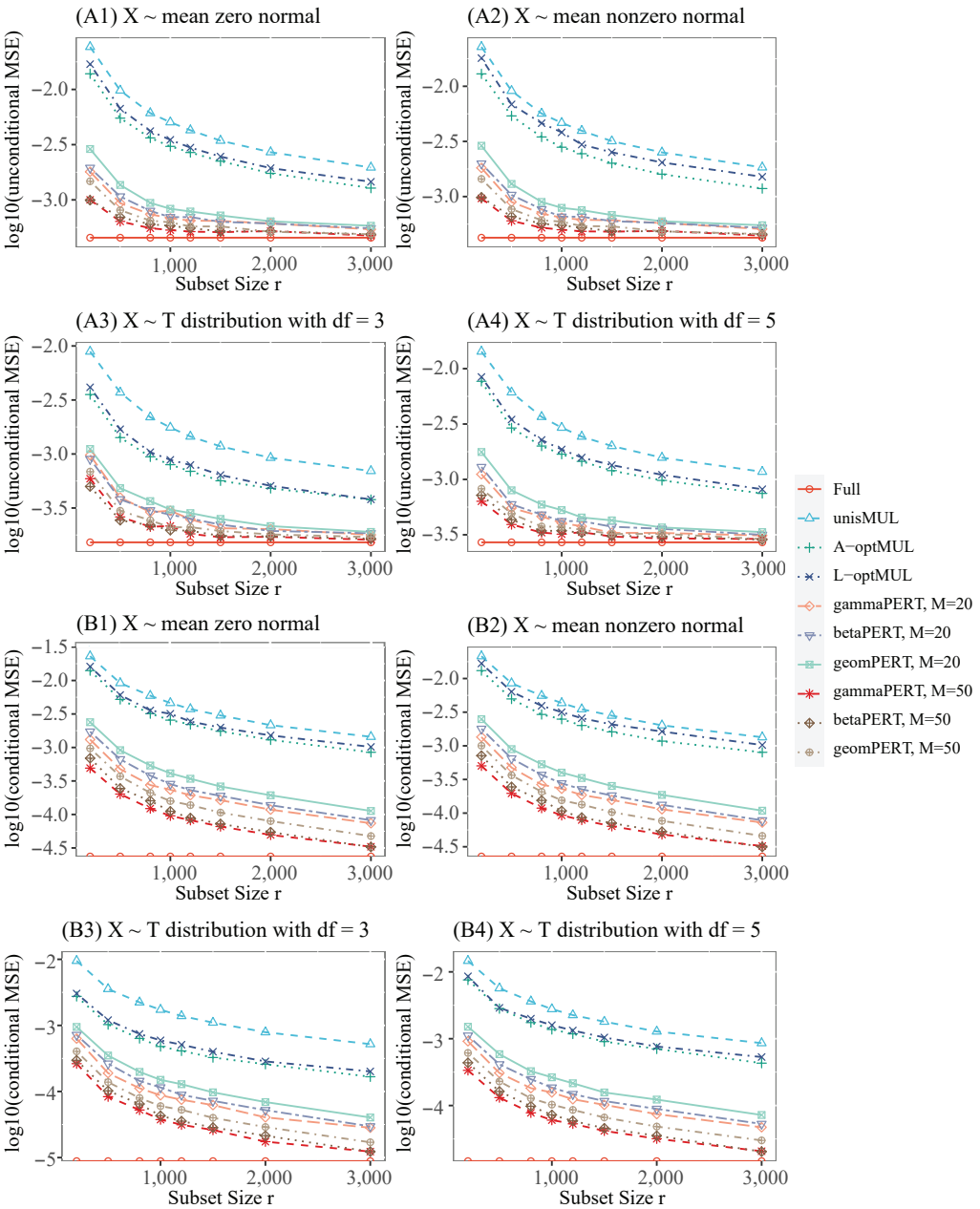

Figure 4. Comparison of empirical MSEs between repeated perturbation subsampling with $M = 20, 50$ and other subsampling methods for a linear regression model based on $N = 1{,}000$ simulations.

