Knowledge-Based Systems 286 (2024) 111426

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Optimal subsampling for high-dimensional ridge regression ☆

Hanyu Li *, Chengmei Niu

College of Mathematics and Statistics, Chongqing University, Chongqing 401331, PR China

ARTICLE INFO

MSC:
62J07

Keywords:
High-dimensional ridge regression
Optimal subsampling
A-optimal design criterion
Two step iterative algorithm

ABSTRACT

We investigate the feature compression of high-dimensional ridge regression using the optimal subsampling technique. Specifically, based on the basic framework of random sampling algorithm on feature for ridge regression and the A-optimal design criterion, we first obtain a set of optimal subsampling probabilities. Considering that the obtained probabilities are uneconomical, we then propose the nearly optimal ones. With these probabilities, a two step iterative algorithm is established which has lower computational cost and higher accuracy. We provide theoretical analysis and numerical experiments to support the proposed methods. Numerical results demonstrate the decent performance of our methods.

1. Introduction

For the famous linear model

$$y = A\beta + \upsilon,$$

where $y \in \mathbb{R}^n$ is the response vector, $A \in \mathbb{R}^{n\times p}$ is the design matrix, $\beta \in \mathbb{R}^p$ is the parameter vector, and $\upsilon \in \mathbb{R}^n$ is the standardized Gaussian noise vector, ridge regression [1], also known as the least squares regression with Tikhonov regularization [2], has the following form

$$\min_{\beta} \frac{1}{2}\|y - A\beta\|_2^2 + \frac{\lambda}{2}\|\beta\|_2^2, \tag{1.1}$$

where λ is the regularized parameter, and the corresponding estimator is

$$\hat{\beta}_{rls} = (A^T A + \lambda I)^{-1} A^T y.$$

In this paper, we focus only on the case $p > n$, i.e., the high-dimensional ridge regression. For this case, the dominant computational cost of the above estimator is from the matrix inversion which takes $O(p^3)$ flops by direct computation. One way to reduce the cost is to use the Sherman–Morrison–Woodbury formula, which leads to $O(n^2 p)$ flops. Another straightforward way of amelioration is to solve the problem (1.1) in the dual space. Specifically, we first solve the dual problem of (1.1),

$$\min_{z} \frac{1}{2\lambda}\|A^T z\|_2^2 + \frac{1}{2}\|z\|_2^2 - z^T y, \tag{1.2}$$

and the solution is

$$\hat{z}^* = \lambda(AA^T + \lambda I)^{-1} y. \tag{1.3}$$

Then, setting

$$\hat{\beta}_{rls} = \frac{A^T \hat{z}^*}{\lambda} \tag{1.4}$$

gives the estimator of (1.1) in an alternative form

$$\hat{\beta}_{rls} = A^T(AA^T + \lambda I)^{-1} y. \tag{1.5}$$

More details can be found in [3]. Now, the dominant computational cost is also $O(n^2 p)$ which appears in the computation of AA^T. However, it is still prohibitive when $p \gg n$.

To reduce the computational cost, some scholars considered the randomized sketching technique [4–9]. The main idea is to compress the design matrix A to be a small one $\hat{A}$ by post-multiplying it by a random matrix $S \in \mathbb{R}^{p\times r}$ with $r \ll p$, i.e., $\hat{A} = AS$, and hence the reduced regression can be called the compressed ridge regression. There are two most common ways to generate S: random projection and random sampling. The former can be the (sub)Gaussian matrix [6,7,9], the sub-sampled randomized Hadamard transform (SRHT) [4–7,9], the sub-sampled randomized Fourier transform [7], and the CountSketch (also called the sparse embedding matrix) [6], and the latter can be the uniform sampling and the importance sampling [8].

Specifically, building on (1.3) and (1.4), Lu et al. [4] presented the following estimator

$$\hat{\beta}_L = \frac{SS^T A^T \tilde{z}_L}{\lambda},$$

where S is the SRHT and

$$\tilde{z}_L = \lambda(ASS^T A^T + \lambda I)^{-1} y \tag{1.6}$$

☆ The work was supported by the National Natural Science Foundation of China (No. 11671060) and the Natural Science Foundation Project of CQ CSTC, China (No. cstc2019jcyj-msxmX0267).

* Corresponding author.
E-mail addresses: hyli@cqu.edu.cn (H. Li), chengmeiniu@cqu.edu.cn (C. Niu).

https://doi.org/10.1016/j.knosys.2024.111426
Received 18 April 2022; Received in revised form 9 September 2023; Accepted 19 January 2024
Available online 20 January 2024

is the solution to the dual problem of the following compressed ridge regression

$$\min_{\beta_H} \frac{1}{2}\|y - AS\beta_H\|_2^2 + \frac{\lambda}{2}\|\beta_H\|_2^2, \tag{1.7}$$

and obtained a risk bound. Soon afterwards, for S generated by the product of sparse embedding matrix and SRHT, Chen et al. [5] developed an estimator as follows:

$$\hat{\beta}_C = A^T(AS)^{\dagger T}(\lambda(AS)^{\dagger T} + AS)^{\dagger} y, \tag{1.8}$$

where † denotes the Moore–Penrose inverse, and provided an estimation error bound and a risk bound. Later, Avron et al. [6] proposed the estimator $\hat{\beta}_A = A^T\hat{b}$, where

$$\hat{b} = \operatorname{argmin}_b \frac{1}{2}\|ASS^TA^Tb\|_2^2 - y^TAA^Tb + \frac{1}{2}\|y\|_2^2 + \frac{\lambda}{2}\|S^TA^Tb\|_2^2$$

with S being the CountSketch, SRHT, or Gaussian matrix. The above problem is the sketch of the following regression problem

$$\min_b \frac{1}{2}\|AA^Tb\|_2^2 - y^TAA^Tb + \frac{1}{2}\|y\|_2^2 + \frac{\lambda}{2}\|A^Tb\|_2^2,$$

which is transformed from (1.1). Additionally, Wang et al. [7] and Lacotte and Pilanci [9] applied the dual random projection proposed in [10,11] to the high-dimensional ridge regression. By the way, there are some works on compressed least squares regression [12–19], which can be written in the following form

$$\hat{\alpha}_{ls} = \operatorname{argmin}_{\alpha} \frac{1}{2}\|y - AS\alpha\|_2^2, \tag{1.9}$$

where S is typically the (sub)Gaussian matrix.

To the best of our knowledge, there is few work of applying random sampling to high-dimensional ridge regression. We only found a work of [8], which proposed an iterative algorithm by using the random sampling with the column leverage scores or ridge leverage scores as the sampling probabilities. This algorithm can be viewed as an extension of the method in [6]. However, there are some works on compressed least squares regression via random sampling. As far as we know, Drineas et al. [20] first applied the random sampling with column leverage scores or approximated ones as the sampling probabilities to the least squares regression and established the following estimator

$$\hat{\beta}_D = A^T(AS)^{\dagger T}(AS)^{\dagger} y,$$

which can be regarded as a special case of (1.8). Later, Slawski [18] investigated (1.9) using uniform sampling, and discussed the predictive performance.

In this paper, we will consider the application of random sampling on high-dimensional ridge regression further. Inspired by the technique of the optimal subsampling used in e.g., [21–29], we will mainly investigate the optimal subsampling probabilities for compressed ridge regression. The nearly optimal subsampling probabilities and a two step iterative algorithm are also derived. The optimal subsampling is a very active field in recent years, which was first proposed for least squares regression [21] and then for logistic regression [22]. Later, some scholars applied the technique to softmax regression [25], generalized linear models [26,27], quantile regression [28], ridge regression [29], and so on. These works all focus on large sample problems, i.e., $n \gg p$, while we concern the high-dimensional setting, i.e., $p \gg n$.

The remainder of this paper is organized as follows. The basic framework of random sampling algorithm and the optimal subsampling probabilities are presented in Section 2. In Section 3, we propose the nearly optimal subsampling probabilities and a two step iterative algorithm. The detailed theoretical analyses of the proposed methods are also presented in Sections 2 and 3, respectively. In Section 4, we provide some numerical experiments to test our methods. The concluding remarks of the whole paper are summarized in Section 5. The proofs of all the main theorems are given in the appendix.

Before moving to the next section, we introduce some standard notations used in this paper.

For the matrix $A \in \mathbb{R}^{n\times p}$, A_i, A^j, $\|A\|_2$ and $\|A\|_F$ denote its i-th column, j-th row, spectral norm and Frobenius norm, respectively. Also, its thin SVD is given as $A = U\Sigma V^T$, where $U \in \mathbb{R}^{n\times\rho}$ and $V \in \mathbb{R}^{p\times\rho}$ have orthogonal columns, and $\Sigma \in \mathbb{R}^{\rho\times\rho}$ is a diagonal matrix with the diagonal entries, i.e., the singular values of A, satisfying $\sigma_1(A) \geq \sigma_2(A) \geq \cdots \geq \sigma_\rho(A) > 0$.

For the above matrix V, its row norms $\|V^i\|_2$ with $i = 1, \ldots, p$ are called the column leverage scores, which are similar to the famous statistical leverage scores defined as the diagonal entries of the hat matrix $A(A^TA)^{\dagger}A^T$.

For the matrix $X = V\Sigma_\lambda$, where Σ_λ is a diagonal matrix with the diagonal entries being $\sqrt{\frac{\sigma_j(A)^2}{\lambda+\sigma_j(A)^2}}$ $(j = 1, \ldots, \rho)$, its row norms $\|X^i\|_2$ with $i = 1, \ldots, p$ are called the ridge leverage scores. For further explanations on the above two definitions, see [8].

In addition, $O_p(1)$ denotes that a sequence of random variables are bounded in probability and $o_p(1)$ represents that the sequence converges to zero in probability. More details can refer to [30, Chap. 2]. In our case, we also use $O_{p|\mathcal{F}_n}$ to denote that a sequence of random variables are bounded in conditional probability given $\mathcal{F}_n$, where $\mathcal{F}_n = (A, y)$ denotes the full data matrix. Especially, for any matrix G whose entries are sequences of random variables, $G = O_p(1)$ means that all the entries of G are bounded in probability, $G = O_{p|\mathcal{F}_n}(1)$ represents that its entries are bounded in conditional probability given $\mathcal{F}_n$, and $G = o_p(1)$ symbolizes that its entries converge to zero in probability.

2. Optimal subsampling

In this section, we will present the basic framework of random sampling algorithm, propose the optimal subsampling probabilities, and obtain the corresponding error analysis.

2.1. Algorithm and optimal subsampling probabilities

Given a set of probabilities, i.e., the random sampling matrix S, our approximate estimator

$$\hat{\beta} = A^T(ASS^TA^T + \lambda I)^{-1} y \tag{2.1}$$

of the high-dimensional ridge regression (1.1) is the combination of the solution to the compressed dual problem,

$$\operatorname{argmin}_z \frac{1}{2\lambda}\|S^TA^Tz\|_2^2 + \frac{1}{2}\|z\|_2^2 - z^Ty, \tag{2.2}$$

i.e.,

$$\hat{z} = \lambda(ASS^TA^T + \lambda I)^{-1} y, \tag{2.3}$$

and (1.4). That is, we first solve the problem (2.2) and then get the approximate estimator through (1.4). Note that this approach is different from the one in [4] though the expressions of $\hat{z}$ in (2.3) and $\widetilde{z}_L$ in (1.6) are the same. In fact, the authors in [4] first solve the compressed ridge regression (1.7) in the dual space and then find the estimator of the compressed regression via (1.4). Finally, the approximate estimator of the original ridge regression is recovered by the random matrix S. The detailed process of our approach, i.e., the basic framework of random sampling algorithm, is listed in Algorithm 1.

Remark 2.1. In Algorithm 1, the parameter λ can be determined by K-fold cross-validation, leave-one-out cross-validation, or generalized cross-validation, see e.g., [29]. Since the main focus of this paper is the performance of subsampling on high-dimensional ridge regression, we omit the investigation of the choice of λ.

Now, we investigate the sampling probabilities $\{\pi_i\}_{i=1}^p$ in Algorithm 1, which play a critical role on the performance of the algorithm. Below are some well known probabilities discussed in the literature.

- **Uniform sampling (UNI)**: $\pi_i^{UNI} = \frac{1}{p}$.

Algorithm 1 Random Sampling Algorithm for High-dimensional Ridge Regression (RSHRR)

Input: $y \in \mathbb{R}^n$, $A \in \mathbb{R}^{n\times p}$, the regularized parameter λ, the sampling size r with $r \ll p$, and the sampling probabilities $\{\pi_i\}_{i=1}^p$ with $\pi_i \geq 0$ such that $\sum_{i=1}^p \pi_i = 1$.
Output: the dual solution $\hat{z}$ and the primal solution $\hat{\beta}$.

1. initialize $S \in \mathbb{R}^{p\times r}$ to an all-zeros matrix.
2. for $t \in 1, \cdots, r$ do
   - pick $i_t \in [p]$ such that $\Pr(i_t = i) = \pi_i$.
   - set $S_{i_t,t} = \frac{1}{\sqrt{r\pi_{i_t}}}$.
3. end
4. calculate $\hat{z}$ as in (2.3).
5. return $\hat{\beta} = \frac{A^T\hat{z}}{\lambda}$.

- **Column sampling (COL)**: $\pi_i^{COL} = \frac{\|A_i\|_2^2}{\sum_{i=1}^p \|A_i\|_2^2}$.
- **Leverage sampling (LEV)** [8]: $\pi_i^{LEV} = \frac{\|V^i\|_2^2}{\sum_{i=1}^p \|V^i\|_2^2}$.
- **Ridge leverage sampling (RLEV)**[8]: $\pi_i^{RLEV} = \frac{\|X^i\|_2^2}{\sum_{i=1}^p \|X^i\|_2^2}$.

In the following, we discuss a new set of sampling probabilities, i.e., the optimal subsampling probabilities, which can be derived by combining the asymptotic variance of the estimators from Algorithm 1 and the A-optimal design criterion [31]. Considering the property of trace [32, Section 7.7] and the variance $\mathrm{Var}(\hat{\beta} - \hat{\beta}_{rls}|\mathcal{F}_n) = \frac{1}{\lambda^2}A^T\mathrm{Var}(\hat{z} - \hat{z}^*|\mathcal{F}_n)A$, to let the trace $\mathrm{tr}(\mathrm{Var}(\hat{\beta}-\hat{\beta}_{rls}|\mathcal{F}_n))$ attain its minimum, it suffices to make $\mathrm{tr}(\mathrm{Var}(\hat{z}-\hat{z}^*|\mathcal{F}_n))$ get its minimum. Thus, we mainly investigate the asymptotic variance of the dual estimator $\hat{z}$. As done in e.g., [21, 22,24–28], several conditions are first presented in Condition 2.1. They essentially describe the information on moment conditions and are mainly used to derive two auxiliary lemmas, i.e., Lemma A.1 and Lemma A.2, which are in turn indispensable for the proof of Theorem 2.1.

Condition 2.1. *For the design matrix $A \in \mathbb{R}^{n\times p}$, we assume that*

$$\sum_{i=1}^p \frac{\|A_i\|_2^6}{\pi_i^2 p^3} = O_p(1), \tag{2.4}$$

$$\sum_{i=1}^p \frac{A_iA_i^T\|A_i\|_2^2}{p^2\pi_i} = O_p(1), \tag{2.5}$$

$$\sum_{i=1}^p \frac{\|A_i\|_2^2}{p} = O_p(1), \tag{2.6}$$

$$\sum_{i=1}^p \frac{A_iA_i^T}{p} = O_p(1), \tag{2.7}$$

where π_i with $i = 1, \ldots, p$ are the given probabilities.

