<table>
<thead>
<tr>
<th><strong>Statistica Sinica Preprint No:</strong> SS-2022-0020</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Title</strong></td>
</tr>
<tr>
<td>A Perturbation Subsampling for Large Scale Data</td>
</tr>
<tr>
<td><strong>Manuscript ID</strong></td>
</tr>
<tr>
<td>SS-2022-0020</td>
</tr>
<tr>
<td><strong>URL</strong></td>
</tr>
<tr>
<td><a href="http://www.stat.sinica.edu.tw/statistica/">http://www.stat.sinica.edu.tw/statistica/</a></td>
</tr>
<tr>
<td><strong>DOI</strong></td>
</tr>
<tr>
<td>10.5705/ss.202022.0020</td>
</tr>
<tr>
<td><strong>Complete List of Authors</strong></td>
</tr>
<tr>
<td>Yujing Yao and Zhezhen Jin</td>
</tr>
<tr>
<td><strong>Corresponding Authors</strong></td>
</tr>
<tr>
<td>Zhezhen Jin</td>
</tr>
<tr>
<td><strong>E-mails</strong></td>
</tr>
<tr>
<td><a href="mailto:zj7@columbia.edu">zj7@columbia.edu</a></td>
</tr>
</tbody>
</table>
A Perturbation Subsampling for Large Scale Data

Yujing Yao and Zhezhen Jin

Department of Biostatistics, 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 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 $X$ is a $d$-dimensional vector of covariates. The relationship between $Y$ and $X$ can be characterized by a $d$-dimensional unknown parameter $\beta_0 \in \mathbb{R}^d$, where

$$\beta_0 = \arg \min_{\beta \in \mathbb{R}^d} Ef(\beta, X, Y),$$  \hspace{1cm} (2.1)

with $f(\beta, x, y)$ being a continuous, convex objective function with respect to $\beta$. Throughout this paper, we assume that $Ef(\beta, X, Y)$ exists and is finite. Based on the independent and
identically distributed (i.i.d.) sample \((y_i, x_i)\), for \(i = 1, \cdots, n\), we can obtain the estimator of \(\beta_0\) by minimizing the empirical analog of the convex objective function:

\[
\hat{\beta}_n = \arg \min_{\beta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\beta, x_i, y_i). \tag{2.2}
\]

In general, the resulting estimator is an \(M\)-estimator [Nemirovski, 1992]. It is the maximum likelihood estimator (MLE) when the function \(f(\beta, x_i, y_i)\) is the minus of the log-likelihood of \((y_i, x_i)\), and is the \(L_p\)-norm estimator if \(f(\beta, x_i, y_i) = |y_i - x_i^T \beta|^p\), for a known \(p > 1\) and \(i = 1, \cdots, n\).

### 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 \(i\)th Bernoulli random variable takes the value one, then the \(i\)th sample is selected, otherwise, the \(i\)th 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 $U_{n,i} \sim \text{Bernoulli}(q_n)$,
   
   \[ q_n = r_n/n. \]

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

3 **Estimation**: Minimize the perturbed objective function to obtain an estimator
   \[ \tilde{\beta}_n = \arg\min_{\beta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^n W_{n,i} f(\beta, x_i, y_i), \text{ where } W_{n,i} = U_{n,i} V_{n,i}. \]  
   \[ (2.3) \]

**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, \( V_n \sim \text{Gamma}(1/q_n, 1) \), \( V_n \sim \text{Exponential}(1/q_n) \);

(b) Scaled Beta distribution, for example, \( V_n \sim 3/q_n \text{Beta}(1, 2) \), \( V_n \sim \text{Uniform}(0, 2/q_n) \);

(c) Half normal distribution, for example, \( V_n \sim \text{Half-Normal}(0, q_n^2 \pi/2) \).

2. Discrete distributions:

(a) Geometric distribution, for example, \( V_n \sim \text{Geometric}(q_n) \);

(b) Negative binomial distribution, for example, \( V_n \sim \text{Negative Binomial}(1/q_n, 1/2) \);

(c) Poisson distribution, for example, \( V_n \sim \text{Poisson}(1/q_n) \).

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.
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 \( U_{n,i} \sim \text{Bernoulli}(q_n) \).

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

\[
E(V_{n,k,i}) = 1/q_n \quad \text{and} \quad \text{var}(V_{n,k,i}) = b_n^2.
\]

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

\[
\tilde{\beta}_{n,k} = \arg \min_{\beta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} W_{n,k,i} f(\beta, x_i, y_i), \quad \text{where} \quad W_{n,k,i} = U_{n,i} V_{n,k,i}.
\] (2.4)

5 **end**

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

\[
\text{var}(\tilde{\beta}_n|D_n) = \frac{1}{b_n q_n (m-1) T} \sum_{k=1}^{m} (\tilde{\beta}_{n,k} - \frac{1}{m} \sum_{k=1}^{m} \tilde{\beta}_{n,k}) (\tilde{\beta}_{n,k} - \frac{1}{m} \sum_{k=1}^{m} \tilde{\beta}_{n,k})^T,
\] (2.5)

where \( D_n = \{x_i, y_i\}_{i=1}^{n} \), and the unconditional variance can be estimated by

\[
\text{var}(\tilde{\beta}_n) = (r_n/(na_n) + 1) \text{var}(\tilde{\beta}_n|D_n), \quad \text{where} \quad a_n = 1 - q_n + b_n^2 q_n^2.
\] (2.6)
2.3 Repeated perturbation subsampling for large-scale data