5. Application

5.1. Oceanographic data

In this section, we apply our proposed procedure to analyze the California Cooperative Oceanic Fisheries Investigations (CalCOFI) Hydrographic

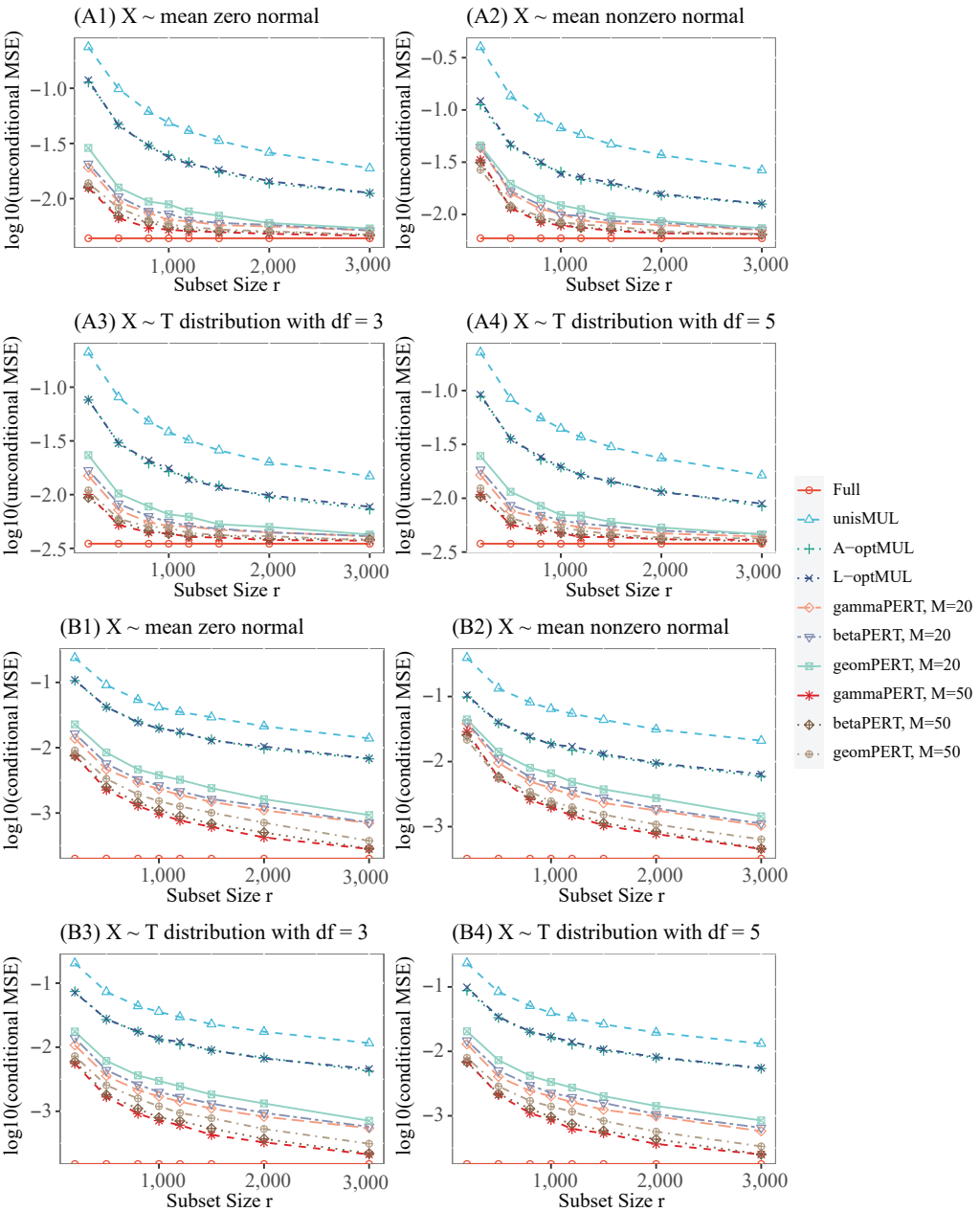

Figure 5. Comparison of empirical MSEs between repeated perturbation subsampling with $M = 20, 50$ and other subsampling methods for a logistic regression model based on $N = 1{,}000$ simulations.