Remark 2.2. With respect to uniform sampling, i.e., $\pi_i = p^{-1}$, the conditions (2.4) and (2.5) are equivalent to

$$\sum_{i=1}^p \frac{\|A_i\|_2^6}{p} = O_p(1), \quad \sum_{i=1}^p \frac{A_iA_i^T\|A_i\|_2^2}{p} = O_p(1). \tag{2.8}$$

In this case, to make (2.8) hold, it is sufficient to suppose that $\mathrm{E}(\|A_i\|_2^6) < \infty$. Furthermore, the conditions (2.6) and (2.7) hold if $\mathrm{E}(\|A_i\|_2^2) < \infty$.

Remark 2.3. The above moment type conditions are wild. For example, if the entries of A obey the sub-Gaussian distribution [33], then all the conditions mentioned above are satisfied. The reason is that the sub-Gaussian distribution owns finite moments up to any finite order.

With the above conditions, we can present the following asymptotic distribution theorem.

Theorem 2.1. *Assume that the conditions (2.4), (2.5), (2.6), and (2.7) are satisfied. Then, as $p \to \infty$, $r \to \infty$, conditional on $\mathcal{F}_n$ in probability, the estimator $\hat{z}$ constructed by Algorithm 1 satisfies*

$$V^{-1/2}(\hat{z} - \hat{z}^*) \xrightarrow{L} N(0, I), \tag{2.9}$$

where the notation $\xrightarrow{L}$ represents the convergence in distribution, and

$$V = (\frac{M_A}{p})^{-1}\frac{V_c}{r}(\frac{M_A}{p})^{-1}$$

with $M_A = AA^T + \lambda I$ and $V_c = \sum_{i=1}^p \frac{A_iA_i^T\hat{z}^*\hat{z}^{*T}A_iA_i^T}{p^2\pi_i}$.

Following the A-optimal design criterion and the asymptotic variance V in (2.9), we can provide the optimal subsampling probabilities for Algorithm 1 by minimizing the trace $\mathrm{tr}(V)$. Noting that M_A does not depend on π_i and is nonnegative definite, we get that $V_c(\pi_1) \preccurlyeq V_c(\pi_2)$ is equivalent to $V(\pi_1) \preccurlyeq V(\pi_2)$ for any two sampling probability sets $\pi_1 = \{\pi_i^{(1)}\}_{i=1}^p$ and $\pi_2 = \{\pi_i^{(2)}\}_{i=1}^p$. Thus, we can simplify the optimal criterion by avoiding computing M_A^{-1}, namely, we can calculate the optimal subsampling probabilities by minimizing $\mathrm{tr}(V_c)$ instead of $\mathrm{tr}(V)$. Actually, this can be viewed as the L-optimal design criterion [31] with $L = rp^{-2}M_A^2$.

Theorem 2.2. *For Algorithm 1, when*

$$\pi_i^{OPL} = \frac{|\hat{\beta}_{rls(i)}|\,\|A_i\|_2}{\sum_{i=1}^p |\hat{\beta}_{rls(i)}|\,\|A_i\|_2}, \quad i = 1, \ldots, p, \tag{2.10}$$

where $\hat{\beta}_{rls(i)}$ is the ith entry of the ridge estimator $\hat{\beta}_{rls}$, $\mathrm{tr}(V_c)$ achieves its minimum.

Remark 2.4. When $\lambda \to 0^+$, (2.10) can be degraded to the optimal subsampling probabilities of the compressed least squares regression.

Remark 2.5. Note that $V_c = \lambda^2\sum_{i=1}^p \frac{\hat{\beta}_{rls}^2A_iA_i^T}{p^2\pi_i}$. Thus, by

$$|\hat{\beta}_{rls}(i)| = \|A_i^T(AA^T + \lambda I)^{-1}y\|_2 \leq \|A_i\|_2\|(AA^T + \lambda I)^{-1}y\|_2,$$

we have

$$\begin{aligned}\mathrm{tr}(V_c) &\leq \frac{\lambda^2\|(AA^T + \lambda I)^{-1}y\|_2^2}{p^2}\sum_{i=1}^p \frac{\|A_i\|_2^4}{\pi_i} \\ &= \frac{\lambda^2\|(AA^T + \lambda I)^{-1}y\|_2^2}{p^2}\sum_{i=1}^p \pi_i\sum_{i=1}^p \frac{\|A_i\|_2^4}{\pi_i}.\end{aligned}$$

Further, by Cauchy–Schwarz inequality, we obtain

$$\frac{\lambda^2\|(AA^T + \lambda I)^{-1}y\|_2^2}{p^2}\sum_{i=1}^p \pi_i\sum_{i=1}^p \frac{\|A_i\|_2^4}{\pi_i} \geq \frac{\lambda^2\|(AA^T + \lambda I)^{-1}y\|_2}{p^2}(\sum_{i=1}^p \|A_i\|_2^2)^2.$$

Thus, analogous to Theorem 2.2, we get that when

$$\pi_i = \pi_i^{COL} = \frac{\|A_i\|_2^2}{\sum_{i=1}^p \|A_i\|_2^2}, \tag{2.11}$$

the upper bound of $\mathrm{tr}(V_c)$, i.e., $\frac{\lambda^2\|(AA^T+\lambda I)^{-1}y\|_2^2}{p^2}\sum_{i=1}^p \frac{\|A_i\|_2^4}{\pi_i}$, reaches the minimum. Obviously, (2.11) is easier to compute compared with (2.10). However, we has to lose some accuracy as expense in this case.

Similarly, based on $\|A_i\|_2^2 \leq \|A\|_F^2$, we have

$$\mathrm{tr}(V_c) \leq \frac{\lambda^2\|A\|_F^2}{p^2}\sum_{i=1}^p \frac{\hat{\beta}_{rls(i)}^2}{\pi_i} = \frac{\lambda^2\|A\|_F^2}{p^2}\sum_{i=1}^p \pi_i\sum_{i=1}^p \frac{\hat{\beta}_{rls(i)}^2}{\pi_i}$$

and

$$\frac{\lambda^2\|A\|_F^2}{p^2}\sum_{i=1}^p \pi_i\sum_{i=1}^p \frac{\hat{\beta}_{rls(i)}^2}{\pi_i} \geq \frac{\lambda^2\|A\|_F^2}{p^2}(\sum_{i=1}^p |\hat{\beta}_{rls(i)}|)^2.$$

Then, we find that when

$$\pi_i = \pi_i^{RSIS} = \frac{|\hat{\beta}_{rls(i)}|}{\sum_{i=1}^{p}|\hat{\beta}_{rls(i)}|},$$

the above upper bound of $\mathrm{tr}(V_c)$ reaches the minimum. Surprisingly, π_i^{RSIS} corresponds to the screening criteria of iteratively thresholded ridge regression screener given in [34]. This fact implies that the screener with the probabilities in (2.10) may perform better than the one in [34].

2.2. Error analysis for RSHRR

We first give an estimation error bound.

Theorem 2.3. *Assume that*

$$c_1\|V^i\|_2 \le \|A_i\|_2 \le c_2\|V^i\|_2 \text{ and}$$
$$s_1\|V^i\|_2\|y\|_2 \le |\hat{\beta}_{rls(i)}| \le s_2\|V^i\|_2\|y\|_2, \quad i = 1,\dots,p, \tag{2.12}$$

where $0 < c_1 \le c_2$ *and* $0 < s_1 \le s_2$, *and let* $r \ge \frac{32s_2c_2\rho}{3s_1c_1\epsilon^2}\ln(\frac{4\rho}{\delta})$ *with* $\epsilon,\delta \in (0,1)$. *Then, for* S *formed by* $\pi_i = \pi_i^{OPL}$ *and any* ϵ, *with the probability at least* $1-\delta$, $\hat{\beta}$ *constructed by Algorithm 1 satisfies*

$$\|\hat{\beta} - \hat{\beta}_{rls}\|_2 \le \epsilon\|\hat{\beta}_{rls}\|_2, \tag{2.13}$$

where $\hat{\beta}_{rls}$ *is as in* (1.5).

Remark 2.6. The assumptions in (2.12) are reasonable and reachable due to $A_i = U\Sigma(V^i)^T$ and

$$\hat{\beta}_{rls(i)} = A_i^T(AA^T+\lambda I)^{-1}y = V^i(\Sigma+\lambda\Sigma^{-1})^{-1}U^Ty.$$

In fact, for the worst case, $c_1 = \sigma_\rho(A)$, $c_2 = \sigma_1(A)$, and s_1 and s_2 are controlled by $\min_{j=1,\dots,\rho}\{\frac{\sigma_j(A)}{\sigma_j^2(A)+\lambda}\}$ and $\max_{j=1,\dots,\rho}\{\frac{\sigma_j(A)}{\sigma_j^2(A)+\lambda}\}$, respectively. The aim for introducing the parameters c_1, c_2, s_1, and s_2 here is to simplify the expression of r.

In the following, we provide a risk bound, in which the risk function is defined as

$$\mathrm{risk}(\hat{y}) = \frac{1}{n}\mathrm{E}_y(\|\hat{y} - A\beta\|_2^2),$$

where E_y denotes the expectation on y, and $\hat{y}$ denotes the prediction of $A\beta$.

Theorem 2.4. *Suppose that the setting is the same as the one in Theorem 2.3, and let* $\mu = \sqrt{\sum_{j=1}^{\rho}\frac{\sigma_j^2(A)}{(\sigma_j^2(A)+\lambda)^2}}$. *Then, for* S *formed by* $\pi_i = \pi_i^{OPL}$ *and any* ϵ, *with probability at least* $1-\delta$,

$$\mathrm{risk}(\hat{y}) \le \mathrm{risk}(y_*) + \frac{3\epsilon}{n}\|A\|_2^2(\mu^2 + \|\beta\|_2^2),$$

where $\hat{y} = A\hat{\beta}$ *with* $\hat{\beta}$ *constructed by Algorithm 1 and* $y_* = A\hat{\beta}_{rls}$.

3. Two step iterative algorithm

Considering that the sampling probabilities (2.10) are uneconomic since they are required to figure out $\hat{\beta}_{rls}$, we now present the approximate ones. Specifically, we first apply Algorithm 1 with $\pi_i = \pi_i^{COL}$ and the sampling size being r_0 to return an approximation $\widetilde{\beta}$ of $\hat{\beta}_{rls}$. Then, a set of probabilities $\{\pi_i^{NOPL}\}_{i=1}^p$ are obtained by replacing $\hat{\beta}_{rls(i)}$ in (2.10) with $\widetilde{\beta}_{(i)}$, i.e.,

$$\pi_i^{NOPL} = \frac{|\widetilde{\beta}_{(i)}|\,\|A_i\|_2}{\sum_{i=1}^{p}|\widetilde{\beta}_{(i)}|\,\|A_i\|_2}, \quad i = 1,\dots,p. \tag{3.1}$$

We call them the nearly optimal subsampling probabilities. Moreover, to further reduce the estimation error, we bring in the iterative method. The key motivation is that if $\|\hat{\beta}_t - \hat{\beta}_{rls}\|_2 \le \epsilon\|\hat{\beta}_{t-1} - \hat{\beta}_{rls}\|_2$ holds at the tth iteration, then a solution owning the estimation error bound $\epsilon^m\|\hat{\beta}_0 - \hat{\beta}_{rls}\|_2$ will be returned when the approximation process is repeated m times. Putting the above discussions together, we propose a two step iterative algorithm, i.e., Algorithm 2.

Algorithm 2 Two Step Iterative Algorithm for High-dimensional Ridge Regression

Input: $y \in \mathbb{R}^n$, $A \in \mathbb{R}^{n\times p}$, the regularized parameter λ, the iterative number m, the sampling size r and r_0, where $r_0 \ll r \ll p$.
Output: the dual estimator $\hat{z}_m$ and the recovered solution $\hat{\beta}_m$.
Step 1:

1. initialize $S^* \in \mathbb{R}^{p\times r_0}$ to an all-zeros matrix.
2. for $i \in 1,\cdots,p$ do
   - $\pi_i^{COL} = \frac{\|A_i\|_2^2}{\sum_{i=1}^p\|A_i\|_2^2}$.
3. end
4. for $t \in 1,\cdots,r_0$ do
   - pick $i_t \in [p]$ such that $\Pr(i_t = i) = \pi_i$.
   - $S^*_{i_t,t} = \frac{1}{\sqrt{r_0\pi_{i_t}}}$.
5. end
6. compute $A^* = AS^*$.
7. compute $C = (A^*A^{*T} + \lambda I)^{-1}$.

Step 2:

1. set $\hat{z}_0 = 0$.
2. for $t \in 1,\cdots,m$ do
   - $\hat{\beta}_{t-1} = \frac{1}{\lambda}A^T\hat{z}_{t-1}$.
   - $b_t = y - A\hat{\beta}_{t-1} - \hat{z}_{t-1}$.
   - $\widetilde{z} = \lambda Cb_t$.
   - $\widetilde{\beta} = \frac{A^T\widetilde{z}}{\lambda}$.
   - compute π_i^{NOPL} by (3.1).
   - compute $\hat{w}_t$ by applying Algorithm 1 with $y = b_t$ and $\pi_i = \pi_i^{NOPL}$.
   - $\hat{z}_t = \hat{z}_{t-1} + \hat{w}_t$.
3. end
4. return $\hat{z}_m$ and $\hat{\beta}_m = \frac{A^T\hat{z}_m}{\lambda}$.

Remark 3.1. The step 2 of Algorithm 2 can be viewed as a variant of iterative Hessian sketch (IHS) [7]. This is because, at the tth iteration, applying Algorithm 1 for finding $\hat{w}_t$ is equivalent to applying Hessian sketch to the residual between z and $\hat{z}_{t-1}$. That is, at the tth iteration, we need to solve the following problem

$$\min_{w_t}\frac{1}{2\lambda}\|S^TA^Tw_t\|_2^2 + \frac{1}{2}\|w_t\|_2^2 - w_t^Tb_t,$$

where $w_t = z - z_{t-1}$ and S is constructed by π_i^{NOPL}.

In addition, the step 2 of Algorithm 2 is also similar to Algorithm 1 in [8]. However, the key ideas of the two methods are different. As mentioned above, the former is essentially the IHS, which is in turn the specialization of Newton sketch in least squares problem [35]. While, the latter can be regraded as the preconditioned Richardson iteration [36, Chap. 2] for solving $(AA^T+\lambda I)z = \lambda y$ with pre-conditioner $P^{-1} = (ASS^TA^T + \lambda I)^{-1}$ and the step-size being one. Moreover, its random sampling matrix S is fixed during the iteration.

Remark 3.2. The computational cost of Algorithm 2 includes two main parts. The first one is for computing $\widetilde{\beta}$ and hence π_i^{NOPL}, which mainly appears in the step 1 of Algorithm 2 and costs $O(np + n^2r_0)$, and the other one is $O(mn^2r + mnp)$ consumed for deriving $\hat{\beta}_m$, which constitutes the step 2 of Algorithm 2. As a result, the total cost is $O(n^2r_0 + mn^2r + mnp)$.

By contrast, it suffices to run the step 2 in Algorithm 2 if π_i^{OPL}, π_i^{LEV}, π_i^{RLEV}, π_i^{UNI}, or π_i^{COL} is used to generate S. In addition, when π_i^{OPL} is employed, C in Algorithm 2 should be $(AA^T+\lambda I)^{-1}$, and when π_i^{LEV}, π_i^{RLEV}, π_i^{UNI}, or π_i^{COL} is adopted, the lines 5–7 of the step 2 of Algorithm 2 can be omitted. Consequently, taking the above first three probabilities for obtaining $\hat{\beta}_m$ costs $O(n^2p+mn^2r+mnp)$,[1] and applying π_i^{UNI} and π_i^{COL} for $\hat{\beta}_m$ spends $O(mn^2r+mnp)$.