Algorithm 3: A repeated perturbation subsampling algorithm

1 Prespecify the number of parallels \( M(>1) \):

2 for fixed \( l, l = 1, \ldots, M \) do

3 Subset: Generate \( n \) i.i.d. r.v. \( \{U_{n,i,l}\}_{i=1}^{n} \), where \( U_{n,i,l} \sim \text{Bernoulli}(q_n) \).

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

5 Stochastic weighting: Generate \( n \) i.i.d. nonnegative r.v. \( \{V_{n,i,l,k}\}_{i=1}^{n} \) with

\[
E(V_{n,i,l,k}) = 1/q_n \quad \text{and} \quad \text{var}(V_{n,i,l,k}) = b_n^2.
\]

6 Point estimation: Minimize the perturbed objective function such that

\[
\bar{\beta}_{n,l,k} = \arg \min_{\beta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} W_{n,i,l,k} f(\beta, x_i, y_i), \quad \text{where} \quad W_{n,i,l,k} = U_{n,i,l} V_{n,i,l,k}.
\] (2.7)

7 end

8 Variance estimation: The estimate of \( \text{var}(\bar{\beta}_{n,l}) \) can be obtained as in (2.6).

9 end

10 Combination: Obtain the final estimator as

\[
\bar{\beta}_n^{(M)} = \left( \sum_{l=1}^{M} \text{var}(\bar{\beta}_{n,l})^{-1} \right)^{-1} \sum_{l=1}^{M} \text{var}(\bar{\beta}_{n,l})^{-1} \bar{\beta}_{n,l}.
\] (2.8)

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

\[
\text{var}(\bar{\beta}_n^{(M)} | \mathcal{D}_n) = \frac{1}{M-1} \sum_{l=1}^{M} (\bar{\beta}_{n,l} - \frac{1}{M} \sum_{l=1}^{M} \bar{\beta}_{n,l}) (\bar{\beta}_{n,l} - \frac{1}{M} \sum_{l=1}^{M} \bar{\beta}_{n,l})^T
\]

and

\[
\text{var}(\bar{\beta}_n^{(M)}) = \left( \sum_{l=1}^{M} \bar{e}_l e_l^T \right)^{-1} / (na_n) + I_{d \times d} \text{var}(\bar{\beta}_n^{(M)} | \mathcal{D}_n),
\]

(2.9)

where

\[
\bar{e}_l = \sum_{l=1}^{M} \text{var}(\bar{\beta}_{n,l})^{-1} \text{var}(\bar{\beta}_{n,l})^{-1} \text{and} \quad a_n = 1 - q_n + b_n^2 q_n^2.
\]
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 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 $D_n = \{y_i, x_i\}_{i=1}^n$. 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 $\text{var}^*$ 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 $\beta$ is compact in $\mathbb{R}^d$. The $\beta_0$ satisfying (2.1) is an interior point of the parameter space and is unique.

Assumption 2. The first and second gradients of the convex objective function $f(\beta, x, y)$ with respect to $\beta$ in a neighborhood of $\beta_0$ exist and are finite. The gradients are denoted by $\dot{f}$ and $\ddot{f}$ respectively, given by $\dot{f}(\beta_0, x, y) = \frac{\partial f(\beta, x, y)}{\partial \beta}\bigg|_{\beta=\beta_0}$ and $\ddot{f}(\beta_0, x, y) = \frac{\partial^2 f(\beta, x, y)}{\partial \beta \partial \beta^T}\bigg|_{\beta=\beta_0}$.

We further assume that the matrix $E(\ddot{f}(\beta_0, X, Y))$ is positive definite.

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

Assumption 1 is required to guarantee the consistency of the minimizer of the convex objective function. Assumption 2 guarantees $E(||\dot{f}(\beta, X, Y)||^2) < \infty$ for each $\beta$ in a neighborhood of $\beta_0$, 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 $\limsup_{n \to \infty} q_n^{1+\alpha} EW_n^{2+\alpha} < \infty$, 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

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

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

$$(C_{1n}^{-1}M_{1n}C_{1n}^{-1})^{-1/2}\sqrt{r_n/a_n}(\bar{\beta}_n - \hat{\beta}_n) \xrightarrow{D} N_d(0, I_{d \times d}), \text{ as } r_n \rightarrow \infty, \text{ and } n \rightarrow \infty, \quad (3.2)$$
where $\mathbf{M}_{1n} = \frac{1}{n} \sum_{i=1}^{n} \hat{f}(\hat{\beta}_n, \mathbf{x}_i, y_i) \hat{f}(\hat{\beta}_n, \mathbf{x}_i, y_i)^T$, $\mathbf{C}_{1n} = \frac{1}{n} \sum_{i=1}^{n} \hat{f}(\hat{\beta}_n, \mathbf{x}_i, y_i)$, and $a_n = 1 - q_n + b_n q_n^2$.

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

$$\Pr(||\hat{\beta}_n - \beta_0|| \geq \epsilon) \to 0, \text{ as } r_n \to \infty \text{ and } n \to \infty,$$

for any $\epsilon > 0$, and

$$(C_2^{-1}M_2C_2^{-1})^{-1/2} \sqrt{r_n/d_n}(\hat{\beta}_n - \beta_0) \xrightarrow{D} N_d(0, \mathbf{I}_{d 	imes d}), \text{ as } r_n \to \infty \text{ and } n \to \infty,$$

where $M_2 = E(\hat{f}(\beta_0, \mathbf{X}, Y)' \hat{f}(\beta_0, \mathbf{X}, Y))$, $C_2 = E(\check{f}(\beta_0, \mathbf{X}, Y))$, and $d_n = 1 + b_n q_n^2$.