data set. The Hydrographic data set provides the longest duration oceanographic data since 1949, and is available at `https://calcofi.org/data/oceanographic-data/`. The current data set has 887,018 records, with $n = 682{,}876$ complete cases. We use the complete cases to examine the effect of

Table 4. Empirical coverage probabilities of β_{10} based on the proposed unconditional variance estimation method from Algorithm 3, with different subset sizes, for a linear regression model with *M=50, **M=200, based on $N = 1{,}000$ simulations.

Design matrix	Size r	gammaPERT*	betaPERT*	geomPERT*	gammaPERT**	betaPERT**	geomPERT**
Mean zero	500	0.958	0.950	0.932	0.962	0.960	0.954
normal	1,000	0.932	0.950	0.946	0.956	0.954	0.948
	1,200	0.954	0.960	0.948	0.952	0.960	0.948
	1,500	0.954	0.952	0.938	0.956	0.962	0.964
	2,000	0.944	0.950	0.948	0.962	0.954	0.958
	3,000	0.942	0.954	0.950	0.958	0.952	0.956
Mean nonzero	500	0.956	0.958	0.936	0.950	0.964	0.946
normal	1,000	0.936	0.940	0.936	0.954	0.958	0.944
	1,200	0.952	0.954	0.936	0.944	0.950	0.952
	1,500	0.958	0.956	0.948	0.950	0.956	0.952
	2,000	0.934	0.946	0.948	0.954	0.948	0.956
	3,000	0.942	0.954	0.946	0.956	0.942	0.954
T3	500	0.974	0.976	0.948	0.978	0.978	0.974
	1,000	0.962	0.958	0.956	0.970	0.966	0.964
	1,200	0.958	0.972	0.964	0.966	0.974	0.966
	1,500	0.966	0.964	0.952	0.972	0.968	0.968
	2,000	0.960	0.956	0.944	0.958	0.958	0.962
	3,000	0.958	0.968	0.956	0.962	0.960	0.964
T5	500	0.964	0.950	0.934	0.962	0.952	0.956
	1,000	0.962	0.950	0.962	0.960	0.958	0.958
	1,200	0.950	0.954	0.954	0.960	0.950	0.964
	1,500	0.956	0.956	0.952	0.954	0.966	0.952
	2,000	0.960	0.946	0.960	0.956	0.954	0.958
	3,000	0.950	0.954	0.942	0.956	0.952	0.960

salinity (in grams of salt per kilogram of water (g/kg)) and oxygen (mixing ratio in ml/L) on the sea surface temperature using the following linear regression model (Bograd and Lynn (2003); Sivasankari and Anandan (2020)),

$$\text{Sea surface temperature} = \beta_0 + \beta_1 \text{Salinity} + \beta_2 \text{Oxygen} + \text{error}. \tag{5.1}$$

Table 6 shows the analysis results. The analysis with the full data set shows that both salinity and oxygen are significantly associated with sea surface temperature (salinity: 2.406, SE 0.002, p-value < 0.001; oxygen: 4.269, SE 0.011, p-value < 0.001). The proposed repeated perturbation subsampling with $r_n = 10000$, $M = 50$, and $m = 200$ yields similar results. The conditional standard errors are approximately the same as or larger than $\sqrt{na_n/(r_n M)}$ (1.17 for gammaPERT, 1.34 for betaPERT, 1.64 for geomPERT) times the full-data standard errors, which is consistent with Corollary 1. These results indicate that the point estimation is robust across different subset perturbations. The unconditional variances of the perturbation estimates are approximately h_n (2.37 for gammaPERT, 2.80 for betaPERT, 3.69 for geomPERT) times the variances of the full-data estimates, which is consistent with Corollary 2.