Therefore, if r_0 is much smaller than p, Algorithm 2 with π_i^{NOPL} will be much cheaper than the algorithm with π_i^{OPL}, π_i^{LEV}, or π_i^{RLEV}. Otherwise, these algorithms will have similar cost. As expected, Algorithm 2 with π_i^{UNI} or π_i^{COL} is always quite cheap.

Next, we show that the difference of $\hat{z}_*$ and $\hat{z}_1$ still obeys asymptotically normal distribution, where $\hat{z}_1$ is returned from Algorithm 2 with $m=1$.

Theorem 3.1. *Suppose that the conditions (2.6) and (2.7) hold, and let*

$$N_1\|A_i\|_2\|y\|_2 \le|\widetilde{\beta}_{(i)}|\le N_2\|A_i\|_2\|y\|_2 \text{ and}$$
$$N_3\|A_i\|_2\|y\|_2 \le|\hat{\beta}_{rls(i)}|\le N_4\|A_i\|_2\|y\|_2, \quad i=1,\dots,p, \tag{3.2}$$

where $\widetilde{\beta}_{(i)}$ is as in Algorithm 2, $0<N_1\le N_2$, and $0<N_3\le N_4$. Then, as $p\to\infty$, $r\to\infty$, $r_0\to\infty$, conditional on $\mathcal{F}_n$ and $\widetilde{\beta}$ in probability, the dual estimator $\hat{z}_1$ constructed by Algorithm 2 satisfies

$$V_{OPL}^{-1/2}(\hat{z}_1-\hat{z}^*)\xrightarrow{L} N(0,I), \tag{3.3}$$

where

$$V_{OPL}=(\frac{M_A}{p})^{-1}\frac{V_{cOPL}}{r}(\frac{M_A}{p})^{-1}$$

with

$$V_{cOPL}=\sum_{i=1}^{p}\frac{A_iA_i^T\hat{z}^*\hat{z}^{*T}A_iA_i^T}{p^2\pi_i^{OPL}}=\sum_{i=1}^{p}|\hat{\beta}_{rls(i)}|\,\|A_i\|_2\sum_{i=1}^{p}\frac{A_iA_i^T\hat{z}^*\hat{z}^{*T}A_iA_i^T}{p^2|\hat{\beta}_{rls(i)}|\,\|A_i\|_2}.$$

Now, we provide an estimation error bound of our algorithm.

Theorem 3.2. *To the assumptions of Theorem 2.3, add that*

$$s_3\|V^i\|_2\|y\|_2\le|\widetilde{\beta}_{(i)}|\le s_4\|V^i\|_2\|y\|_2, \quad i=1,\dots,p, \tag{3.4}$$

where $\widetilde{\beta}_{(i)}$ is as in Algorithm 2 and $0<s_3\le s_4$, the initial value $\hat{z}_0$ is set as 0, and let $r\ge\frac{32s_4c_2\rho}{3s_3c_1\epsilon^2}\ln(\frac{4p}{\delta})$ with $\epsilon,\delta\in(0,1)$ and $m<\frac{1}{\delta}$. Then, for $\widetilde{S}$ constructed by π_i^{NOPL} and any ϵ, with the probability at least $1-m\delta$, $\hat{\beta}_m$ generated from Algorithm 2 satisfies

$$\|\hat{\beta}_m-\hat{\beta}_{rls}\|_2\le\epsilon^m\|\hat{\beta}_{rls}\|_2. \tag{3.5}$$

Remark 3.3. The bound (3.5) can be used to determine the iteration number. Specifically, it is enough to do $\log_\epsilon\iota$ iterations to get $\|\hat{\beta}_m-\hat{\beta}_{rls}\|_2\le\iota\|\hat{\beta}_{rls}\|_2$.

4. Numerical experiments

In this section, we provide the numerical results of experiments with simulation data and real data. All experiments are implemented on a laptop running MATLAB software with 16 GB random-access memory (RAM).

[1] Note that the computational complexity only contains the main cost of algorithms. Hence, the algorithms may perform a little differently in computing time though they have the same complexity.

4.1. Simulation data—Example 1

In this example, the simulation data is generated as done in [18]. Specifically, we first produce an n-by-p matrix B randomly, whose entries are drawn i.i.d. from the $N(0,1)$ distribution and SVD is denoted as $U_B\Sigma_BV_B^T$ with $U_B\in\mathbb{R}^{n\times n}$, $\Sigma_B\in\mathbb{R}^{n\times n}$ and $V_B\in\mathbb{R}^{p\times n}$. Then, we get A by replacing Σ_B with Σ_0, i.e, $A=U_B\Sigma_0V_B^T$, where Σ_0 is a diagonal matrix with polynomial decay diagonal entries $\sigma_j(j=1,\dots,n)$, namely, $\sigma_j\propto 9\times j^{-8}$. Furthermore, we construct the response vector y by $y=A\beta+\varsigma$, where $\beta\in\mathbb{R}^p$ and $\varsigma\in\mathbb{R}^n$ have i.i.d. $N(0,1)$ entries.

In the specific experiments, we set $n=800$ and $p=20000$, and split the data into 500 training part A and 300 testing part A_{test}. The description on parameters of the experiments is summarized in Table 1, the explanation on six sampling methods is given in Table 2, and the numerical results on accuracy, i.e., the estimation error $\frac{\|\hat{\beta}_m-\hat{\beta}_{rls}\|_2}{\|\hat{\beta}_{rls}\|_2}$ and the prediction error $\frac{\|A_{test}\hat{\beta}_m-A_{test}\hat{\beta}_{rls}\|_2}{\|A_{test}\hat{\beta}_{rls}\|_2}$, and CPU time[2] are shown in Tables 3–4 and Figs. 1–4. Note that all the error results shown in figures are on log-scale and all the numerical results are based on 50 replications of Algorithm 2.

In the first experiment, we aim to show that the estimators established by OPL and NOPL have better performance. The corresponding numerical results are presented in Tables 3–4 and Figs. 1–3. From these tables and figures, it is obvious to find that OPL and NOPL outperform other methods on estimation and prediction accuracy no matter what r and λ are and when m is moderate, and the differences can be more than nine orders of magnitude; see e.g., the case on $m=3$ in Tables 3–4. But they need more computing time than COL and UNI. However, the improvement in accuracy is more than the sacrifice of calculation cost, and fortunately, OPL and NOPL are cheaper than LEV and RLEV. What is more, we can observe that NOPL has extremely similar accuracy to OPL, and the former consumes less running time. In addition, in most cases, the errors of all the methods decrease when r, λ and m increase. Whereas, when m is large enough, they are almost unchanged but very small, and all the methods perform similarly. This is mainly because iteration can reduce error and, in that case, the sampling ways have negligible effect on accuracy.

For the second experiment, we compare the methods OPL and NOPL with different r_0. According to the numerical results displayed in Fig. 4, it is evident to conclude that for different r_0, NOPL is able to achieve significantly similar accuracy to OPL but spends less computational cost. The latter confirms the complexity analysis in Remark 3.2.

4.2. Simulation data—Example 2

For this example, we produce the simulation data as done in [8]. Specifically, we construct an n-by-p design matrix $A=PDQ^T+\alpha M$, where $P\in\mathbb{R}^{n\times n}$ is a random matrix with i.i.d. $N(0,1)$ entries, $D\in\mathbb{R}^{n\times n}$ is a diagonal matrix with diagonal entries $D_{ii}=(1-\frac{i-1}{p})^i(i=1,\dots,n)$, $Q\in\mathbb{R}^{p\times n}$ is a random column orthonormal matrix, $M\in\mathbb{R}^{n\times p}$ is a noise matrix with i.i.d. $N(0,1)$ entries, and $\alpha>0$ is a parameter used to balance PDQ^T and M. In addition, the response vector $y\in\mathbb{R}^n$ is generated according to $y=A\beta+\gamma\varsigma$, where $\beta\in\mathbb{R}^p$ and $\varsigma\in\mathbb{R}^p$ are constructed by i.i.d. $N(0,1)$ entries. In the specific experiments, we set $n=800$, $p=20000$, $\alpha=0.0001$ and $\gamma=0.5$, divide the data into 500 training part and 300 testing part, and repeat the implementations in Section 4.1 with different r, r_0, λ and m shown in Table 5.

From the numerical results presented in Tables 6–7 and Figs. 5–8, we can gain the similar observations to the ones in Section 4.1. That is, taking different r and λ, and suitable m, OPL and NOPL always perform better than other methods on accuracy, however, need more CPU time compared with COL and UNI. And, OPL and NOPL still show better computational efficiency than LEV and RLEV. Besides, when setting a

[2] To ensure fairness, the CPU time includes the time of computing $\hat{\beta}_{rls}$.

Table 1
Description of two experiments for example 1.

Kinds	Comparison	r	λ	m	r_0	Results
1	six methods	500 to 5000	10	3	100 (NOPL)	Figs. 1–3(a)
		1000	1 to 50	3	100 (NOPL)	Figs. 1–3(b)
		1000	10	1 to 15	100 (NOPL)	Tables 1–3 & Fig. 3(c)
2	OPL and NOPL	2000	10	3	100 to 2000 (NOPL)	Fig. 4

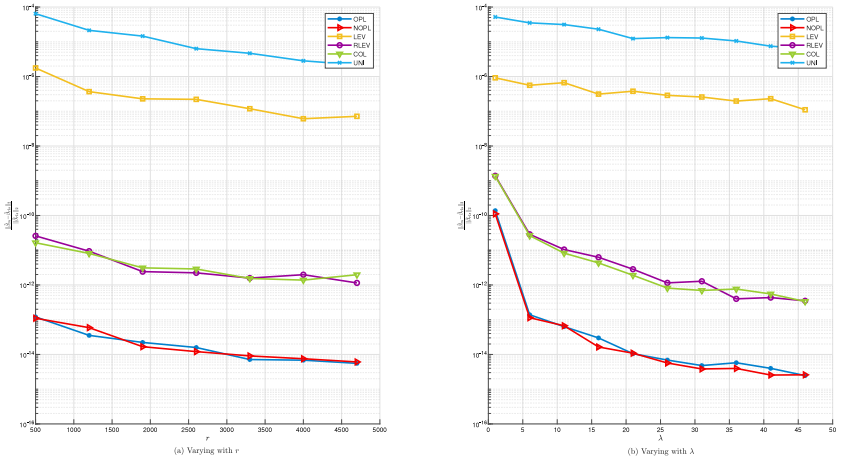

Fig. 1. Comparison of estimation errors using different methods for example 1.

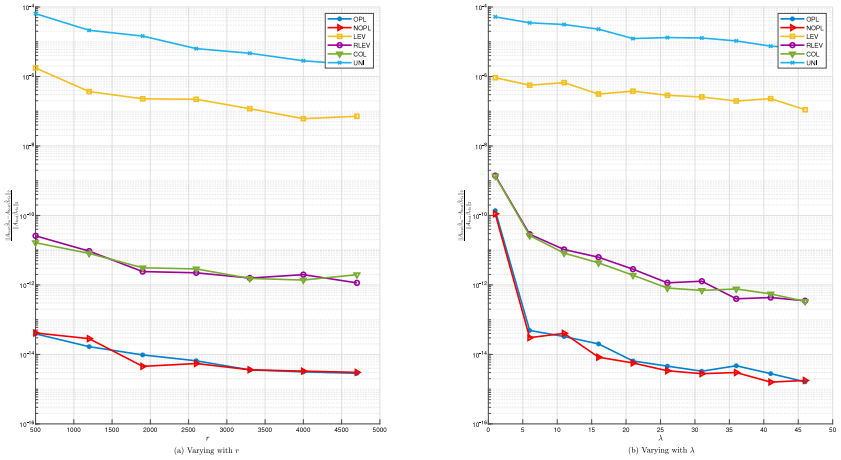

Fig. 2. Comparison of prediction errors using different methods for example 1.

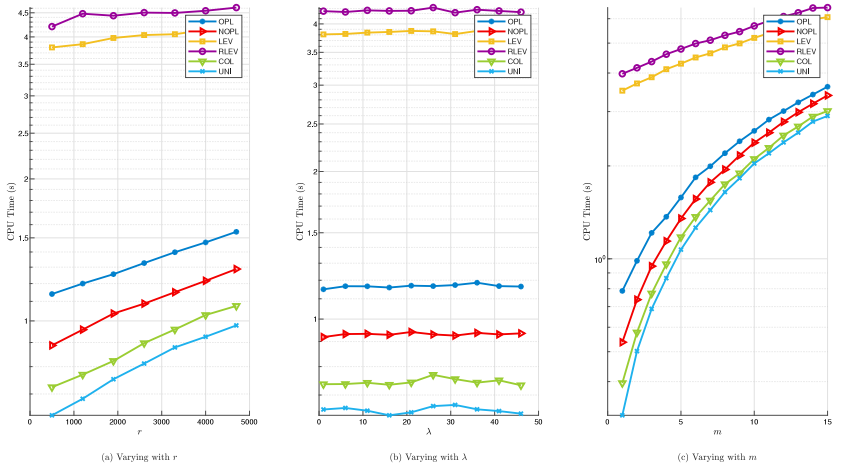

Fig. 3. Comparison of CPU time using different methods for example 1.

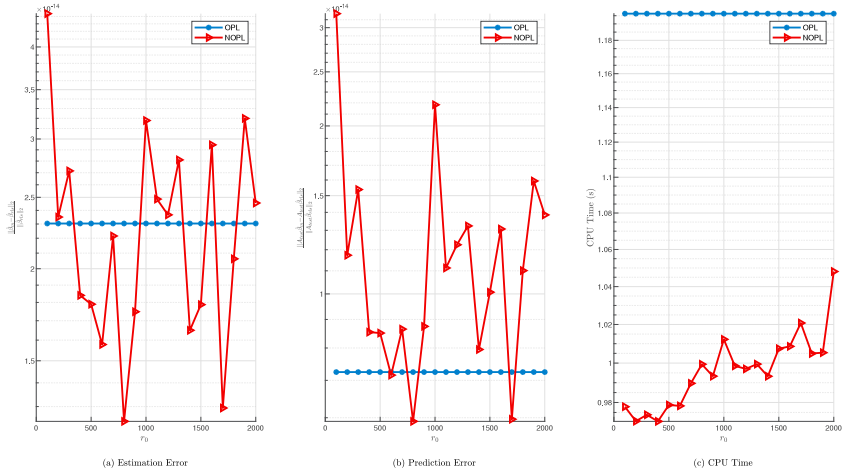

Fig. 4. Comparison of OPL and NOPL with different r_0 for example 1.

Table 2
Explanation of sampling methods with different probabilities.

Method	π_i	Expression
OPL	π_i^{OPL}	$\lvert\hat{\beta}_{rls(i)}\rvert \lVert A_i\rVert_2 / \sum_{i=1}^{p} \lvert\hat{\beta}_{rls(i)}\rvert \lVert A_i\rVert_2$
NOPL	π_i^{NOPL}	$\lvert\widetilde{\beta}_{(i)}\rvert \lVert A_i\rVert_2 / \sum_{i=1}^{p} \lvert\widetilde{\beta}_{(i)}\rvert \lVert A_i\rVert_2$
LEV	π_i^{LEV}	$\lVert V^i\rVert_2^2 / \sum_{i=1}^{p} \lVert V^i\rVert_2^2$
RLEV	π_i^{RLEV}	$\lVert X^i\rVert_2^2 / \sum_{i=1}^{p} \lVert X^i\rVert_2^2$
COL	π_i^{COL}	$\lVert A_i\rVert_2^2 / \sum_{i=1}^{p} \lVert A_i\rVert_2^2$
UNI	π_i^{UNI}	$1/p$

Table 3
Comparison of estimation errors using different m for example 1.