**Remark 5.** The estimate of the unconditional variance of $\hat{\beta}_n$ 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 $\hat{\beta}_n$, 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,

$$(a_n M) \sum_{l=1}^{M} e_l e_l^T C_{1n}^{-1} M_{1n} C_{1n}^{-1})^{-1/2} \sqrt{r_n M}(\hat{\beta}_n^{(M)} - \hat{\beta}_n) \xrightarrow{D} N_d(0, \mathbf{I}_{d \times d}), \text{ as } r_n \to \infty \text{ and } n \to \infty,$$

(3.5)
where \( e_l = \left( \sum_{l=1}^{M} \text{var}(\tilde{\beta}_{n,l})^{-1} \right)^{-1} \text{var}(\tilde{\beta}_{n,l})^{-1} \), 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 \( r_n M \geq n \):

\[
\left( h_n C_2^{-1} M_2 C_2^{-1} \right)^{-1/2} \sqrt{n} (\tilde{\beta}_n^{(M)} - \beta_0) \xrightarrow{D} N_d(0, I_{d \times d}), \quad \text{as } r_n \to \infty \text{ and } n \to \infty, \tag{3.6}
\]

where \( e_l = \left( \sum_{l=1}^{M} \text{var}(\tilde{\beta}_{n,l})^{-1} \right)^{-1} \text{var}(\tilde{\beta}_{n,l})^{-1} \), and \( h_n = I_{d \times d} + \frac{na_n}{r_n} \sum_{l=1}^{M} e_l e_l^T \).

**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 \( \tilde{\beta}_n^{(M)} \) can be approximated by the ratio of the adjusting factors, that is, \( r_n (\sum_{l=1}^{M} e_l e_l^T)^{-1}/(na_n) + I_{d \times d} \) times the corresponding estimate of the conditional variance of \( \tilde{\beta}_n^{(M)} \). 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:

\[
(C_2^{-1} M_2 C_2^{-1})^{-1/2} \sqrt{n} (\hat{\beta}_n - \beta_0) \xrightarrow{D} N_d(0, I_{d \times d}). \tag{3.7}
\]

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 $X_{n \times d}$ of the regression models is generated from the following distributions:

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

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

3. T3 data from a multivariate $t$-distribution $t_3(0, \Sigma)$, with $\Sigma_{i,j} = 0.5^{|i-j|}$.

4. T5 data from a multivariate $t$-distribution $t_5(0, \Sigma)$, with $\Sigma_{i,j} = 0.5^{|i-j|}$.

The true regression coefficients are set to be a $3 \times 1$ vector of ones ($d = 3$). A total of $N = 1000$ data sets are generated, with a sample size of $n = 10000$ 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 defined as $\frac{1}{N} \sum_{i=1}^{N} ||\hat{\beta}_{n,i} - \beta_0||^2$. The conditional MSE is defined as $\frac{1}{N} \sum_{i=1}^{N} ||\tilde{\beta}_{n,i} - \hat{\beta}_n||^2$. 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 \cite{R Core Team 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 $\text{Gamma}(1/q_n, 1)$ distribution (gammaPERT);
5. Perturbation subsampling estimator based on $2/q_n \text{Beta}(1, 1)$ distribution (betaPERT);
6. Perturbation subsampling estimator based on $\text{Geometric}(q_n)$ 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 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 $m = 1000$ for the linear regression model. The empirical coverage of the 95% confidence interval for $\beta_{10}$ 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...
Figure 1: Empirical MSEs for a linear regression model, based on $N = 1000$ simulations.
Figure 2: Empirical MSEs for a logistic regression model, based on $N = 1000$ simulations.
Figure 3: Empirical MSEs for a probit regression model, based on $N = 1000$ simulations.
Table 1: Empirical coverage probabilities of $\beta_{10}$ 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.

<table>
<thead>
<tr>
<th>Design matrix</th>
<th>Size</th>
<th>$r$</th>
<th>gammaPERT*</th>
<th>betaPERT*</th>
<th>geomPERT*</th>
<th>gammaPERT**</th>
<th>betaPERT**</th>
<th>geomPERT**</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean zero</td>
<td>1000</td>
<td>0.945</td>
<td>0.915</td>
<td>0.850</td>
<td>0.947</td>
<td>0.897</td>
<td>0.846</td>
<td></td>
</tr>
<tr>
<td>normal</td>
<td>1200</td>
<td>0.947</td>
<td>0.912</td>
<td>0.868</td>
<td>0.950</td>
<td>0.924</td>
<td>0.850</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.956</td>
<td>0.942</td>
<td>0.872</td>
<td>0.943</td>
<td>0.941</td>
<td>0.863</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1800</td>
<td>0.950</td>
<td>0.935</td>
<td>0.877</td>
<td>0.951</td>
<td>0.922</td>
<td>0.867</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.949</td>
<td>0.940</td>
<td>0.865</td>
<td>0.958</td>
<td>0.932</td>
<td>0.871</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.939</td>
<td>0.943</td>
<td>0.885</td>
<td>0.946</td>
<td>0.928</td>
<td>0.889</td>
<td></td>
</tr>
<tr>
<td>Mean nonzero</td>
<td>1000</td>
<td>0.949</td>
<td>0.926</td>
<td>0.849</td>
<td>0.950</td>
<td>0.913</td>
<td>0.843</td>
<td></td>
</tr>
<tr>
<td>normal</td>
<td>1200</td>
<td>0.949</td>
<td>0.920</td>
<td>0.871</td>
<td>0.955</td>
<td>0.917</td>
<td>0.836</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.962</td>
<td>0.944</td>
<td>0.873</td>
<td>0.951</td>
<td>0.939</td>
<td>0.867</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1800</td>
<td>0.954</td>
<td>0.927</td>
<td>0.875</td>
<td>0.951</td>
<td>0.927</td>
<td>0.879</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.948</td>
<td>0.935</td>
<td>0.872</td>
<td>0.951</td>
<td>0.936</td>
<td>0.874</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.941</td>
<td>0.945</td>
<td>0.886</td>
<td>0.943</td>
<td>0.937</td>
<td>0.891</td>
<td></td>
</tr>
<tr>
<td>T3</td>
<td>1000</td>
<td>0.942</td>
<td>0.922</td>
<td>0.837</td>
<td>0.950</td>
<td>0.925</td>
<td>0.854</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.952</td>
<td>0.919</td>
<td>0.868</td>
<td>0.925</td>
<td>0.926</td>
<td>0.869</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.950</td>
<td>0.916</td>
<td>0.871</td>
<td>0.939</td>
<td>0.934</td>
<td>0.866</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1800</td>
<td>0.951</td>
<td>0.927</td>
<td>0.868</td>
<td>0.948</td>
<td>0.936</td>
<td>0.881</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.945</td>
<td>0.925</td>
<td>0.872</td>
<td>0.950</td>
<td>0.937</td>
<td>0.877</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.944</td>
<td>0.944</td>
<td>0.910</td>
<td>0.936</td>
<td>0.936</td>
<td>0.907</td>
<td></td>
</tr>
<tr>
<td>T5</td>
<td>1000</td>
<td>0.948</td>
<td>0.923</td>
<td>0.859</td>
<td>0.944</td>
<td>0.936</td>
<td>0.847</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.958</td>
<td>0.923</td>
<td>0.860</td>
<td>0.932</td>
<td>0.935</td>
<td>0.841</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.963</td>
<td>0.926</td>
<td>0.867</td>
<td>0.952</td>
<td>0.919</td>
<td>0.865</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1800</td>
<td>0.950</td>
<td>0.926</td>
<td>0.875</td>
<td>0.946</td>
<td>0.918</td>
<td>0.886</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.944</td>
<td>0.938</td>
<td>0.864</td>
<td>0.952</td>
<td>0.921</td>
<td>0.896</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.953</td>
<td>0.945</td>
<td>0.905</td>
<td>0.954</td>
<td>0.947</td>
<td>0.912</td>
<td></td>
</tr>
</tbody>
</table>
Table 2: Empirical coverage probabilities of $\beta_{10}$ 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.