Table 5. Empirical coverage probabilities of β_{10} based on the proposed unconditional variance estimation method from Algorithm 3, with different subset sizes, for a logistic regression model with *M=50, **M=200, based on $N = 1{,}000$ simulations.

Design matrix	Size r	gammaPERT*	betaPERT*	geomPERT*	gammaPERT**	betaPERT**	geomPERT**
Mean zero	500	0.946	0.964	0.958	0.948	0.944	0.964
normal	1,000	0.948	0.954	0.964	0.952	0.946	0.964
	1,200	0.950	0.952	0.950	0.964	0.952	0.950
	1,500	0.962	0.946	0.966	0.950	0.954	0.956
	2,000	0.952	0.954	0.966	0.954	0.958	0.952
	3,000	0.952	0.952	0.956	0.952	0.952	0.952
Mean nonzero	500	0.926	0.926	0.944	0.916	0.932	0.944
normal	1,000	0.938	0.938	0.950	0.948	0.952	0.950
	1,200	0.948	0.938	0.950	0.942	0.958	0.956
	1,500	0.946	0.944	0.944	0.952	0.956	0.962
	2,000	0.950	0.944	0.956	0.946	0.956	0.948
	3,000	0.950	0.964	0.950	0.956	0.954	0.960
T3	500	0.952	0.952	0.956	0.948	0.946	0.960
	1,000	0.942	0.956	0.938	0.948	0.942	0.950
	1,200	0.928	0.944	0.950	0.940	0.948	0.944
	1,500	0.928	0.938	0.946	0.930	0.938	0.940
	2,000	0.944	0.936	0.934	0.936	0.934	0.936
	3,000	0.920	0.926	0.926	0.928	0.928	0.930
T5	500	0.954	0.962	0.964	0.964	0.960	0.968
	1,000	0.952	0.960	0.948	0.956	0.966	0.970
	1,200	0.956	0.964	0.954	0.962	0.968	0.966
	1,500	0.956	0.950	0.972	0.962	0.964	0.952
	2,000	0.950	0.948	0.952	0.966	0.962	0.968
	3,000	0.954	0.960	0.956	0.966	0.960	0.962

Table 6. Estimation of coefficients and standard errors for salinity and oxygen in the linear association with sea surface temperature based on the CalCOFI Hydrographic data set. The results from the full data set and from repeated perturbation algorithms with $M = 50$ and $r_n = 10{,}000$ are presented.

	Full		gammaPERT		betaPERT		geomPERT	
	Coef.	SE	Coef. (Cond. SE)	SE	Coef. (Cond. SE)	SE	Coef. (Cond. SE)	SE
Intercept	-141.7	0.370	-141.9(0.750)	0.983	-142.8(1.032)	1.280	-141.5(0.919)	1.080
Salinity	4.269	0.011	4.275(0.022)	0.029	4.299(0.030)	0.037	4.262(0.027)	0.031
Oxygen	2.406	0.002	2.404(0.004)	0.006	2.411(0.006)	0.007	2.402(0.005)	0.006

5.2. Supersymmetric benchmark data set

To demonstrate using the proposed approach with logistic regression and probit regression models, we analyze the supersymmetric (SUSY) benchmark data set of Baldi, Sadowski and Whiteson (2014) and Wang, Zhu and Ma (2018). The data set is available at the Machine Learning Repository (Lichman (2013)) at `https://archive.ics.uci.edu/ml/datasets/SUSY`, and has $n = 5{,}000{,}000$ records. The machine learning method with the full data set used in Baldi, Sadowski and Whiteson (2014) requires very large computer memory and an advanced processor. Calculating $n = 5{,}000{,}000$ nonuniform sampling probabilities for the optimal subsampling method took a much longer time.

Table 7. Estimation of the regression coefficients of the logistic classifier using 18 kinematic features for the supersymmetric benchmark data set. The results from the full data set and the repeated perturbation algorithms with $M = 50$ and $r_n = 10{,}000$ are presented.