Methods	m				
	1	2	3	4	10
OPL	2.0458e−06	1.8172e−08	6.5331e−14	1.5007e−15	1.1798e−15
NOPL	2.8882e−06	1.8012e−08	7.3978e−14	1.4667e−15	1.1803e−15
LEV	0.00756	7.745e−05	3.8597e−07	4.4529e−09	1.1796e−15
RLEV	0.00026	4.7143e−08	1.848e−11	1.6885e−15	1.1801e−15
COL	0.00027	3.7182e−08	1.4237e−11	1.9279e−15	1.1827e−15
UNI	0.02601	0.00118	3.7761e−05	6.8432e−07	2.0803e−15

Table 4
Comparison of prediction errors using different m for example 1.

Methods	m				
	1	2	3	4	10
OPL	1.0255e−06	1.8173e−08	3.2869e−14	8.034e−16	3.6206e−16
NOPL	1.374e−06	1.8014e−08	4.8969e−14	7.4065e−16	3.651e−16
LEV	0.00758	7.7457e−05	3.8601e−07	4.4533e−09	3.6472e−16
RLEV	0.00026	4.7143e−08	1.8481e−11	1.0908e−15	3.6414e−16
COL	0.00027	3.7183e−08	1.4237e−11	1.347e−15	3.7391e−16
UNI	0.02601	0.00118	3.7764e−05	6.8438e−07	1.3422e−15

proper r_0 or a large λ, NOPL and OPL have similar accuracy but the former needs less running time. Unfortunately, when r_0 is very large, NOPL loses its advantage in CPU time. This is consistent with the discussions on computational cost given in Remark 3.2. In addition, unlike the results on accuracy in Section 4.1, the corresponding improvement of OPL and NOPL over other methods is not very remarkable in this example, about one order of magnitude. This is mainly because the data here is more even than the one from Example 1 and the importance sampling is well-known to be more suitable for uneven data.

4.3. Real data—gene expression cancer RNA-seq data set

The data set is from the UCI machine learning repository, which can be found in http://archive.ics.uci.edu/ml/datasets/gene+expression+cancer+RNA-Seq. Here, we only take the first 700 samples (400 for training part and 300 for testing part) with 20531 real-valued features, and centralize the design matrix. The response vector consists of 1, 2, 3, 4 and 5 labels, which represent five different types of tumors, i.e., PRAD, LUAD, BRCA, KIRC and COAD. We also centralize it.

We repeat the experiments in Sections 4.1 and 4.2 with different r, r_0, λ and m. More details are put in Table 8.

The numerical results are displayed in Tables 9–10 and Figs. 9–12, and the conclusions summarized from these tables and figures are akin to the ones found in Sections 4.1 and 4.2. Namely, compared with UNI and COL, the accuracy of OPL and NOPL is dramatically improved at the cost of slightly computational efficiency, and OPL performs better than LEV and RLEV on accuracy and computing time. Although NOPL is only a little better than LEV and RLEV on accuracy, it owns great advantage of CPU time. When taking a proper r_0, NOPL can be a well approximation of OPL but consumes less computing time. However, when r_0 is very large, NOPL will lose its superiority in computational cost. In addition, for this real data, the choice of λ has little influence on accuracy.

4.4. Real data—gisette data set

This data set is also from the UCI machine learning repository, which can be found in http://archive.ics.uci.edu/ml/datasets/Gisette. In our experiments, the first 200 samples (100 for training part and 100 for testing part) with 5000 real-valued features are taken, and the response vector is made up with ± 1 labels. Also, we centralize the response vector and design matrix prior to analysis.

As done in Section 4.3, we can repeat the experiments in Sections 4.1 and 4.2 with different r, r_0, λ and m described in Table 11. Considering that the obtained observations are similar to the ones from Section 4.3, we omit the related numerical results here. Instead, as done in [5,8], we next consider an alternative accuracy metric, i.e., $\lvert\frac{g(\hat{\beta}_m)}{g(\hat{\beta}_{rls})} - 1\rvert$, where $g(x) = \lVert Ax - y\rVert_2^2 + \lambda\lVert x\rVert_2^2$.

The numerical results are shown in Tables 12–14. The observations are also similar to the ones from Sections 4.1 and 4.2. To be more specific, whatever the values of r and m are, for accuracy, OPL and NOPL almost always outperform other methods. Similarly, as for CPU time, OPL and NOPL are inferior to UNI and COL, but are superior to LEV and RLEV. Only when r_0 is not particularly large, NOPL has good performance on both accuracy and computing time, and qualifies as a well alternative to OPL. Besides, the change of λ also has little effect on accuracy and CPU time. While, to save space, the corresponding results are omitted.

Table 5
Description of two experiments for example 2.

Kinds	Comparison	r	λ	m	r_0	Results
1	six methods	3000 to 10000	20	15	2000 (NOPL)	Figs. 5–7(a)
		5000	1 to 200	15	2000 (NOPL)	Figs. 5–7(b)
		5000	20	1 to 30	2000 (NOPL)	Tables 6–7 & Fig. 7(c)
2	OPL and NOPL	5000	20	15	500 to 20000 (NOPL)	Fig. 8

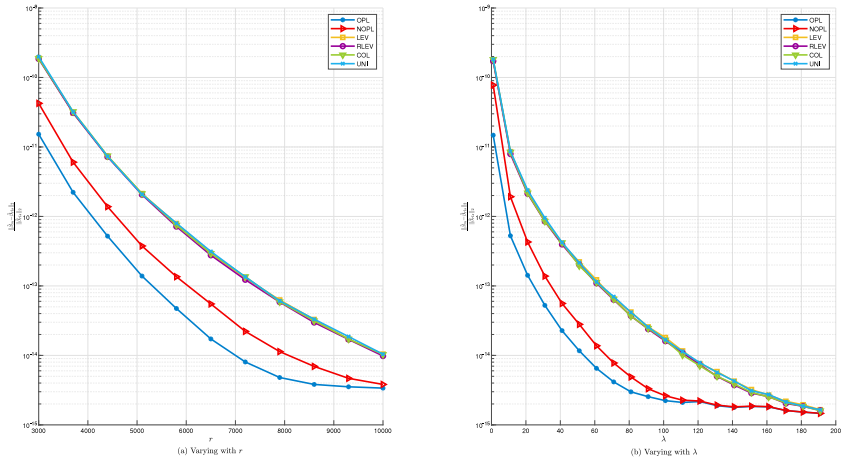

Fig. 5. Comparison of estimation errors using different methods for example 2.

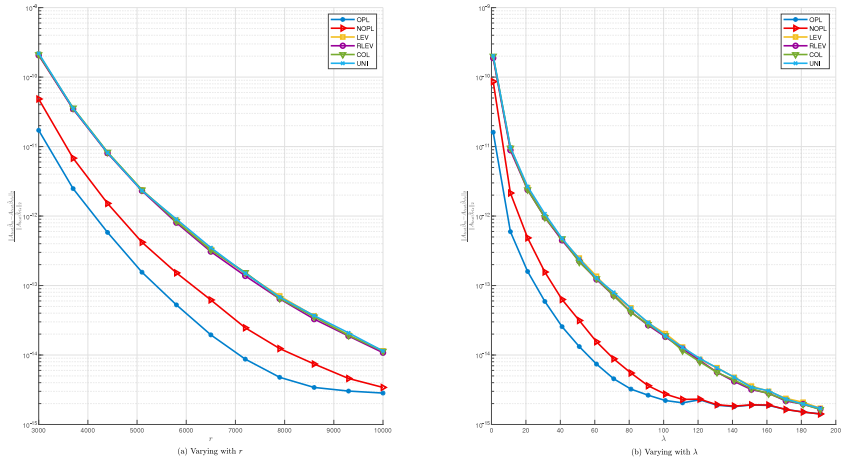

Fig. 6. Comparison of prediction errors using different methods for example 2.

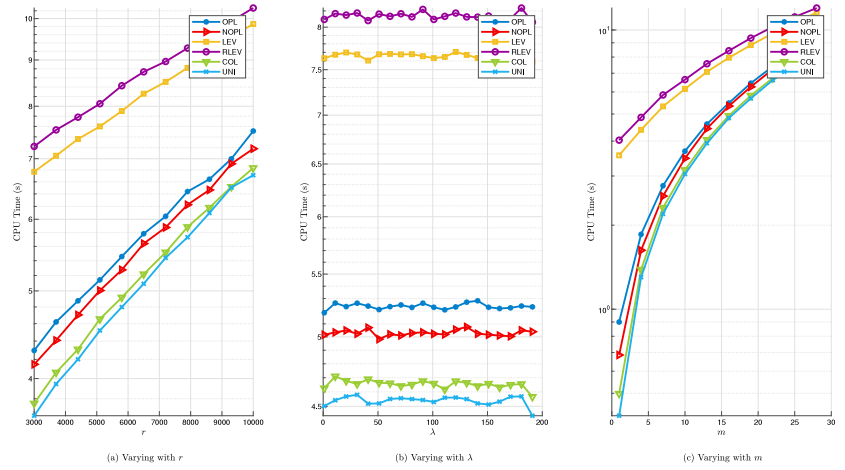

Fig. 7. Comparison of CPU time using different methods for example 2.

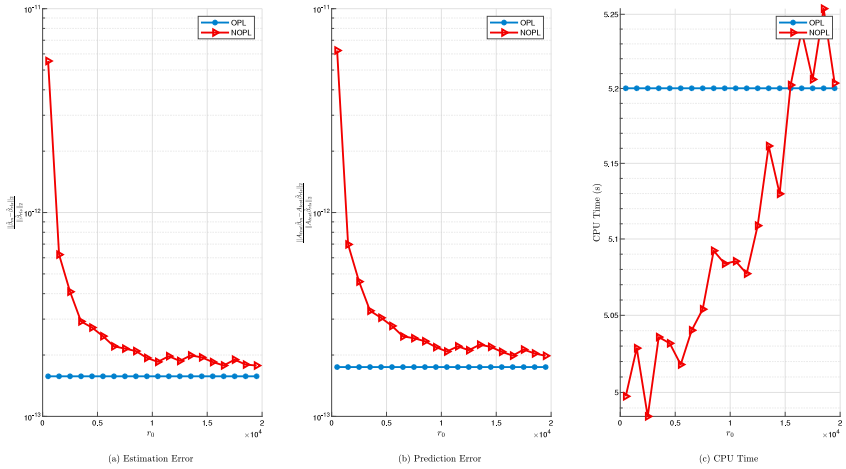

Fig. 8. Comparison of OPL and NOPL with different r_0 for example 2.

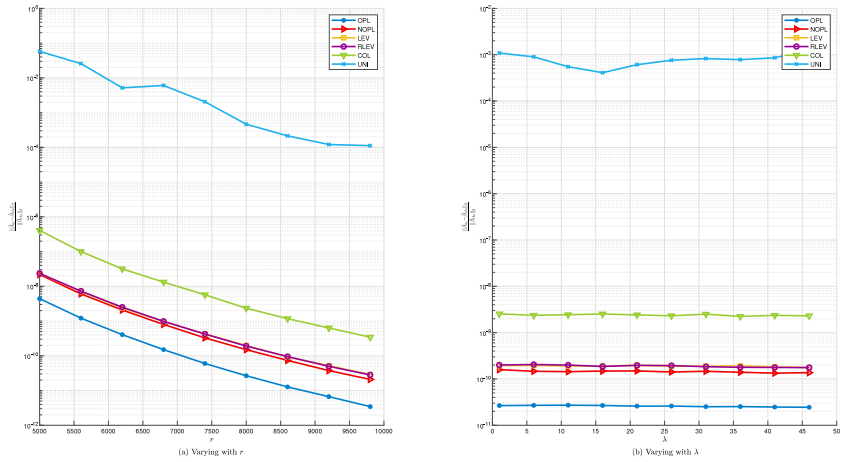

Fig. 9. Comparison of estimation errors for different methods for Gene Expression Cancer RNA-Seq data set.

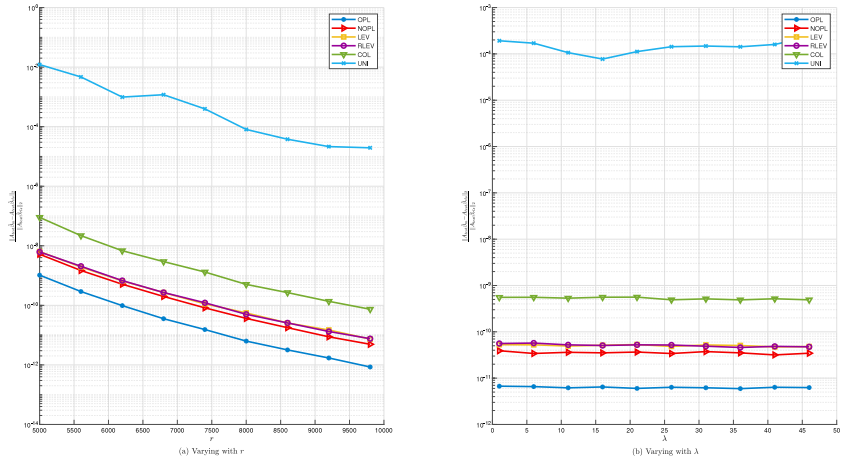

Fig. 10. Comparison of prediction errors for different methods for Gene Expression Cancer RNA-Seq data set.

5. Concluding remarks

In this paper, we explore the optimal subsampling probabilities of high-dimensional ridge regression under the A-optimal design criterion and provide a corresponding algorithm. To make the probabilities cheaper and more practical, we give a set of nearly optimal ones. Moreover, a two step iterative algorithm is also provided to further improve the accuracy of the estimator. For the proposed algorithms, we give detailed theoretical analysis and extensive experiments. Numerical results show that our methods, i.e., OPL and NOPL, outperform the existing methods on accuracy, and the cheaper NOPL can be a good substitute for OPL. An interesting future work is to study other high-dimensional and complex regression problems, e.g., the high-dimensional quantile regression.

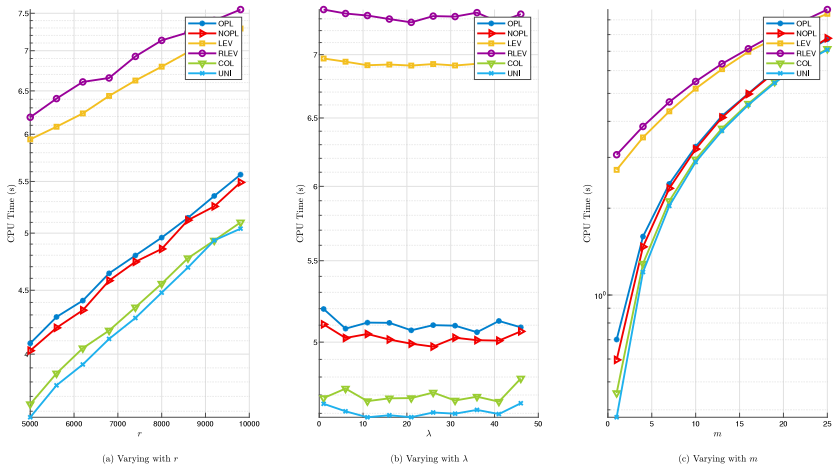

Fig. 11. Comparison of CPU time for different methods for Gene Expression Cancer RNA-Seq data set.