<table>
<thead>
<tr>
<th>Design matrix</th>
<th>Size $r$</th>
<th>gammaPERT*</th>
<th>betaPERT*</th>
<th>geomPERT*</th>
<th>gammaPERT**</th>
<th>betaPERT**</th>
<th>geomPERT**</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean zero</td>
<td>1000</td>
<td>0.956</td>
<td>0.924</td>
<td>0.848</td>
<td>0.947</td>
<td>0.909</td>
<td>0.857</td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.941</td>
<td>0.932</td>
<td>0.864</td>
<td>0.954</td>
<td>0.924</td>
<td>0.844</td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.941</td>
<td>0.921</td>
<td>0.881</td>
<td>0.950</td>
<td>0.940</td>
<td>0.848</td>
</tr>
<tr>
<td></td>
<td>1800</td>
<td>0.953</td>
<td>0.925</td>
<td>0.872</td>
<td>0.941</td>
<td>0.922</td>
<td>0.872</td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.957</td>
<td>0.931</td>
<td>0.881</td>
<td>0.952</td>
<td>0.924</td>
<td>0.885</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.939</td>
<td>0.936</td>
<td>0.884</td>
<td>0.943</td>
<td>0.934</td>
<td>0.908</td>
</tr>
<tr>
<td>Mean nonzero</td>
<td>1000</td>
<td>0.946</td>
<td>0.932</td>
<td>0.847</td>
<td>0.950</td>
<td>0.931</td>
<td>0.845</td>
</tr>
<tr>
<td>normal</td>
<td>1200</td>
<td>0.938</td>
<td>0.927</td>
<td>0.845</td>
<td>0.963</td>
<td>0.924</td>
<td>0.868</td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.948</td>
<td>0.930</td>
<td>0.862</td>
<td>0.957</td>
<td>0.941</td>
<td>0.877</td>
</tr>
<tr>
<td></td>
<td>1800</td>
<td>0.944</td>
<td>0.937</td>
<td>0.878</td>
<td>0.960</td>
<td>0.918</td>
<td>0.871</td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.961</td>
<td>0.935</td>
<td>0.885</td>
<td>0.951</td>
<td>0.912</td>
<td>0.869</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.953</td>
<td>0.946</td>
<td>0.911</td>
<td>0.949</td>
<td>0.936</td>
<td>0.907</td>
</tr>
<tr>
<td>T3</td>
<td>1000</td>
<td>0.942</td>
<td>0.924</td>
<td>0.837</td>
<td>0.945</td>
<td>0.931</td>
<td>0.854</td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.943</td>
<td>0.945</td>
<td>0.863</td>
<td>0.937</td>
<td>0.915</td>
<td>0.860</td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.952</td>
<td>0.944</td>
<td>0.863</td>
<td>0.949</td>
<td>0.921</td>
<td>0.874</td>
</tr>
<tr>
<td></td>
<td>1800</td>
<td>0.961</td>
<td>0.922</td>
<td>0.864</td>
<td>0.947</td>
<td>0.934</td>
<td>0.849</td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.955</td>
<td>0.947</td>
<td>0.859</td>
<td>0.941</td>
<td>0.936</td>
<td>0.867</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.945</td>
<td>0.929</td>
<td>0.894</td>
<td>0.944</td>
<td>0.935</td>
<td>0.900</td>
</tr>
<tr>
<td>T5</td>
<td>1000</td>
<td>0.950</td>
<td>0.922</td>
<td>0.852</td>
<td>0.943</td>
<td>0.932</td>
<td>0.838</td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.954</td>
<td>0.920</td>
<td>0.855</td>
<td>0.950</td>
<td>0.906</td>
<td>0.859</td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.961</td>
<td>0.934</td>
<td>0.844</td>
<td>0.961</td>
<td>0.926</td>
<td>0.861</td>
</tr>
<tr>
<td></td>
<td>1800</td>
<td>0.942</td>
<td>0.928</td>
<td>0.862</td>
<td>0.942</td>
<td>0.936</td>
<td>0.866</td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.957</td>
<td>0.939</td>
<td>0.882</td>
<td>0.945</td>
<td>0.919</td>
<td>0.880</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.954</td>
<td>0.947</td>
<td>0.910</td>
<td>0.946</td>
<td>0.957</td>
<td>0.891</td>
</tr>
</tbody>
</table>
level. The empirical coverage remains stable when the perturbation number increases from $m = 500$ to $m = 1000$. 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);