Kinematic	Full		gammaPERT		betaPERT		geomPERT	
Feathers	Coef.	SE	Coef. (Cond. SE)	SE	Coef. (Cond. SE)	SE	Coef. (Cond. SE)	SE
feature1	4.683	0.010	4.711(0.033)	0.034	4.742(0.039)	0.040	4.725(0.053)	0.054
feature2	2.326	0.009	2.316(0.031)	0.033	2.362(0.034)	0.035	2.387(0.047)	0.049
feature3	−1.714	0.008	−1.756(0.026)	0.028	−1.694(0.034)	0.035	−1.724(0.041)	0.042
feature4	−0.623	0.005	−0.617(0.014)	0.014	−0.623(0.017)	0.017	−0.623(0.023)	0.023
feature5	−1.601	0.014	−1.643(0.054)	0.056	−1.629(0.054)	0.056	−1.621(0.066)	0.068
feature6	−0.410	0.004	−0.400(0.013)	0.014	−0.395(0.016)	0.017	−0.440(0.020)	0.020
feature7	0.470	0.005	0.501(0.015)	0.016	0.462(0.020)	0.021	0.479(0.022)	0.023
feature8	1.106	0.012	1.120(0.038)	0.039	0.990(0.051)	0.053	1.059(0.054)	0.055
feature9	0.317	0.005	0.317(0.016)	0.017	0.293(0.018)	0.018	0.294(0.019)	0.019
feature10	−2.038	0.034	−2.133(0.121)	0.127	−2.214(0.126)	0.131	−2.050(0.132)	0.135
feature11	0.533	0.009	0.546(0.038)	0.040	0.480(0.036)	0.037	0.562(0.051)	0.052
feature12	0.098	0.005	0.098(0.02)	0.021	0.099(0.017)	0.018	0.096(0.026)	0.026
feature13	0.204	0.036	0.274(0.134)	0.141	0.371(0.135)	0.140	0.169(0.148)	0.151
feature14	0.004	0.001	0.005(0.004)	0.004	−0.001(0.004)	0.004	0.005(0.005)	0.005
feature15	−0.002	0.001	−0.002(0.003)	0.003	0.001(0.005)	0.005	0.002(0.006)	0.006
feature16	0.001	0.001	0.005(0.004)	0.005	−0.002(0.004)	0.004	0.014(0.005)	0.005
feature17	−0.0002	0.001	−0.0001(0.004)	0.005	0.001(0.004)	0.004	−0.011(0.004)	0.005
feature18	−0.0001	0.001	−0.004(0.003)	0.003	−0.002(0.005)	0.005	−0.004(0.006)	0.006

Table 8. Estimation of the regression coefficients of the probit classifier using 18 kinematic features for the supersymmetric benchmark data set. The results from the full data set and the repeated perturbation algorithms with $M = 50$ and $r_n = 10{,}000$ are presented.