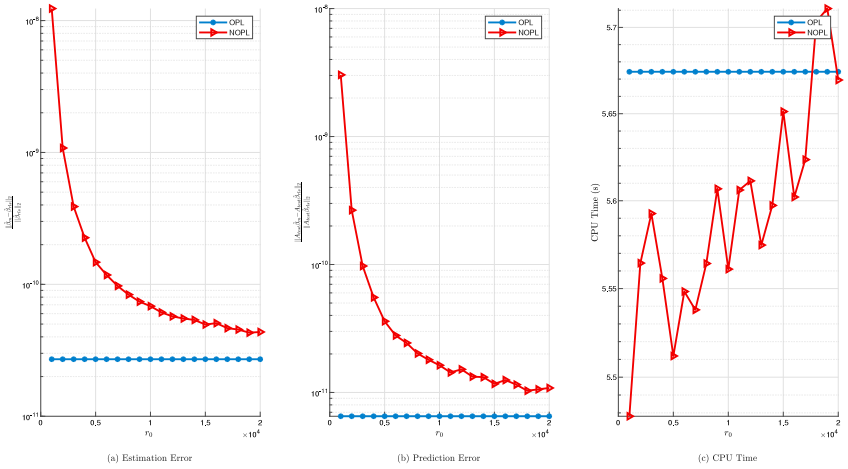

Fig. 12. Comparison of OPL and NOPL with different r_0 and Gene Expression Cancer RNA-Seq data set.

Table 6
Comparison of estimation errors using different m for example 2.

Methods	m					
	1	4	10	13	16	22
OPL	0.14132	0.00038	2.8853e−09	7.9237e−12	2.3316e−14	3.2211e−15
NOPL	0.15264	0.00053	6.0318e−09	2.0625e−11	6.9721e−14	3.2309e−15
LEV	0.16803	0.00081	1.8649e−08	8.9978e−11	3.9035e−13	3.2226e−15
RLEV	0.16685	0.00079	1.8415e−08	8.1578e−11	3.9137e−13	3.2184e−15
COL	0.16751	0.00080	1.805e−08	8.7775e−11	4.3033e−13	3.2151e−15
UNI	0.16783	0.00084	1.8114e−08	8.5709e−11	4.4034e−13	3.2336e−15

Table 7
Comparison of prediction errors using different m for example 2.

Methods	m					
	1	4	10	13	16	22
OPL	0.158	0.00042	3.2237e−09	8.8176e−12	2.6332e−14	2.714e−15
NOPL	0.17002	0.00059	6.8145e−09	2.3055e−11	7.7973e−14	2.726e−15
LEV	0.18753	0.00091	2.0976e−08	1.0165e−10	4.3414e−13	2.7205e−15
RLEV	0.18829	0.00089	2.0577e−08	9.2623e−11	4.3598e−13	2.7082e−15
COL	0.18507	0.00091	2.0125e−08	9.6612e−11	4.7958e−13	2.7158e−15
UNI	0.18471	0.00093	2.0088e−08	9.6145e−11	4.959e−13	2.7352e−15

CRediT authorship contribution statement

Hanyu Li: Conceptualization, Funding acquisition, Methodology, Supervision, Writing – original draft, Writing – review & editing. **Chengmei Niu:** Data curation, Investigation, Methodology, Software, Visualization, Writing – original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Appendix A. Proof of Theorem 2.1

We start by establishing two lemmas.

Lemma A.1. *Assuming that the conditions (2.4), (2.5) and (2.6) are satisfied, we have*

$$\sum_{i=1}^{p} \pi_i \| \frac{e_i}{p} \|_2^3 = O_p(1), \tag{A.1}$$

where $e_i = (\frac{A_i A_i^T}{\pi_i} + \lambda I)\hat{z}^* - \widetilde{y}$ *with* $\widetilde{y} = \lambda y$ *and* $\hat{z}^*$ *being as in (1.3).*

Table 8
Description of two experiments using Gene Expression Cancer RNA-Seq data set.

Kinds	Comparison	r	λ	m	r_0	Results
1	six methods	5000 to 10000	10	16	5000 (NOPL)	Figs. 9–11(a)
		8000	1 to 50	16	5000 (NOPL)	Figs. 9–11(b)
		8000	10	1 to 26	5000 (NOPL)	Tables 9–10 & Fig. 11(c)
2	OPL and NOPL	8000	10	16	1000 to 20531 (NOPL)	Fig. 12

Table 9
Comparison of estimation errors using different m for Gene Expression Cancer RNA-Seq data set.

Methods	m				
	1	4	13	16	19
OPL	0.21365	**0.00226**	2.5971e−09	2.7892e−11	2.7727e−13
NOPL	0.23603	**0.00335**	1.0478e−08	1.5595e−10	2.202e−12
LEV	0.24382	0.0038	1.244e−08	1.9204e−10	2.8633e−12
RLEV	0.25115	0.0038	1.2808e−08	1.864e−10	2.9497e−12
COL	0.26221	0.00619	9.6599e−08	2.391e−09	6.0913e−11
UNI	0.46642	0.05523	0.00193	0.00053	0.00056

Table 10
Comparison of prediction errors using different m for Gene Expression Cancer RNA-Seq data set.

Methods	m				
	1	4	13	16	19
OPL	0.06048	**0.00053**	6.3389e−10	6.5315e−12	6.632e−14
NOPL	0.06688	**0.00081**	2.633e−09	3.8972e−11	5.344e−13
LEV	0.08874	0.00099	3.3232e−09	5.4092e−11	7.7124e−13
RLEV	0.096764	0.00107	3.3518e−09	4.8333e−11	7.8531e−13
COL	0.07601	0.00137	2.0208e−08	5.2209e−10	1.3269e−11
UNI	0.20076	0.01479	0.00036	9.664e−05	8.8783e−05

Proof. With $e_i = (\frac{A_iA_i^T}{\pi_i} + \lambda I)\hat{z}^* - \widetilde{y}$ and (1.3), it is easy to see that

$$\sum_{i=1}^{p}\pi_i\|\frac{e_i}{p}\|_2^3 = \frac{1}{p^3}\sum_{i=1}^{p}\pi_i\|(\frac{A_iA_i^T}{\pi_i} + \lambda I)(AA^T+\lambda I)^{-1}\widetilde{y} - \widetilde{y}\|_2^3.$$

Then, considering the basic triangle inequality and the fact that $\sum_{i=1}^{p}\pi_i = 1$, we can have

$$\begin{aligned}\sum_{i=1}^{p}\pi_i\|\frac{e_i}{p}\|_2^3 \le& \frac{1}{p^3}[\sum_{i=1}^{p}\pi_i\|(\frac{A_iA_i^T}{\pi_i} + \lambda I)(AA^T+\lambda I)^{-1}\widetilde{y}\|_2^3] + \frac{\|\widetilde{y}\|_2^3}{p^3}\\ &+ 3\frac{1}{p^3}[\sum_{i=1}^{p}\pi_i\|(\frac{A_iA_i^T}{\pi_i} + \lambda I)(AA^T+\lambda I)^{-1}\widetilde{y}\|_2^2\|\widetilde{y}\|_2]\\ &+ 3\frac{1}{p^3}[\sum_{i=1}^{p}\pi_i\|(\frac{A_iA_i^T}{\pi_i} + \lambda I)(AA^T+\lambda I)^{-1}\widetilde{y}\|_2\|\widetilde{y}\|_2^2]\\ \le& \frac{\|\widetilde{y}\|_2^3\sigma_n^3(AA^T+\lambda I)}{p^3}(\sum_{i=1}^{p}\frac{\|A_iA_i^T\|_2^3}{\pi_i^2} + 3\lambda\sum_{i=1}^{p}\frac{\|A_iA_i^T\|_2^2}{\pi_i}\\ &+ 3\lambda^2\sum_{i=1}^{p}\|A_iA_i^T\|_2 + \lambda^3) + \frac{\|\widetilde{y}\|_2^3}{p^3}\\ &+ 3\frac{\|\widetilde{y}\|_2^3\sigma_n^2(AA^T+\lambda I)}{p^3}(\sum_{i=1}^{p}\frac{\|A_iA_i^T\|_2^2}{\pi_i} + 2\lambda\sum_{i=1}^{p}\|A_iA_i^T\|_2 + \lambda^2)\\ &+ 3\frac{\|\widetilde{y}\|_2^3\sigma_n(AA^T+\lambda I)}{p^3}(\sum_{i=1}^{p}\|A_iA_i^T\|_2 + \lambda). \end{aligned} \tag{A.2}$$

Following

$$\frac{\|\widetilde{y}\|_2^2}{p} = o_p(1), \tag{A.3}$$

which can be derived from $np^{-1}\to 0$, and noting (2.4), (2.5), (2.6) and (A.2), we can get

$$\begin{aligned}\sum_{i=1}^{p}\pi_i\|\frac{e_i}{p}\|_2^3 \le& \frac{\|\widetilde{y}\|_2^3\sigma_n^3(AA^T+\lambda I)}{p^3}(\sum_{i=1}^{p}\frac{\|A_i\|_2^6}{\pi_i^2})\\ &+ o_p(1) \quad \text{by (2.5), (2.6), (A.2), and (A.3)}\\ =&O_p(1). \quad \text{by (2.4)}\end{aligned}$$

Thus, (A.1) is arrived. □

Lemma A.2. *Suppose that the conditions (2.5) and (2.7) hold. Then, conditional on $\mathcal{F}_n$ in probability,*

$$\frac{\widehat{M}_A - M_A}{p} = O_{p|\mathcal{F}_n}(r^{-1/2}), \tag{A.4}$$

$$\frac{e^*}{p} = O_{p|\mathcal{F}_n}(r^{-1/2}), \tag{A.5}$$

where $M_A = AA^T + \lambda I$, $\hat{M}_A = ASS^TA^T + \lambda I$ with $S\in\mathbb{R}^{p\times r}$ constructed as in Algorithm 1, and $e^* = (\hat{M}_A\hat{z}^* - \widetilde{y})$ with $\widetilde{y} = \lambda y$ and $\hat{z}^*$ being as in (1.3).

Proof. First, note that

$$\begin{aligned}&\frac{1}{p^2}\sum_{i=1}^{p}\pi_i[(\frac{A_iA_i^T}{\pi_i} + \lambda I) - (AA^T+\lambda I)][(\frac{A_iA_i^T}{\pi_i} + \lambda I) - (AA^T+\lambda I)]\\ &= \frac{1}{p^2}\sum_{i=1}^{p}\pi_i(\frac{A_iA_i^T}{\pi_i} - AA^T)(\frac{A_iA_i^T}{\pi_i} - AA^T)\\ &= \frac{1}{p^2}\sum_{i=1}^{p}\frac{A_iA_i^TA_iA_i^T}{\pi_i} - \frac{AA^TAA^T}{p^2}\\ &= O_p(1),\end{aligned}$$

where the last equality is from (2.5) and (2.7). This result implies, for any n-dimensional vector ℓ with finite elements,

$$\begin{aligned}&\frac{1}{p^2}\sum_{i=1}^{p}\pi_i[(\frac{A_iA_i^T}{\pi_i} + \lambda I) - (AA^T+\lambda I)]\ell\ell^T[(\frac{A_iA_i^T}{\pi_i} + \lambda I) - (AA^T+\lambda I)]\\ &= \frac{1}{p^2}\sum_{i=1}^{p}\frac{A_iA_i^T\ell\ell^TA_iA_i^T}{\pi_i} - \frac{AA^T\ell\ell^TAA^T}{p^2} = O_p(1).\end{aligned} \tag{A.6}$$

Thus, following $\mathrm{E}(\hat{M}_A \mid A) = M_A$, it is natural to get

$$\begin{aligned}\mathrm{Var}(\frac{(\hat{M}_A - M_A)\ell}{p} \mid A) =& \mathrm{E}[(\frac{\hat{M}_A - M_A}{p})\ell\ell^T(\frac{\hat{M}_A - M_A}{p}) \mid A]\\ =&\frac{1}{rp^2}\sum_{i=1}^{p}\pi_i[(\frac{A_iA_i^T}{\pi_i} + \lambda I)\\ &- (AA^T+\lambda I)]\ell\ell^T[(\frac{A_iA_i^T}{\pi_i} + \lambda I) - (AA^T+\lambda I)]\\ =&O_p(r^{-1}),\end{aligned}$$

which together with the Markov's inequality implies (A.4).

Combining (A.3) and (A.6), we can get

$$\begin{aligned}&\frac{1}{p^2}\sum_{i=1}^{p}\pi_i\hat{z}^{*T}(\frac{A_iA_i^T}{\pi_i} + \lambda I)\ell\ell^T(\frac{A_iA_i^T}{\pi_i} + \lambda I)\hat{z}^*\\ &= \frac{1}{p^2}\hat{z}^{*T}(AA^T+\lambda I)\ell\ell^T(AA^T+\lambda I)\hat{z}^* + O_p(1).\end{aligned} \tag{A.7}$$

Thus, considering $e^* = \sum_{t=1}^{r}\frac{1}{r}e_{i_t}$ with $e_{i_t} = (\frac{A_{i_t}A_{i_t}^T}{\pi_{i_t}} + \lambda I)\hat{z}^* - \widetilde{y}$ and $\mathrm{E}(e_{i_t} \mid \mathcal{F}_n) = 0$, and (A.7), we can obtain

$$\mathrm{Var}(\frac{\ell^Te^*}{p} \mid \mathcal{F}_n) = \ell^T\mathrm{E}[(\frac{e^*}{p})(\frac{e^*}{p})^T \mid \mathcal{F}_n]\ell = \frac{1}{rp^2}\ell^T(\sum_{i=1}^{p}\pi_ie_ie_i^T)\ell$$

Table 11
Description of two experiments using Gisette data set.

Kinds	Comparison	r	λ	m	r_0	Results
1	six methods	1000 to 4300	10	10	900 (NOPL)	Tables 12–13
		2000	10	4 to 16	900 (NOPL)	Tables 12–13
2	OPL and NOPL	2000	10	10	700 to 4700 (NOPL)	Table 14

Table 12
Comparison of accuracy using different r and m for Gisette data set.

Methods	r				m		
	1000	1900	2500	4300	4	10	16
OPL	3.5349e−06	3.982e−08	7.6293e−09	3.1004e−10	0.00036	2.9281e−08	2.72e−12
NOPL	1.2591e − 05	1.7699e−07	2.9862e−08	1.3174e−09	0.00052	1.0058e−07	1.7147e−11
LEV	1.3525e−05	1.6463e−07	4.1241e−08	2.0237e−09	0.00066	1.1394e−07	2.7071e−11
RLEV	1.2316e−05	1.7819e−07	4.015e−08	2.0591e−09	0.00076	1.2599e−07	2.5367e−11
COL	2.3882e−05	2.5821e−07	5.2512e−08	2.3897e−09	0.00057	2.2314e−07	6.3088e−11
UNI	0.0017	9.6027e−06	2.3042e−06	6.0028e−08	0.00285	6.4834e−06	2.1745e−08

Table 13
Comparison of CPU time using different r and m for Gisette data set.