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

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

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

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

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

The A-optimal and L-optimal nonuniform sampling probabilities are obtained using the procedures in \cite{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 $X_{n \times d}$ is generated from the $t$-distribution. The repeated perturbation subsampling estimators have much smaller MSEs. Increasing $M$ from 20 to 50 yields
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 $N = 1000$ simulations.
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 $N = 1000$ simulations.
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 $N = 1000$ simulations across all subsample sizes and four design matrices.

<table>
<thead>
<tr>
<th>Method</th>
<th>unisMUL</th>
<th>A-optsMUL</th>
<th>L-optsMUL</th>
<th>gammaPERT</th>
<th>betaPERT</th>
<th>geomPERT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU time(s)</td>
<td>0.008</td>
<td>4.576</td>
<td>4.231</td>
<td>1.368</td>
<td>1.632</td>
<td>1.414</td>
</tr>
</tbody>
</table>

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 $\beta_{10}$ 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.

5. Application

5.1 Oceanographic data

In this section, we apply our proposed procedure to analyze the California Cooperative Oceanic Fisheries Investigations (CalCOFI) Hydrographic data set. The Hydrographic
Table 4: Empirical coverage probabilities of $\beta_{10}$ 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 $N=1000$ simulations.

<table>
<thead>
<tr>
<th>Design matrix</th>
<th>Size</th>
<th>$r$</th>
<th>gammaPERT*</th>
<th>betaPERT*</th>
<th>geomPERT*</th>
<th>gammaPERT**</th>
<th>betaPERT**</th>
<th>geomPERT**</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean zero</td>
<td>500</td>
<td>0.958</td>
<td>0.950</td>
<td>0.932</td>
<td>0.962</td>
<td>0.960</td>
<td>0.954</td>
<td></td>
</tr>
<tr>
<td>normal</td>
<td>1000</td>
<td>0.932</td>
<td>0.950</td>
<td>0.946</td>
<td>0.956</td>
<td>0.954</td>
<td>0.948</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.954</td>
<td>0.960</td>
<td>0.948</td>
<td>0.952</td>
<td>0.960</td>
<td>0.948</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.954</td>
<td>0.952</td>
<td>0.938</td>
<td>0.956</td>
<td>0.962</td>
<td>0.964</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.944</td>
<td>0.950</td>
<td>0.948</td>
<td>0.962</td>
<td>0.954</td>
<td>0.958</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.942</td>
<td>0.954</td>
<td>0.950</td>
<td>0.958</td>
<td>0.952</td>
<td>0.956</td>
<td></td>
</tr>
<tr>
<td>Mean nonzero</td>
<td>500</td>
<td>0.956</td>
<td>0.958</td>
<td>0.936</td>
<td>0.950</td>
<td>0.964</td>
<td>0.946</td>
<td></td>
</tr>
<tr>
<td>normal</td>
<td>1000</td>
<td>0.936</td>
<td>0.940</td>
<td>0.936</td>
<td>0.954</td>
<td>0.958</td>
<td>0.944</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.952</td>
<td>0.954</td>
<td>0.936</td>
<td>0.944</td>
<td>0.950</td>
<td>0.952</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.958</td>
<td>0.956</td>
<td>0.948</td>
<td>0.950</td>
<td>0.956</td>
<td>0.952</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.934</td>
<td>0.946</td>
<td>0.948</td>
<td>0.954</td>
<td>0.948</td>
<td>0.956</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.942</td>
<td>0.954</td>
<td>0.946</td>
<td>0.956</td>
<td>0.942</td>
<td>0.954</td>
<td></td>
</tr>
<tr>
<td>T3</td>
<td>500</td>
<td>0.974</td>
<td>0.976</td>
<td>0.948</td>
<td>0.978</td>
<td>0.978</td>
<td>0.974</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>0.962</td>
<td>0.958</td>
<td>0.956</td>
<td>0.970</td>
<td>0.966</td>
<td>0.964</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.958</td>
<td>0.972</td>
<td>0.964</td>
<td>0.966</td>
<td>0.974</td>
<td>0.966</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.966</td>
<td>0.964</td>
<td>0.952</td>
<td>0.972</td>
<td>0.968</td>
<td>0.968</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.960</td>
<td>0.956</td>
<td>0.944</td>
<td>0.958</td>
<td>0.958</td>
<td>0.962</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.958</td>
<td>0.968</td>
<td>0.956</td>
<td>0.962</td>
<td>0.960</td>
<td>0.964</td>
<td></td>
</tr>
<tr>
<td>T5</td>
<td>500</td>
<td>0.964</td>
<td>0.950</td>
<td>0.934</td>
<td>0.962</td>
<td>0.952</td>
<td>0.956</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>0.962</td>
<td>0.950</td>
<td>0.962</td>
<td>0.960</td>
<td>0.958</td>
<td>0.958</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.950</td>
<td>0.954</td>
<td>0.954</td>
<td>0.960</td>
<td>0.950</td>
<td>0.964</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.956</td>
<td>0.956</td>
<td>0.952</td>
<td>0.954</td>
<td>0.966</td>
<td>0.952</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.960</td>
<td>0.946</td>
<td>0.960</td>
<td>0.956</td>
<td>0.954</td>
<td>0.958</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.950</td>
<td>0.954</td>
<td>0.942</td>
<td>0.956</td>
<td>0.952</td>
<td>0.960</td>
<td></td>
</tr>
</tbody>
</table>
Table 5: Empirical coverage probabilities of $\beta_{10}$ 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 $N = 1000$ simulations.