Kinematic	Full		gammaPERT		betaPERT		geomPERT	
Feathers	Coef.	SE	Coef. (Cond. SE)	SE	Coef. (Cond. SE)	SE	Coef. (Cond. SE)	SE
feature1	2.665	0.006	2.689(0.022)	0.023	2.727(0.025)	0.026	2.717(0.036)	0.037
feature2	−1.065	0.004	−1.087(0.017)	0.017	−1.047(0.021)	0.022	−1.074(0.026)	0.027
feature3	1.274	0.005	1.272(0.020)	0.020	1.309(0.022)	0.023	1.317(0.031)	0.031
feature4	−1.041	0.008	−1.067(0.037)	0.038	−1.065(0.038)	0.039	−1.073(0.049)	0.050
feature5	0.868	0.007	0.859(0.027)	0.028	0.759(0.037)	0.038	0.821(0.039)	0.040
feature6	−0.329	0.003	−0.328(0.009)	0.009	−0.331(0.011)	0.011	−0.328(0.015)	0.015
feature7	−0.206	0.002	−0.200(0.008)	0.009	−0.210(0.011)	0.012	−0.234(0.013)	0.013
feature8	0.299	0.003	0.320(0.010)	0.011	0.289(0.013)	0.014	0.310(0.015)	0.015
feature9	0.221	0.003	0.216(0.010)	0.011	0.199(0.012)	0.012	0.202(0.014)	0.014
feature10	0.333	0.005	0.346(0.026)	0.027	0.310(0.027)	0.028	0.361(0.038)	0.039
feature11	−0.964	0.019	−1.062(0.078)	0.082	−1.107(0.086)	0.089	−1.009(0.091)	0.092
feature12	0.096	0.003	0.085(0.013)	0.013	0.086(0.012)	0.013	0.089(0.019)	0.019
feature13	−0.061	0.020	0.019(0.087)	0.091	0.070(0.091)	0.094	−0.052(0.099)	0.101
feature14	0.002	0.001	0.003(0.002)	0.002	−0.002(0.002)	0.002	0.001(0.003)	0.003
feature15	−0.001	0.001	−0.001(0.002)	0.002	−0.0001(0.003)	0.003	−0.0004(0.004)	0.004
feature16	0.001	0.001	0.003(0.003)	0.003	−0.001(0.002)	0.002	0.006(0.003)	0.003
feature17	−0.0002	0.001	−0.002(0.002)	0.002	−0.001(0.003)	0.003	−0.002(0.004)	0.004
feature18	−0.0001	0.001	−0.0001(0.003)	0.003	0.000(0.002)	0.002	−0.006(0.003)	0.003

The goal of the analysis is to differentiate a process where new supersymmetric particles are produced from a background process, with 18 kinematic features as

covariates. Two models are considered, namely, a logistic regression model and a probit regression model:

$$\text{logit}(Pr\{Y=1\}) = \beta_0 + \sum_{j=1}^{18} \beta_j \text{kinematic feature}_j, \tag{5.2}$$

$$\text{probit}(Pr\{Y=1\}) = \beta_0 + \sum_{j=1}^{18} \beta_j \text{kinematic feature}_j, \tag{5.3}$$

where $Y = 1$ indicates a process where new supersymmetric particles are produced, and $Y = 0$ otherwise. Table 7 shows the results based on a logistic regression analysis, and Table 8 shows the results based on a probit regression analysis. The results based on the full data set show that 14 of the 18 kinematic features capture the difference between the two processes, with kinematic features 1-12 being highly significant. The proposed repeated perturbation subsampling with $r_n = 10000$, $M = 50$, and $m = 200$ yields a similar result that kinematic features 1-12 are significantly associated with the classification of the two processes. The conditional standard errors are stable across different subset perturbations, and are approximately $\sqrt{na_n/(r_nM)}$ (3.16 for gammaPERT, 3.65 for betaPERT, 4.47 for geomPERT) times the standard errors with the full data set. The unconditional variances of the perturbation estimates are approximately h_n (11 for gammaPERT, 14.31 for betaPERT, 20.96 for geomPERT) times the variances of the estimates obtained with the full data set. However, the computation reduced from $O(nd^2)$ to $O(r_nd^2)$, and required much less memory than the estimation based on the full data set did.

6. Discussion

We have proposed a perturbation subsampling method as a parallel approach to sampling, with or without replacement, for analyzing large-scale data by optimizing convex objective functions. We have also developed a repeated perturbation subsampling method that simultaneously provide a valid estimator and its variance estimator, with relatively little computational effort. Further research is needed for the optimal choices of the stochastic weighting random variable, subsampling size, and repeated perturbation numbers for statistical inferences in different types of data and statistical models.

Supplementary Material

The online Supplementary Material provides proofs for Theorems 1 and 2 and Corollaries 1 and 2.

Acknowledgments

We thank the associate editor and two referees for their insightful comments. This work was supported in part by the National Institue on Aging/National Institutes of Health (U19 AG063893, Long Life Family Study).