Methods	r				m		
	1000	1900	2500	4300	4	10	16
OPL	0.14007	0.1697	0.18937	0.25315	0.078267	0.17318	0.27027
NOPL	0.13402	0.16328	0.18895	0.25257	0.077768	0.16846	0.26195
LEV	0.15385	0.18681	0.20962	0.27081	0.10166	0.19108	0.27654
RLEV	0.16093	0.19294	0.21372	0.28097	0.11265	0.19875	0.28931
COL	0.11739	0.15268	0.17089	0.23678	0.06894	0.15783	0.25586
UNI	0.11008	0.1474	0.16925	0.22774	0.066508	0.15448	0.24998

$$
\begin{aligned}
&=\frac{1}{rp^2}\ell^T[\sum_{i=1}^{p}\pi_i((\frac{A_iA_i^T}{\pi_i}+\lambda I)\hat{z}^*-\widetilde{y})((\frac{A_iA_i^T}{\pi_i}+\lambda I)\hat{z}^*-\widetilde{y})^T]\ell\\
&=\frac{1}{rp^2}\sum_{i=1}^{p}\pi_i\hat{z}^{*T}(\frac{A_iA_i^T}{\pi_i}+\lambda I)\ell\ell^T(\frac{A_iA_i^T}{\pi_i}+\lambda I)\hat{z}^*-\frac{\ell^T\widetilde{y}\widetilde{y}^T\ell}{rp^2}\\
&=\frac{1}{r}[\frac{1}{p^2}\hat{z}^{*T}(AA^T+\lambda I)\ell\ell^T(AA^T+\lambda I)\hat{z}^*\\
&\quad+O_p(1)-\frac{\ell^T\widetilde{y}\widetilde{y}^T\ell}{p^2}]\quad\text{by (A.7)}\\
&=\frac{1}{r}[\frac{\widetilde{y}^T\ell\ell^T\widetilde{y}}{p^2}+O_p(1)-\frac{\ell^T\widetilde{y}\widetilde{y}^T\ell}{p^2}]\\
&=O_p(r^{-1}).
\end{aligned}
$$

Consequently, by the Markov's inequality, (A.5) is obtained. □

Proof of Theorem 2.1. Considering that

$$
\begin{aligned}
&\hat{z}=(ASS^TA^T+\lambda I)^{-1}\widetilde{y}=\hat{M}_A^{-1}\widetilde{y},\\
&\hat{z}^*=(AA^T+\lambda I)^{-1}\widetilde{y}=M_A^{-1}\widetilde{y},
\end{aligned}
$$

where $\widetilde{y}=\lambda y$, we can rewrite $\hat{z}-\hat{z}^*$ as

$$
\begin{aligned}
\hat{z}-\hat{z}^* &= (ASS^TA^T+\lambda I)^{-1}(\widetilde{y}-(ASS^TA^T+\lambda I)\hat{z}^*)\\
&= \hat{M}_A^{-1}(\widetilde{y}-\hat{M}_A\hat{z}^*)=-\hat{M}_A^{-1}e^*\\
&= -(\hat{M}_A^{-1}-M_A^{-1}+M_A^{-1})e^*\\
&= -M_A^{-1}e^*-(\hat{M}_A^{-1}-M_A^{-1})e^*\\
&= -M_A^{-1}e^*+\hat{M}_A^{-1}(\hat{M}_A-M_A)M_A^{-1}e^*\\
&= -(\frac{M_A}{p})^{-1}\frac{e^*}{p}+(\frac{\hat{M}_A}{p})^{-1}(\frac{\hat{M}_A-M_A}{p})(\frac{M_A}{p})^{-1}\frac{e^*}{p} \qquad (A.8)\\
&= -(\frac{M_A}{p})^{-1}\frac{e^*}{p}+O_{p|\mathcal{F}_n}(r^{-1}), \qquad (A.9)
\end{aligned}
$$

where the last equality is derived by (2.7) and Lemma A.2. Thus, to prove (2.9), we first prove

$$
(\frac{V_c}{r})^{-1/2}(\frac{e^*}{p})\xrightarrow{L}N(0,I). \qquad (A.10)
$$

Recall that $\frac{e_*}{p}=\sum_{t=1}^{r}\frac{1}{rp}e_{i_t}$ with

$$
e_{i_t}=(\frac{A_{i_t}A_{i_t}^T}{\pi_{i_t}}+\lambda I)\hat{z}^*-\widetilde{y}.
$$

Now, we construct the sequence $\{\frac{e_{i_t}}{p}\}_{t=1}^{r}$. These random vectors are independent and identically distributed and it is easy to get that $\mathrm{E}(\frac{e_{i_t}}{p}\mid\mathcal{F}_n)=0$. Furthermore, noting that

$$
V_c=\sum_{i=1}^{p}\frac{A_iA_i^T\hat{z}^*\hat{z}^{*T}A_iA_i^T}{p^2\pi_i}=O_p(1), \qquad (A.11)
$$

which can be obtained from (2.5), together with (2.7) and (A.3), we have

$$
\begin{aligned}
\mathrm{Var}(\frac{e_{i_t}}{p}\mid\mathcal{F}_n)&=\mathrm{E}(\frac{e_{i_t}e_{i_t}^T}{p^2}\mid\mathcal{F}_n)=\sum_{i=1}^{p}\pi_i\frac{e_ie_i^T}{p^2}\\
&=\sum_{i=1}^{p}\pi_i\frac{[(\frac{A_iA_i^T}{\pi_i}+\lambda I)\hat{z}^*-\widetilde{y}][(\frac{A_iA_i^T}{\pi_i}+\lambda I)\hat{z}^*-\widetilde{y}]^T}{p^2}\\
&=\sum_{i=1}^{p}\pi_ip^{-2}\frac{A_iA_i^T}{\pi_i}\hat{z}^*\hat{z}^{*T}\frac{A_iA_i^T}{\pi_i}+\frac{(\lambda\hat{z}^*-\widetilde{y})\hat{z}^{*T}AA^T}{p^2}\\
&\quad+\frac{AA^T\hat{z}^*(\lambda\hat{z}^*-\widetilde{y})^T}{p^2}+\frac{(\lambda\hat{z}^*-\widetilde{y})(\lambda\hat{z}^*-\widetilde{y})^T}{p^2}\\
&=\sum_{i=1}^{p}\frac{A_iA_i^T\hat{z}^*\hat{z}^{*T}A_iA_i^T}{p^2\pi_i}+o_p(1)\quad\text{by (2.7) and (A.3)}\\
&=V_c+o_p(1) \qquad (A.12)\\
&=O_p(1).\quad\text{by (A.11)} \qquad (A.13)
\end{aligned}
$$

In addition, for any $\xi>0$, we have

$$
\begin{aligned}
&\sum_{t=1}^{r}\mathrm{E}[\|r^{-\frac{1}{2}}p^{-1}e_{i_t}\|_2^2I(\|r^{-\frac{1}{2}}p^{-1}e_{i_t}\|_2>\xi)\mid\mathcal{F}_n]\\
&=\sum_{i=1}^{p}\pi_i\|p^{-1}e_i\|_2^2I(\|r^{-\frac{1}{2}}p^{-1}e_i\|_2>\xi)\\
&\le(r^{\frac{1}{2}}\xi)^{-1}\sum_{i=1}^{p}\pi_i\|p^{-1}e_i\|_2^3\\
&=o_p(1),
\end{aligned}
$$

where the inequality is deduced by the constraint $I(\|r^{-\frac{1}{2}}p^{-1}e_i\|_2>\xi)$, and the last equality is from Lemma A.1. Putting the above discussions together, we find that the Lindeberg–Feller conditions are satisfied in probability. Thus, by the Lindeberg–Feller central limit theorem [30, Proposition 2.27], and noting (A.13), we can acquire

$$
[\mathrm{Var}(\frac{e_{i_t}}{p}\mid\mathcal{F}_n)]^{-1/2}(r^{-1/2}p^{-1}\sum_{t=1}^{r}e_{i_t})\xrightarrow{L}N(0,I),
$$

Table 14
Comparison of OPL and NOPL with different r_0 for Gisette data set.

Methods	Accuracy				CPU time (s)			
	700	1700	3900	4700	700	1700	3900	4700
NOPL	1.5237e − 07	5.9818e−08	4.9386e−08	3.4255e−08	0.15302	0.15994	0.16707	0.17122
OPL	2.5851e−08				0.16999			

which combined with $\frac{e_*}{p} = r^{-1}p^{-1}\sum_{t=1}^{r} e_{i_t}$ and $\mathrm{Var}(\frac{e_*}{p} \mid \mathcal{F}_n) = r^{-1}\mathrm{Var}(\frac{e_{i_t}}{p} \mid \mathcal{F}_n)$ gives

$$[r^{-1}\mathrm{Var}(\frac{e_{i_t}}{p} \mid \mathcal{F}_n)]^{-1/2}(\frac{e_*}{p}) \xrightarrow{L} N(0, I).$$

Thus, by Lemma A.2, (A.12), and the Slutsky's Theorem [37, Theorem 6], we can get (A.10).

Now, we prove (2.9). Following (2.7) and (A.11), it is easy to get

$$V = (\frac{M_A}{p})^{-1}\frac{V_c}{r}(\frac{M_A}{p})^{-1} = O_p(r^{-1}),$$

which together with (A.9) yields

$$\begin{aligned} V^{-1/2}(\hat{z} - \hat{z}^*) &= -V^{-1/2}(\frac{M_A}{p})^{-1}\frac{e_*}{p} + O_{p|\mathcal{F}_n}(r^{-1/2}) \\ &= -V^{-1/2}(\frac{M_A}{p})^{-1}(\frac{V_c}{r})^{1/2}(\frac{V_c}{r})^{-1/2}\frac{e_*}{p} + O_{p|\mathcal{F}_n}(r^{-1/2}). \end{aligned} \tag{A.14}$$

In addition, it is verified that

$$\begin{aligned} &V^{-1/2}(\frac{M_A}{p})^{-1}(\frac{V_c}{r})^{1/2}[V^{-1/2}(\frac{M_A}{p})^{-1}(\frac{V_c}{r})^{1/2}]^T \\ &= V^{-1/2}(\frac{M_A}{p})^{-1}(\frac{V_c}{r})^{1/2}(\frac{V_c}{r})^{1/2}(\frac{M_A}{p})^{-1}V^{-1/2} = I. \end{aligned} \tag{A.15}$$

Thus, combining (A.10), (A.14), and (A.15), by the Slutsky's Theorem, we get the desired result (2.9). □

Appendix B. Proof of Theorem 2.2

According to the Cauchy–Schwarz inequality, we have

$$\begin{aligned} \mathrm{tr}(V_c) &= \sum_{i=1}^{p}\frac{A_i^T\hat{z}^*\hat{z}^{*T}A_i\|A_i\|_2^2}{p^2\pi_i} = \lambda^2\sum_{i=1}^{p}\frac{\hat{\beta}_{rls(i)}^2\|A_i\|_2^2}{p^2\pi_i} \\ &= \frac{\lambda^2}{p^2}\sum_{i=1}^{p}\pi_i\sum_{i=1}^{p}\frac{\hat{\beta}_{rls(i)}^2\|A_i\|_2^2}{\pi_i} \geq \frac{\lambda^2}{p^2}(\sum_{i=1}^{p}|\hat{\beta}_{rls(i)}|\|A_i\|_2)^2, \end{aligned}$$

where the equality in the last inequality holds if and only if π_i is proportional to $|\hat{\beta}_{rls(i)}| \|A_i\|_2$ for some constant $C_0 \geq 0$. Thus, following $\sum_{i=1}^{p}\pi_i = 1$, the desired result (2.10) is obtained.

Appendix C. Proof of Theorem 2.3

We first present two auxiliary lemmas.

Lemma C.1 (*[8, Theorem 23]*). *If J, $H \in \mathbb{R}^{m\times m}$ are real symmetric positive semi-definite matrices such that $\sigma_1(J) \geq \sigma_2(J) \geq \cdots \geq \sigma_m(J)$ and $\sigma_1(H) \geq \sigma_2(H) \geq \cdots \geq \sigma_m(H)$, then*

$$\sigma_j(J - H) \leq \sigma_j\begin{pmatrix} J & 0 \\ 0 & H \end{pmatrix}, \quad j = 1, \ldots, m.$$

Especially,

$$\|J - H\|_2 \leq \max\{\|J\|_2, \|H\|_2\}.$$

Lemma C.2. *For S established by $\pi_i = \pi_i^{OPL}$, assuming that (2.12) holds and letting $r \geq \frac{8s_2c_2\rho}{3s_1c_1\epsilon'^2}\ln(\frac{4\rho}{\delta})$ with $\epsilon' \in (0, \frac{1}{2})$ and $\delta \in (0, 1)$, we have*

$$\|V^TSS^TV - I\|_2 \leq \epsilon',$$

with the probability at least $1 - \delta$.

Proof. The proof can be accomplished along the line of the proof of [8, Theorem 3]. However, for our case, it is necessary to note that

$$\begin{aligned} \|F_t\|_2 &= \|M_tM_t^T - \frac{V^TV}{r}\|_2 \leq \max\{\|M_tM_t^T\|_2, \frac{1}{r}\} \quad \text{by Lemma C.1} \\ &= \frac{1}{r}\max_{1\leq i\leq p}\{\|\frac{(V^i)^T}{\sqrt{\pi_i^{OPL}}}\frac{V^i}{\sqrt{\pi_i^{OPL}}}\|_2, 1\} = \frac{1}{r}\max_{1\leq i\leq p}\{\frac{\|V^i\|_2^2}{\pi_i^{OPL}}, 1\} \\ &= \frac{1}{r}\max_{1\leq i\leq p}\{\frac{\|V^i\|_2^2\sum_{i=1}^{p}|\hat{\beta}_{rls(i)}|\|A_i\|_2}{|\hat{\beta}_{rls(i)}|\|A_i\|_2}, 1\} \quad \text{by (2.10)} \\ &\leq \frac{1}{r}\max_{1\leq i\leq p}\{\frac{\|V^i\|_2^2\sum_{i=1}^{p}s_2c_2\|V^i\|_2^2}{s_1c_1\|V^i\|_2^2}, 1\} \quad \text{by (2.12)} \\ &\leq \frac{1}{r}\max_{1\leq i\leq p}\{\frac{s_2c_2}{s_1c_1}\sum_{i=1}^{p}\|V^i\|_2^2, 1\} \leq \frac{1}{r}\max_{1\leq i\leq p}\{\frac{s_2c_2}{s_1c_1}\rho, 1\} \leq \frac{s_2c_2\rho}{rs_1c_1} \end{aligned}$$

and

$$\begin{aligned} \mathrm{E}(F_t^2) + \frac{(V^TV)^2}{r^2} &= \mathrm{E}(M_tM_t^T\|M_t\|_2^2) = \sum_{i=1}^{p}\pi_i^{OPL}\frac{(V^i)^TV^i\|V^i\|_2^2}{r^2(\pi_i^{OPL})^2} \\ &= \frac{1}{r^2}\sum_{i=1}^{p}\frac{(V^i)^TV^i\|V^i\|_2^2\sum_{i=1}^{p}|\hat{\beta}_{rls(i)}|\|A_i\|_2}{|\hat{\beta}_{rls(i)}|\|A_i\|_2} \quad \text{by (2.10)} \\ &\leqslant \frac{1}{r^2}\sum_{i=1}^{p}\frac{(V^i)^TV^i\|V^i\|_2^2\sum_{i=1}^{p}s_2c_2\|V^i\|_2^2}{s_1c_1\|V^i\|_2^2} \quad \text{by (2.12)} \\ &\leqslant \frac{s_2c_2}{r^2s_1c_1}\sum_{i=1}^{p}(V^i)^TV^i\sum_{i=1}^{p}\|V^i\|_2^2 \\ &= \frac{s_2c_2\rho}{r^2s_1c_1}\sum_{i=1}^{p}(V^i)^TV^i = \frac{s_2c_2\rho}{r^2s_1c_1}I_\rho, \end{aligned}$$

where $F_t = M_tM_t^T - \frac{V^TV}{r}$ with $M_t = \frac{(V^{i_t})^T}{\sqrt{r\pi_{i_t}^{OPL}}}$ and $t = 1, \ldots, r$. □