<table>
<thead>
<tr>
<th>Design matrix</th>
<th>Size</th>
<th>$r$</th>
<th>gammaPERT*</th>
<th>betaPERT*</th>
<th>geomPERT*</th>
<th>gammaPERT**</th>
<th>betaPERT**</th>
<th>geomPERT**</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean zero</td>
<td>500</td>
<td>0.946</td>
<td>0.964</td>
<td>0.958</td>
<td>0.948</td>
<td>0.944</td>
<td>0.964</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>0.948</td>
<td>0.954</td>
<td>0.964</td>
<td>0.952</td>
<td>0.946</td>
<td>0.964</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.950</td>
<td>0.952</td>
<td>0.950</td>
<td>0.964</td>
<td>0.952</td>
<td>0.950</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.962</td>
<td>0.946</td>
<td>0.966</td>
<td>0.950</td>
<td>0.954</td>
<td>0.956</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.952</td>
<td>0.954</td>
<td>0.966</td>
<td>0.954</td>
<td>0.958</td>
<td>0.952</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.952</td>
<td>0.952</td>
<td>0.956</td>
<td>0.952</td>
<td>0.952</td>
<td>0.952</td>
<td></td>
</tr>
<tr>
<td>Mean nonzero</td>
<td>500</td>
<td>0.926</td>
<td>0.926</td>
<td>0.944</td>
<td>0.916</td>
<td>0.932</td>
<td>0.944</td>
<td></td>
</tr>
<tr>
<td>normal</td>
<td>1000</td>
<td>0.938</td>
<td>0.938</td>
<td>0.950</td>
<td>0.948</td>
<td>0.952</td>
<td>0.950</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.948</td>
<td>0.938</td>
<td>0.950</td>
<td>0.942</td>
<td>0.958</td>
<td>0.956</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.946</td>
<td>0.944</td>
<td>0.944</td>
<td>0.952</td>
<td>0.956</td>
<td>0.962</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.950</td>
<td>0.944</td>
<td>0.956</td>
<td>0.946</td>
<td>0.956</td>
<td>0.948</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.950</td>
<td>0.964</td>
<td>0.950</td>
<td>0.956</td>
<td>0.954</td>
<td>0.960</td>
<td></td>
</tr>
<tr>
<td>T3</td>
<td>500</td>
<td>0.952</td>
<td>0.952</td>
<td>0.956</td>
<td>0.948</td>
<td>0.946</td>
<td>0.960</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>0.942</td>
<td>0.956</td>
<td>0.938</td>
<td>0.948</td>
<td>0.942</td>
<td>0.950</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.928</td>
<td>0.944</td>
<td>0.950</td>
<td>0.940</td>
<td>0.948</td>
<td>0.944</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.928</td>
<td>0.938</td>
<td>0.946</td>
<td>0.930</td>
<td>0.938</td>
<td>0.940</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.944</td>
<td>0.936</td>
<td>0.934</td>
<td>0.936</td>
<td>0.934</td>
<td>0.936</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.920</td>
<td>0.926</td>
<td>0.926</td>
<td>0.928</td>
<td>0.928</td>
<td>0.930</td>
<td></td>
</tr>
<tr>
<td>T5</td>
<td>500</td>
<td>0.954</td>
<td>0.962</td>
<td>0.964</td>
<td>0.964</td>
<td>0.960</td>
<td>0.968</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>0.952</td>
<td>0.960</td>
<td>0.948</td>
<td>0.956</td>
<td>0.966</td>
<td>0.970</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>0.956</td>
<td>0.964</td>
<td>0.954</td>
<td>0.962</td>
<td>0.968</td>
<td>0.966</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.956</td>
<td>0.950</td>
<td>0.972</td>
<td>0.962</td>
<td>0.964</td>
<td>0.952</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.950</td>
<td>0.948</td>
<td>0.952</td>
<td>0.966</td>
<td>0.962</td>
<td>0.968</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.954</td>
<td>0.960</td>
<td>0.956</td>
<td>0.966</td>
<td>0.960</td>
<td>0.962</td>
<td></td>
</tr>
</tbody>
</table>
data set provides the longest duration oceanographic data since 1949, and is available at https://calcofi.org/ccdata.html. The current data set has 887,018 records, with \( n = 682,876 \) complete cases. We use the complete cases to examine the effect of 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 \( \sqrt{na_n/(r_nM)} \) (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.

### 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
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 $r_n = 10000$ are presented.

<table>
<thead>
<tr>
<th></th>
<th>Full</th>
<th>gammaPERT</th>
<th>betaPERT</th>
<th>geomPERT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Coef.</td>
<td>SE</td>
<td>Coef. (Cond. SE)</td>
<td>SE</td>
</tr>
<tr>
<td>Intercept</td>
<td>-141.7</td>
<td>0.370</td>
<td>-141.9(0.750)</td>
<td>0.983</td>
</tr>
<tr>
<td>Salinity</td>
<td>4.269</td>
<td>0.011</td>
<td>4.275(0.022)</td>
<td>0.029</td>
</tr>
<tr>
<td>Oxygen</td>
<td>2.406</td>
<td>0.002</td>
<td>2.404(0.004)</td>
<td>0.006</td>
</tr>
</tbody>
</table>

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. 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:

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

$$
\text{probit}(Pr\{Y = 1\}) = \beta_0 + \sum_{j=1}^{18} \beta_j \text{kinematic feature}_j, \quad (5.3)
$$
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 $O(r_n d^2)$, 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 objec-
tive 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 statis-
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 $r_n = 10000$ are presented.