References

Ai, M., Yu, J., Zhang, H. and Wang, H. (2021). Optimal subsampling algorithms for big data regressions. *Statistica Sinica* **31**, 749–772.

Baldi, P., Sadowski, P. and Whiteson, D. (2014). Searching for exotic particles in high-energy physics with deep learning. *Nature Communications* **5**, 1–9.

Bograd, S. J. and Lynn, R. J. (2003). Long-term variability in the southern California current system. *Deep Sea Research Part II: Topical Studies in Oceanography* **50**, 2355–2370.

Boyd, S., Parikh, N., Chu, E., Peleato, B. and Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning* **3**, 1–122.

Dhillon, P., Lu, Y., Foster, D. P. and Ungar, L. (2013). New subsampling algorithms for fast least squares regression. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, 360–368. Curran Associates Inc, New York.

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, Pennsylvania.

Jin, Z., Ying, Z. and Wei, L. (2001). A simple resampling method by perturbing the minimand. *Biometrika* **88**, 381–390.

Jordan, M. I., Lee, J. D. and Yang, Y. (2019). Communication-efficient distributed statistical inference. *Journal of the American Statistical Association* **114**, 668–681.

Keret, N. and Gorfine, M. (2020). Optimal Cox regression subsampling procedure with rare events. *arXiv:2012.02122*.

Lichman, M. (2013). *UCI Machine Learning Repository*. University of California, School of Information and Computer Science, Irvine, CA. Web: `https://archive.ics.uci.edu/ml`.

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

Mahoney, M. W. (2011). Randomized algorithms for matrices and data. *Foundations and Trends® in Machine Learning* **3**, 123–224.

McDonald, R., Hall, K. and Mann, G. (2010). Distributed training strategies for the structured perceptron. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 456–464. Association for Computational Linguistics, Pennsylvania.

Niemiro, W. (1992). Asymptotics for M-estimators defined by convex minimization. *The Annals of Statistics* **20**, 1514–1533.

Quiroz, M., Kohn, R., Villani, M. and Tran, M. N. (2019). Speeding up MCMC by efficient data subsampling. *Journal of the American Statistical Association* **114**, 831–843.

R Core Team (2013). *R: A Language and Environment for Statistical Computing*. Vienna, Austria. R Foundation for Statistical Computing. ISBN: 3-900051-07-0. Web: `http://www.R-project.org`.

Särndal, C. E., Swensson, B. and Wretman, J. (2003). *Model Assisted Survey Sampling*. Springer, New York.

Sivasankari, M. and Anandan, R. (2020). Regression analysis on sea surface temperature. In *Intelligent Computing and Innovation on Data Science*, 595–601. Springer, New York.

Terenin, A., Simpson, D. and Draper, D. (2015). Asynchronous gibbs sampling. *arXiv:1509.08999*.

Wang, H. (2019). More efficient estimation for logistic regression with optimal subsamples. *Journal of Machine Learning Research* **20**, 1–59.

Wang, H. and Ma, Y. (2021). Optimal subsampling for quantile regression in big data. *Biometrika* **108**, 99–112.

Wang, H., Zhu, R. and Ma, P. (2018). Optimal subsampling for large sample logistic regression. *Journal of the American Statistical Association* **113**, 829–844.

Yu, J., Wang, H., Ai, M. and Zhang, H. (2020). Optimal distributed subsampling for maximum quasi-likelihood estimators with massive data. *Journal of the American Statistical Association* **117**, 265–276.

Zuo, L., Zhang, H., Wang, H. and Liu, L. (2021). Sampling-based estimation for massive survival data with additive hazards model. *Statistics in Medicine* **40**, 441–450.

Yujing Yao

Department of Biostatistics, Columbia University, New York, NY 10032, USA.

E-mail: yy2725@cumc.columbia.edu

Zhezhen Jin

Department of Biostatistics, Columbia University, New York, NY 10032, USA.

E-mail: zj7@cumc.columbia.edu

(Received January 2022; accepted September 2022)