Proof of Theorem 2.3. Noting $\hat{\beta} = \frac{1}{\lambda}V\Sigma U^T\hat{z}$ and $\hat{\beta}_{rls} = \frac{1}{\lambda}V\Sigma U^T\hat{z}^*$, we can rewrite (2.13) as

$$\frac{1}{\lambda}\|\Sigma U^T(\hat{z} - \hat{z}^*)\|_2 \leq \frac{\epsilon}{\lambda}\|\Sigma U^T\hat{z}^*\|_2. \tag{C.1}$$

To prove (C.1), we define the loss functions $L(z)$ and $\hat{L}(z)$ as

$$L(z) = \frac{1}{2\lambda}\|A^Tz\|_2^2 + \frac{1}{2}\|z\|_2^2 - z^Ty$$

and

$$\hat{L}(z) = \frac{1}{2\lambda}\|S^TA^Tz\|_2^2 + \frac{1}{2}\|z\|_2^2 - z^Ty.$$

Thus, by Taylor expansion, we can acquire

$$\hat{L}(\hat{z}) = \hat{L}(\hat{z}^*) + (\hat{z} - \hat{z}^*)^T\nabla\hat{L}(\hat{z}^*) + (\hat{z} - \hat{z}^*)^T\nabla^2\hat{L}(z_0)(\hat{z} - \hat{z}^*), \tag{C.2}$$

where $\hat{z}^*$ and $\hat{z}$ minimize the loss functions $L(z)$ and $\hat{L}(z)$, respectively, and $z_0 \in [\hat{z}, \hat{z}^*]$. Moreover, following $(\nabla^2\hat{L}(z_0) - \nabla^2L(z_0))\hat{z}^* = \nabla\hat{L}(\hat{z}^*) - \nabla L(\hat{z}^*)$, which is from

$$\nabla\hat{L}(\hat{z}^*) = (\frac{1}{\lambda}ASS^TA^T + I)\hat{z}^* - y, \ \nabla L(\hat{z}^*) = (\frac{1}{\lambda}AA^T + I)\hat{z}^* - y,$$

and

$$\nabla^2\hat{L}(z_0) = \frac{1}{\lambda}ASS^TA^T + I, \ \nabla^2L(z_0) = \frac{1}{\lambda}AA^T + I, \tag{C.3}$$

we can obtain

$$\hat{L}(\hat{z}^*) + (\hat{z} - \hat{z}^*)^T(\nabla^2\hat{L}(z_0) - \nabla^2L(z_0))\hat{z}^* = \hat{L}(\hat{z}^*) + (\hat{z} - \hat{z}^*)^T(\nabla\hat{L}(\hat{z}^*) - \nabla L(\hat{z}^*)).$$

Thus, considering that

$$\hat{L}(\hat{z}^*) + (\hat{z} - \hat{z}^*)^T(\nabla\hat{L}(\hat{z}^*) - \nabla L(\hat{z}^*)) \le \hat{L}(\hat{z}^*) + (\hat{z} - \hat{z}^*)^T\nabla\hat{L}(\hat{z}^*),$$

which is derived by the fact $(\hat{z} - \hat{z}^*)^T\nabla L(\hat{z}^*) \ge 0$, and noting (C.2), we can gain

$$\hat{L}(\hat{z}^*) + (\hat{z} - \hat{z}^*)^T(\nabla^2\hat{L}(z_0) - \nabla^2 L(z_0))\hat{z}^* \le \hat{L}(\hat{z}) - (\hat{z} - \hat{z}^*)^T\nabla^2\hat{L}(z_0)(\hat{z} - \hat{z}^*).$$

Further, by $\hat{L}(\hat{z}^*) \ge \hat{L}(\hat{z})$, we have

$$(\hat{z} - \hat{z}^*)^T(\nabla^2 L(z_0) - \nabla^2\hat{L}(z_0))\hat{z}^* \ge (\hat{z} - \hat{z}^*)^T\nabla^2\hat{L}(z_0)(\hat{z} - \hat{z}^*),$$

which together with

$$(\hat{z} - \hat{z}^*)^T\nabla^2\hat{L}(z_0)(\hat{z} - \hat{z}^*) \ge (\hat{z} - \hat{z}^*)^T\frac{1}{\lambda}ASS^TA^T(\hat{z} - \hat{z}^*)$$

and (C.3) leads to

$$(\hat{z} - \hat{z}^*)^T(\frac{1}{\lambda}AA^T - \frac{1}{\lambda}ASS^TA^T)\hat{z}^* \ge (\hat{z} - \hat{z}^*)^T\frac{1}{\lambda}ASS^TA^T(\hat{z} - \hat{z}^*).$$

Thus, based on $A = U\Sigma V^T$, it is straightforward to get

$$\begin{aligned}&\frac{1}{\lambda^2}(\hat{z} - \hat{z}^*)^T(U\Sigma^2U^T - U\Sigma V^TSS^TV\Sigma U^T)\hat{z}^*\\ &\quad\ge \frac{1}{\lambda^2}(\hat{z} - \hat{z}^*)^TU\Sigma V^TSS^TV\Sigma U^T(\hat{z} - \hat{z}^*),\end{aligned}$$

which is also allowed to be rewritten as

$$\begin{aligned}&\frac{1}{\lambda^2}[\Sigma U^T(\hat{z} - \hat{z}^*)]^T(I - V^TSS^TV)\Sigma U^T\hat{z}^*\\ &\quad\ge \frac{1}{\lambda^2}[\Sigma U^T(\hat{z} - \hat{z}^*)]^TV^TSS^TV[\Sigma U^T(\hat{z} - \hat{z}^*)].\end{aligned} \tag{C.4}$$

Adding $\frac{1}{\lambda^2}[\Sigma U^T(\hat{z} - \hat{z}^*)]^T[\Sigma U^T(\hat{z} - \hat{z}^*)]$ to both sides of (C.4) gives

$$\begin{aligned}&\frac{1}{\lambda^2}[\Sigma U^T(\hat{z} - \hat{z}^*)]^T(I - V^TSS^TV)\Sigma U^T\hat{z}^*\\ &\quad+ \frac{1}{\lambda^2}[\Sigma U^T(\hat{z} - \hat{z}^*)]^T(I - V^TSS^TV)[\Sigma U^T(\hat{z} - \hat{z}^*)]\\ &\ge \frac{1}{\lambda^2}[\Sigma U^T(\hat{z} - \hat{z}^*)]^T[\Sigma U^T(\hat{z} - \hat{z}^*)].\end{aligned} \tag{C.5}$$

Taking the Euclidean norm on both sides of (C.5), we obtain

$$\begin{aligned}&\frac{1}{\lambda^2}\|\Sigma U^T(\hat{z} - \hat{z}^*)\|_2\|I - V^TSS^TV\|_2\|\Sigma U^T\hat{z}^*\|_2\\ &+ \frac{1}{\lambda^2}\|\Sigma U^T(\hat{z} - \hat{z}^*)\|_2\|I - V^TSS^TV\|_2\|\Sigma U^T(\hat{z} - \hat{z}^*)\|_2\\ &\ge \frac{1}{\lambda^2}\|\Sigma U^T(\hat{z} - \hat{z}^*)\|_2^2,\end{aligned}$$

which combined with Lemma C.2 indicates that

$$\frac{1}{\lambda}\epsilon'\|\Sigma U^T\hat{z}^*\|_2 + \frac{1}{\lambda}\epsilon'\|\Sigma U^T(\hat{z} - \hat{z}^*)\|_2 \ge \frac{1}{\lambda}\|\Sigma U^T(\hat{z} - \hat{z}^*)\|_2. \tag{C.6}$$

By rewriting (C.6) as

$$\frac{1}{\lambda}\|\Sigma U^T(\hat{z} - \hat{z}^*)\|_2 \le \frac{\epsilon'}{1 - \epsilon'}\frac{1}{\lambda}\|\Sigma U^T\hat{z}^*\|_2$$

and considering the fact $\epsilon' < \frac{1}{2}$, we have

$$\frac{1}{\lambda}\|\Sigma U^T(\hat{z} - \hat{z}^*)\|_2 \le \frac{2\epsilon'}{\lambda}\|\Sigma U^T\hat{z}^*\|_2.$$

Thus, setting $\epsilon = 2\epsilon'$, we get (C.1). That is, (2.13) is arrived. □

Appendix D. Proof of Theorem 2.4

The proof can be completed along the line of the proof of Theorem 6 in [5]. However, when we bound $\|R\|_2$ with

$$R = (\lambda\Sigma^{-1} + \Sigma)^{-1}\Sigma(V^TS^TSV - I),$$

Lemma C.2 is adopted but not the oblivious subspace embedding theorem [5, Theorem 5], namely,

$$\begin{aligned}\|R\|_2 &\le \|(\lambda\Sigma^{-1} + \Sigma)^{-1}\Sigma(V^TS^TSV - I)\|_2\\ &\le \|(\lambda\Sigma^{-1} + \Sigma)^{-1}\Sigma\|_2\|V^TS^TSV - I\|_2\\ &\le \epsilon'\|(\lambda\Sigma^{-1} + \Sigma)^{-1}\Sigma\|_2 \quad \text{by Lemma C.2}\\ &\le \epsilon',\end{aligned}$$

where ϵ' satisfies $\epsilon' = \frac{\epsilon}{2}$.

Appendix E. Proof of Theorem 3.1

The proof is similar to the one of Theorem 2.1 (see Appendix A), and we begin by presenting two lemmas.

Lemma E.1. *Assume that the condition (2.6) and (3.2) hold. Then, for $m = 1$ and π_i^{NOPL} in (3.1), we have*

$$\sum_{i=1}^{p}\pi_i^{NOPL}\|\frac{\widetilde{e}_i}{p}\|_2^3 = O_p(1), \tag{E.1}$$

where $\widetilde{e}_i = (\frac{A_iA_i^T}{\pi_i^{NOPL}} + \lambda I)\hat{z}^* - \widetilde{y}$ with $\widetilde{y} = \lambda y$ and $\hat{z}^*$ being as in (1.3).

Proof. Similar to the proof of Lemma A.1, based on (2.6), (3.1), (3.2), (A.2), and (A.3), we have

$$\begin{aligned}\sum_{i=1}^{p}\pi_i^{NOPL}\|\frac{\widetilde{e}_i}{p}\|_2^3 \le&\frac{\|\widetilde{y}\|_2^3\sigma_n^3(AA^T + \lambda I)}{p^3}(\sum_{i=1}^{p}\frac{\|A_iA_i^T\|_2^3}{(\pi_i^{NOPL})^2} + 3\lambda\sum_{i=1}^{p}\frac{\|A_iA_i^T\|_2^2}{\pi_i^{NOPL}}\\ &+ 3\lambda^2\sum_{i=1}^{p}\|A_iA_i^T\|_2 + \lambda^3) + \frac{\|\widetilde{y}\|_2^3}{p^3}\\ &+ 3\frac{\|\widetilde{y}\|_2^3\sigma_n^2(AA^T + \lambda I)}{p^3}(\sum_{i=1}^{p}\frac{\|A_iA_i^T\|_2^2}{\pi_i^{NOPL}}\\ &+ 2\lambda\sum_{i=1}^{p}\|A_iA_i^T\|_2 + \lambda^2)\\ &+ 3\frac{\|\widetilde{y}\|_2^3\sigma_n(AA^T + \lambda I)}{p^3}(\sum_{i=1}^{p}\|A_iA_i^T\|_2 + \lambda)\\ =&\frac{\|\widetilde{y}\|_2^3\sigma_n^3(AA^T + \lambda I)}{p^3}[\sum_{i=1}^{p}\frac{\|A_iA_i^T\|_2^3}{(|\widetilde{\beta}_{(i)}|\|A_i\|_2)^2}(\sum_{i=1}^{p}|\widetilde{\beta}_{(i)}|\|A_i\|_2)^2\\ &+ 3\lambda\sum_{i=1}^{p}\frac{\|A_iA_i^T\|_2^2}{|\widetilde{\beta}_{(i)}|\|A_i\|_2}\sum_{i=1}^{p}|\widetilde{\beta}_{(i)}|\|A_i\|_2 + 3\lambda^2\sum_{i=1}^{p}\|A_iA_i^T\|_2\\ &+ \lambda^3] + \frac{\|\widetilde{y}\|_2^3}{p^3} + 3\frac{\|\widetilde{y}\|_2^3\sigma_n^2(AA^T + \lambda I)}{p^3}(\sum_{i=1}^{p}\frac{\|A_iA_i^T\|_2^2}{|\widetilde{\beta}_{(i)}|\|A_i\|_2}\\ &\sum_{i=1}^{p}|\widetilde{\beta}_{(i)}|\|A_i\|_2 + 2\lambda\sum_{i=1}^{p}\|A_iA_i^T\|_2 + \lambda^2)\\ &+ 3\frac{\|\widetilde{y}\|_2^3\sigma_n(AA^T + \lambda I)}{p^3}(\sum_{i=1}^{p}\|A_iA_i^T\|_2 + \lambda) \quad \text{by (3.1)}\\ \le&\frac{\|\widetilde{y}\|_2^3\sigma_n^3(AA^T + \lambda I)}{p^3}[\frac{N_2^2}{N_1^2}(\sum_{i=1}^{p}\|A_i\|_2^2)^3\\ &+ 3\lambda\frac{N_2}{N_1}(\sum_{i=1}^{p}\|A_i\|_2^2)^2\\ &+ 3\lambda^2\sum_{i=1}^{p}\|A_i\|_2^2 + \lambda^3] + \frac{\|\widetilde{y}\|_2^3}{p^3}\\ &+ 3\frac{\|\widetilde{y}\|_2^3\sigma_n^2(AA^T + \lambda I)}{p^3}[\frac{N_2}{N_1}(\sum_{i=1}^{p}\|A_i\|_2^2)^2\\ &+ 2\lambda\sum_{i=1}^{p}\|A_i\|_2^2 + \lambda^2]\\ &+ 3\frac{\|\widetilde{y}\|_2^3\sigma_n(AA^T + \lambda I)}{p^3}(\sum_{i=1}^{p}\|A_i\|_2^2 + \lambda) \quad \text{by (3.2)}\\ =&O_p(1), \quad \text{by (2.6) and (A.3)}\end{aligned}$$

where the first inequality is gained by replacing π_i in (A.2) with π_i^{NOPL}. Then, (E.1) is obtained. □

Lemma E.2. *To the assumption of Lemma E.1, add that the condition (2.7) holds. Then, for $m = 1$ and π_i^{NOPL} in (3.1), conditional on $\mathcal{F}_n$*

and $\widetilde{\beta}$ in probability, we have

$$\frac{\widetilde{M}_A - M_A}{p} = O_{p|\mathcal{F}_n}(r^{-1/2}), \tag{E.2}$$

$$\frac{\widetilde{e}^*}{p} = O_{p|\mathcal{F}_n}(r^{-1/2}), \tag{E.3}$$

where $M_A = AA^T + \lambda I$, $\widetilde{M}_A = A\widetilde{S}\widetilde{S}^T A^T + \lambda I$ with $\widetilde{S}$ constructed by π_i^{NOPL}, and $\widetilde{e}^* = (\widetilde{M}_A \hat{z}^* - \widetilde{y})$ with $\widetilde{y} = \lambda y$ and $\hat{z}^*$ being as in (1.3).