<table>
<thead>
<tr>
<th>Kinematic Feathers</th>
<th>Full Coef.</th>
<th>SE</th>
<th>Coef. (Cond. SE)</th>
<th>SE</th>
<th>gammaPERT Coef. (Cond. SE)</th>
<th>SE</th>
<th>betaPERT Coef. (Cond. SE)</th>
<th>SE</th>
<th>geomPERT Coef. (Cond. SE)</th>
<th>SE</th>
</tr>
</thead>
<tbody>
<tr>
<td>feature1</td>
<td>4.835</td>
<td>0.010</td>
<td>4.711(0.033)</td>
<td>0.034</td>
<td>4.742(0.039)</td>
<td>0.040</td>
<td>4.725(0.053)</td>
<td>0.054</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature2</td>
<td>2.326</td>
<td>0.009</td>
<td>2.316(0.031)</td>
<td>0.033</td>
<td>2.362(0.034)</td>
<td>0.035</td>
<td>2.387(0.047)</td>
<td>0.049</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature3</td>
<td>-1.714</td>
<td>0.008</td>
<td>-1.756(0.026)</td>
<td>0.028</td>
<td>-1.694(0.034)</td>
<td>0.035</td>
<td>-1.724(0.041)</td>
<td>0.042</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature4</td>
<td>-0.623</td>
<td>0.005</td>
<td>-0.617(0.014)</td>
<td>0.014</td>
<td>-0.623(0.017)</td>
<td>0.017</td>
<td>-0.623(0.023)</td>
<td>0.023</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature5</td>
<td>-1.601</td>
<td>0.014</td>
<td>-1.643(0.054)</td>
<td>0.056</td>
<td>-1.629(0.054)</td>
<td>0.056</td>
<td>-1.621(0.066)</td>
<td>0.068</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature6</td>
<td>-0.410</td>
<td>0.004</td>
<td>-0.400(0.013)</td>
<td>0.014</td>
<td>-0.395(0.016)</td>
<td>0.017</td>
<td>-0.440(0.020)</td>
<td>0.020</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature7</td>
<td>0.470</td>
<td>0.005</td>
<td>0.501(0.015)</td>
<td>0.016</td>
<td>0.462(0.020)</td>
<td>0.021</td>
<td>0.479(0.022)</td>
<td>0.023</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature8</td>
<td>1.106</td>
<td>0.012</td>
<td>1.120(0.038)</td>
<td>0.039</td>
<td>0.990(0.051)</td>
<td>0.053</td>
<td>1.059(0.054)</td>
<td>0.055</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature9</td>
<td>0.317</td>
<td>0.005</td>
<td>0.317(0.016)</td>
<td>0.017</td>
<td>0.293(0.018)</td>
<td>0.018</td>
<td>0.294(0.019)</td>
<td>0.019</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature10</td>
<td>-2.038</td>
<td>0.034</td>
<td>-2.133(0.121)</td>
<td>0.127</td>
<td>-2.214(0.126)</td>
<td>0.131</td>
<td>-2.050(0.132)</td>
<td>0.135</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature11</td>
<td>0.533</td>
<td>0.009</td>
<td>0.546(0.038)</td>
<td>0.040</td>
<td>0.480(0.036)</td>
<td>0.037</td>
<td>0.562(0.051)</td>
<td>0.052</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature12</td>
<td>0.098</td>
<td>0.005</td>
<td>0.098(0.02)</td>
<td>0.021</td>
<td>0.099(0.017)</td>
<td>0.018</td>
<td>0.096(0.026)</td>
<td>0.026</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature13</td>
<td>0.204</td>
<td>0.036</td>
<td>0.274(0.134)</td>
<td>0.141</td>
<td>0.371(0.135)</td>
<td>0.140</td>
<td>0.169(0.148)</td>
<td>0.151</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature14</td>
<td>0.004</td>
<td>0.001</td>
<td>0.005(0.004)</td>
<td>0.004</td>
<td>-0.001(0.004)</td>
<td>0.004</td>
<td>0.005(0.005)</td>
<td>0.005</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature15</td>
<td>-0.002</td>
<td>0.001</td>
<td>-0.002(0.003)</td>
<td>0.003</td>
<td>0.001(0.005)</td>
<td>0.005</td>
<td>0.002(0.006)</td>
<td>0.006</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature16</td>
<td>0.001</td>
<td>0.001</td>
<td>0.005(0.004)</td>
<td>0.005</td>
<td>-0.002(0.004)</td>
<td>0.004</td>
<td>0.014(0.005)</td>
<td>0.005</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature17</td>
<td>-0.0002</td>
<td>0.001</td>
<td>-0.0001(0.004)</td>
<td>0.005</td>
<td>0.001(0.004)</td>
<td>0.004</td>
<td>-0.011(0.004)</td>
<td>0.005</td>
<td></td>
<td></td>
</tr>
<tr>
<td>feature18</td>
<td>-0.0001</td>
<td>0.001</td>
<td>-0.004(0.003)</td>
<td>0.003</td>
<td>-0.002(0.005)</td>
<td>0.005</td>
<td>-0.004(0.006)</td>
<td>0.006</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
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 $r_n = 10000$ are presented.

<table>
<thead>
<tr>
<th>Kinematic Feathers</th>
<th>Coef.</th>
<th>SE</th>
<th>Coef. (Cond. SE)</th>
<th>SE</th>
<th>Coef. (Cond. SE)</th>
<th>SE</th>
<th>Coef. (Cond. SE)</th>
<th>SE</th>
</tr>
</thead>
<tbody>
<tr>
<td>feature1</td>
<td>2.665</td>
<td>0.006</td>
<td>2.689(0.022)</td>
<td>0.023</td>
<td>2.727(0.025)</td>
<td>0.026</td>
<td>2.717(0.036)</td>
<td>0.037</td>
</tr>
<tr>
<td>feature2</td>
<td>-1.065</td>
<td>0.004</td>
<td>-1.087(0.017)</td>
<td>0.017</td>
<td>-1.047(0.021)</td>
<td>0.022</td>
<td>-1.074(0.026)</td>
<td>0.027</td>
</tr>
<tr>
<td>feature3</td>
<td>1.274</td>
<td>0.005</td>
<td>1.272(0.020)</td>
<td>0.020</td>
<td>1.309(0.022)</td>
<td>0.023</td>
<td>1.317(0.031)</td>
<td>0.031</td>
</tr>
<tr>
<td>feature4</td>
<td>-1.041</td>
<td>0.008</td>
<td>-1.067(0.037)</td>
<td>0.038</td>
<td>-1.065(0.038)</td>
<td>0.039</td>
<td>-1.073(0.049)</td>
<td>0.050</td>
</tr>
<tr>
<td>feature5</td>
<td>0.868</td>
<td>0.007</td>
<td>0.859(0.027)</td>
<td>0.028</td>
<td>0.759(0.037)</td>
<td>0.038</td>
<td>0.821(0.039)</td>
<td>0.040</td>
</tr>
<tr>
<td>feature6</td>
<td>-0.329</td>
<td>0.003</td>
<td>-0.328(0.009)</td>
<td>0.009</td>
<td>-0.331(0.011)</td>
<td>0.011</td>
<td>-0.328(0.015)</td>
<td>0.015</td>
</tr>
<tr>
<td>feature7</td>
<td>-0.206</td>
<td>0.002</td>
<td>-0.200(0.008)</td>
<td>0.009</td>
<td>-0.210(0.011)</td>
<td>0.012</td>
<td>-0.234(0.013)</td>
<td>0.013</td>
</tr>
<tr>
<td>feature8</td>
<td>0.299</td>
<td>0.003</td>
<td>0.320(0.010)</td>
<td>0.011</td>
<td>0.289(0.013)</td>
<td>0.014</td>
<td>0.310(0.015)</td>
<td>0.015</td>
</tr>
<tr>
<td>feature9</td>
<td>0.221</td>
<td>0.003</td>
<td>0.216(0.010)</td>
<td>0.011</td>
<td>0.199(0.012)</td>
<td>0.012</td>
<td>0.202(0.014)</td>
<td>0.014</td>
</tr>
<tr>
<td>feature10</td>
<td>0.333</td>
<td>0.005</td>
<td>0.346(0.026)</td>
<td>0.027</td>
<td>0.310(0.027)</td>
<td>0.028</td>
<td>0.361(0.038)</td>
<td>0.039</td>
</tr>
<tr>
<td>feature11</td>
<td>-0.964</td>
<td>0.019</td>
<td>-1.062(0.078)</td>
<td>0.082</td>
<td>-1.107(0.086)</td>
<td>0.089</td>
<td>-1.009(0.091)</td>
<td>0.092</td>
</tr>
<tr>
<td>feature12</td>
<td>0.096</td>
<td>0.003</td>
<td>0.085(0.013)</td>
<td>0.013</td>
<td>0.086(0.012)</td>
<td>0.013</td>
<td>0.089(0.019)</td>
<td>0.019</td>
</tr>
<tr>
<td>feature13</td>
<td>-0.061</td>
<td>0.020</td>
<td>0.019(0.087)</td>
<td>0.091</td>
<td>0.070(0.091)</td>
<td>0.094</td>
<td>-0.052(0.099)</td>
<td>0.101</td>
</tr>
<tr>
<td>feature14</td>
<td>0.002</td>
<td>0.001</td>
<td>0.003(0.002)</td>
<td>0.002</td>
<td>-0.002(0.002)</td>
<td>0.002</td>
<td>0.001(0.003)</td>
<td>0.003</td>
</tr>
<tr>
<td>feature15</td>
<td>-0.001</td>
<td>0.001</td>
<td>-0.001(0.002)</td>
<td>0.002</td>
<td>-0.0001(0.003)</td>
<td>0.003</td>
<td>-0.0004(0.004)</td>
<td>0.004</td>
</tr>
<tr>
<td>feature16</td>
<td>0.001</td>
<td>0.001</td>
<td>0.003(0.003)</td>
<td>0.003</td>
<td>-0.001(0.002)</td>
<td>0.002</td>
<td>0.006(0.003)</td>
<td>0.003</td>
</tr>
<tr>
<td>feature17</td>
<td>-0.0002</td>
<td>0.001</td>
<td>-0.002(0.002)</td>
<td>0.002</td>
<td>-0.001(0.003)</td>
<td>0.003</td>
<td>-0.002(0.004)</td>
<td>0.004</td>
</tr>
<tr>
<td>feature18</td>
<td>-0.0001</td>
<td>0.001</td>
<td>-0.0001(0.003)</td>
<td>0.003</td>
<td>0.000(0.002)</td>
<td>0.002</td>
<td>-0.006(0.003)</td>
<td>0.003</td>
</tr>
</tbody>
</table>
tical 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 Institute on Aging/National Institutes of Health (U19 AG063893, Long Life Family Study)

References


Yujing Yao

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

E-mail: yy2725@cumc.columbia.edu

Corresponding author: Zhezhen Jin

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

E-mail: zj7@cumc.columbia.edu

Phone: (212) 305-9404

Fax: (212) 305-9408