Proof. The proof can be completed similar to the proof of Lemma A.2. We only need to replace π_i with π_i^{NOPL}, and note that

$$\begin{aligned}
\frac{1}{p^2}\sum_{i=1}^{p}\frac{A_iA_i^TA_iA_i^T}{\pi_i^{NOPL}} &= \frac{1}{p^2}\Big(\sum_{i=1}^{p}\frac{A_iA_i^TA_iA_i^T}{|\widetilde{\beta}_{(i)}|\,\|A_i\|_2}\Big)\Big(\sum_{i=1}^{p}|\widetilde{\beta}_{(i)}|\,\|A_i\|_2\Big)\\
&\le \frac{N_2}{N_1p^2}\Big(\sum_{i=1}^{p}\frac{A_iA_i^TA_iA_i^T}{\|A_i\|_2^2}\Big)\Big(\sum_{i=1}^{p}\|A_i\|_2^2\Big)\\
&= \frac{N_2}{N_1p^2}\Big(\sum_{i=1}^{p}A_iA_i^T\Big)\Big(\sum_{i=1}^{p}\|A_i\|_2^2\Big)\\
&= O_p(1), \quad \text{by (2.6) and (2.7)}
\end{aligned} \tag{E.4}$$

$$\begin{aligned}
\mathrm{E}(\widetilde{M}_A \mid A) &= \mathrm{E}_{\widetilde{\beta}}[\mathrm{E}(\widetilde{M}_A \mid A, \widetilde{\beta})],\\
\mathrm{Var}[\frac{(\widetilde{M}_A - M_A)\ell}{p} \mid A] &= \mathrm{E}_{\widetilde{\beta}}\{\mathrm{Var}[\frac{(\widetilde{M}_A - M_A)\ell}{p} \mid A, \widetilde{\beta}]\},\\
\mathrm{E}(\widetilde{e}_{i_t} \mid \mathcal{F}_n) &= \mathrm{E}_{\widetilde{\beta}}[\mathrm{E}(\widetilde{e}_{i_t} \mid \mathcal{F}_n, \widetilde{\beta})],\\
\mathrm{Var}(\frac{\ell^T\widetilde{e}^*}{p} \mid \mathcal{F}_n) &= \mathrm{E}_{\widetilde{\beta}}[\mathrm{Var}(\frac{\ell^T\widetilde{e}^*}{p} \mid \mathcal{F}_n, \widetilde{\beta})],
\end{aligned}$$

where $\mathrm{E}_{\widetilde{\beta}}$ denotes the expectation on $\widetilde{\beta}$. □

Remark E.1. The results (E.2) and (E.3) still hold when $\widetilde{M}_A = AS^*S^{*T}A^T + \lambda I$ with $S^* \in \mathbb{R}^{p\times r_0}$ formed by π_i^{COL}.

Corollary E.1. *For $S^* \in \mathbb{R}^{p\times r_0}$ formed by π_i^{COL}, $\widetilde{z} = (AS^*S^{*T}A^T + \lambda I)^{-1}\widetilde{y}$ constructed by Algorithm 2 satisfies*

$$\|\widetilde{z} - \hat{z}^*\|_2 = O_{p|\mathcal{F}_n}(r_0^{-1/2}). \tag{E.5}$$

Proof. Similar to (A.8), considering (2.7) and Remark E.1, we can get

$$\widetilde{z} - \hat{z}^* = -(\frac{M_A}{p})^{-1}\frac{\widetilde{e}^*}{p} + (\frac{\widetilde{M}_A}{p})^{-1}(\frac{\widetilde{M}_A - M_A}{p})(\frac{M_A}{p})^{-1}\frac{\widetilde{e}^*}{p} = O_{p|\mathcal{F}_n}(r_0^{-1/2}),$$

which suggests that (E.5) holds. □

Proof of Theorem 3.1. Similar to the proof of Theorem 2.1 , noting (2.6), (2.7), (E.4), and Lemmas E.1 and E.2, and replacing π_i and e_{i_t} in the proof of Theorem 2.1 with π_i^{NOPL} and $\widetilde{e}_{i_t}$, respectively, we first get

$$\hat{z}_1 - \hat{z}^* = -(\frac{M_A}{p})^{-1}\frac{\widetilde{e}^*}{p} + O_{p|\mathcal{F}_n}(r^{-1}), \tag{E.6}$$

$$(\frac{\widetilde{V}_c}{r})^{-1/2}(\frac{\widetilde{e}_*}{p}) \xrightarrow{L} N(0, I),$$

where

$$\hat{z}_1 = (A\widetilde{S}\widetilde{S}^TA^T + \lambda I)^{-1}\widetilde{y} = \widetilde{M}_A^{-1}\widetilde{y},$$

$$\widetilde{V}_c = \sum_{i=1}^{p}\frac{A_iA_i^T\hat{z}^*\hat{z}^{*T}A_iA_i^T}{p^2\pi_i^{NOPL}} = O_p(1).$$

To get (3.3), in the following, we need to further prove

$$V_{OPL}^{-1/2}(\hat{z}_1 - \hat{z}^*) = -V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}(\frac{\widetilde{V}_c}{r})^{1/2}(\frac{\widetilde{V}_c}{r})^{-1/2}\frac{\widetilde{e}^*}{p} + O_{p|\mathcal{F}_n}(r^{-1/2}), \tag{E.7}$$

where $V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}(\frac{\widetilde{V}_c}{r})^{1/2}$ satisfies

$$V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}(\frac{\widetilde{V}_c}{r})^{1/2}[V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}(\frac{\widetilde{V}_c}{r})^{1/2}]^T = I + O_{p|\mathcal{F}_n}(r_0^{-1/2}). \tag{E.8}$$

Considering (2.6), (2.7), (2.10) and (3.2), we first obtain

$$\begin{aligned}
\frac{1}{p^2}\sum_{i=1}^{p}\frac{A_iA_i^TA_iA_i^T}{\pi_i^{OPL}} &= \frac{1}{p^2}\Big(\sum_{i=1}^{p}\frac{A_iA_i^TA_iA_i^T}{|\hat{\beta}_{rls(i)}|\,\|A_i\|_2}\Big)\Big(\sum_{i=1}^{p}|\hat{\beta}_{rls(i)}|\,\|A_i\|_2\Big) \quad \text{by (2.10)}\\
&\le \frac{N_4}{N_3p^2}\Big(\sum_{i=1}^{p}\frac{A_iA_i^TA_iA_i^T}{\|A_i\|_2^2}\Big)\Big(\sum_{i=1}^{p}\|A_i\|_2^2\Big) \quad \text{by (3.2)}\\
&= \frac{N_4}{N_3p^2}\Big(\sum_{i=1}^{p}A_iA_i^T\Big)\Big(\sum_{i=1}^{p}\|A_i\|_2^2\Big)\\
&= O_p(1), \quad \text{by (2.6) and (2.7)}
\end{aligned}$$

which indicates

$$V_{cOPL} = \sum_{i=1}^{p}\frac{A_iA_i^T\hat{z}^*\hat{z}^{*T}A_iA_i^T}{p^2\pi_i^{OPL}} = O_p(1). \tag{E.9}$$

From (2.7) and (E.9), it is evident to get

$$V_{OPL} = (\frac{M_A}{p})^{-1}\frac{V_{cOPL}}{r}(\frac{M_A}{p})^{-1} = O_p(r^{-1}), \tag{E.10}$$

which combined with (E.6) suggests that (E.7) holds, that is,

$$\begin{aligned}
V_{OPL}^{-1/2}(\hat{z}_1 - \hat{z}^*) &= -V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}\frac{\widetilde{e}^*}{p} + O_{p|\mathcal{F}_n}(r^{-1/2})\\
&= -V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}(\frac{\widetilde{V}_c}{r})^{1/2}(\frac{\widetilde{V}_c}{r})^{-1/2}\frac{\widetilde{e}^*}{p} + O_{p|\mathcal{F}_n}(r^{-1/2}).
\end{aligned}$$

Now, we need to demonstrate that (E.8) also holds. Evidently, it suffices to show that

$$V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}\frac{\widetilde{V}_c - V_{cOPL}}{r}(\frac{M_A}{p})^{-1}V_{OPL}^{-1/2} = O_{p|\mathcal{F}_n}(r_0^{-1/2}), \tag{E.11}$$

because

$$\begin{aligned}
&V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}(\frac{\widetilde{V}_c}{r})^{1/2}[V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}(\frac{\widetilde{V}_c}{r})^{1/2}]^T\\
&= V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}\frac{\widetilde{V}_c}{r}(\frac{M_A}{p})^{-1}V_{OPL}^{-1/2}\\
&= V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}\frac{V_{cOPL}}{r}(\frac{M_A}{p})^{-1}V_{OPL}^{-1/2}\\
&+ V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}\frac{\widetilde{V}_c - V_{cOPL}}{r}(\frac{M_A}{p})^{-1}V_{OPL}^{-1/2}\\
&= I + V_{OPL}^{-1/2}(\frac{M_A}{p})^{-1}\frac{\widetilde{V}_c - V_{cOPL}}{r}(\frac{M_A}{p})^{-1}V_{OPL}^{-1/2}.
\end{aligned}$$

Noting

$$\widetilde{V}_c = \underbrace{[\frac{1}{p}\Big(\sum_{i=1}^{p}\frac{A_iA_i^T\hat{z}^*\hat{z}^{*T}A_iA_i^T}{|\widetilde{\beta}_{(i)}|\,\|A_i\|_2}\Big)]}_{\Phi_1}\underbrace{[\frac{1}{p}\Big(\sum_{i=1}^{p}|\widetilde{\beta}_{(i)}|\,\|A_i\|_2\Big)]}_{\Phi_2},$$

$$V_{cOPL} = \underbrace{[\frac{1}{p}\Big(\sum_{i=1}^{p}\frac{A_iA_i^T\hat{z}^*\hat{z}^{*T}A_iA_i^T}{|\hat{\beta}_{rls(i)}|\,\|A_i\|_2}\Big)]}_{\Phi_3}\underbrace{[\frac{1}{p}\Big(\sum_{i=1}^{p}|\hat{\beta}_{rls(i)}|\,\|A_i\|_2\Big)]}_{\Phi_4},$$

and the basic triangle inequality, we gain

$$\begin{aligned}
\|\widetilde{V}_c - V_{cOPL}\|_2 &= \|\Phi_1\Phi_2 - \Phi_3\Phi_4\|_2\\
&\le \|\Phi_1 - \Phi_3\|_2\|\Phi_2\|_2 + \|\Phi_2 - \Phi_4\|_2\|\Phi_3\|_2.
\end{aligned}$$

Following (2.6), (3.2), (A.3), and (E.5), it is evident to gain

$$\begin{aligned}
\|\Phi_1 - \Phi_3\|_2 &\le \|\frac{1}{p}\sum_{i=1}^{p}\frac{A_iA_i^T\hat{z}^*\hat{z}^{*T}A_iA_i^T}{\|A_i\|_2}(\frac{1}{|\widetilde{\beta}_{(i)}|} - \frac{1}{|\hat{\beta}_{rls(i)}|})\|_2\\
&\le \frac{1}{p}\sum_{i=1}^{p}\frac{\lambda^2\hat{\beta}_{rls(i)}^2\|A_i\|_2^2}{\|A_i\|_2}(\frac{|\widetilde{\beta}_{(i)} - \hat{\beta}_{rls(i)}|}{|\hat{\beta}_{rls(i)}|\,|\widetilde{\beta}_{(i)}|})\\
&\le \frac{\lambda N_4}{pN_1}\sum_{i=1}^{p}\frac{\|A_i\|_2^3}{\|A_i\|_2^2}(\|A_i\|_2\|\widetilde{z} - \hat{z}^*\|_2) \quad \text{by (3.2)}\\
&= \|\widetilde{z} - \hat{z}^*\|_2\sum_{i=1}^{p}\frac{\lambda N_4\|A_i\|_2^2}{pN_1} = O_{p|\mathcal{F}_n}(r_0^{-1/2}), \quad \text{by (2.6) and (E.5)}
\end{aligned}$$

$$\|\Phi_2\|_2 \le \frac{N_2\|y\|_2}{p}\sum_{i=1}^{p}\|A_i\|_2^2 = O_p(1). \quad \text{by (2.6), (3.2), and (A.3)}$$

Similarly, we have $\|\Phi_2 - \Phi_4\|_2 = O_{p|\mathcal{F}_n}(r_0^{-1/2})$ and $\|\Phi_3\|_2 = O_p(1)$. Therefore, we get

$$\|\widetilde{V} - V_{cOPL}\|_2 = O_{p|\mathcal{F}_n}(r_0^{-1/2}),$$

which combined with (2.7) and (E.10) yields (E.11). Putting the above discussions and the Slutsky's Theorem together, the result (3.3) follows.

Appendix F. Proof of Theorem 3.2

Before providing the proof of Theorem 3.2, we first present a lemma.

Lemma F.1. *To the assumption of Lemma C.2, add that (3.4) holds and* $r \ge \frac{32s_4c_2\rho}{3s_3c_1\epsilon^2}\ln(\frac{4\rho}{\delta})$ *with* $\epsilon, \delta \in (0,1)$. *Then, for any* ϵ, $\hat{w}_t$ *obtained from the* t*th iteration of Algorithm 2 satisfies*

$$\|\frac{A^T\hat{w}_t}{\lambda} - \frac{A^Tw_t^*}{\lambda}\|_2 \le \epsilon\|\frac{A^Tw_t^*}{\lambda}\|_2, \tag{F.1}$$

where w_t^* *is the solution of*

$$\min_{w_t}\frac{1}{2\lambda}\|A^Tw_t\|_2^2 + \frac{1}{2}\|w_t\|_2^2 - w_t^Tb_t.$$

Proof. The proof can be completed along the line of the proof of Theorem 2.3. Particularly, in this case, Lemma C.2 still holds for $S = \widetilde{S}$, where $\widetilde{S}$ is formed by π_i^{NOPL}. □

Proof of Theorem 3.2. At the tth iteration, following the discussion in Remark 3.1 and (F.1), and setting

$$\triangle_t^* = \frac{A^Tw_t^*}{\lambda} = \frac{A^T\hat{z}^*}{\lambda} - \frac{A^T\hat{z}_{t-1}}{\lambda}$$

and $\hat{\triangle}_t = \frac{A^T\hat{w}_t}{\lambda}$ as the estimator of $\triangle_t^*$, we can have

$$\begin{aligned}\|\hat{\triangle}_t - \triangle_t^*\|_2 &\le \epsilon\|\triangle_t^*\|_2 \quad \text{by (F.1)}\\ &= \epsilon\|\frac{A^T\hat{z}^*}{\lambda} - \frac{A^T\hat{z}_{t-1}}{\lambda}\|_2\\ &= \epsilon\|\frac{A^T(\hat{z}_{t-2} + w_{t-1}^*)}{\lambda} - \frac{A^T(\hat{z}_{t-2} + \hat{w}_{t-1})}{\lambda}\|_2\\ &\le \epsilon\|\hat{\triangle}_{t-1} - \triangle_{t-1}^*\|_2 \le \epsilon^2\|\triangle_{t-1}^*\|_2.\end{aligned}$$

As a result,

$$\begin{aligned}\|\hat{\triangle}_m - \triangle_m^*\|_2 &\le \epsilon\|\hat{\triangle}_{m-1} - \triangle_{m-1}^*\|_2 \le \epsilon^m\|\triangle_1^*\|_2\\ &\le \epsilon^m\|\frac{A^T\hat{z}^*}{\lambda} - \frac{A^T\hat{z}_0}{\lambda}\|_2\\ &= \epsilon^m\|\frac{A^T\hat{z}^*}{\lambda}\|_2 = \epsilon^m\|\hat{\beta}_{rls}\|_2.\end{aligned}$$

Considering that $\hat{\beta}_m - \hat{\beta}_{rls} = \hat{\triangle}_m - \triangle_m^*$, the conclusion is arrived.

References

[1] A.E. Hoerl, R.W. Kennard, Ridge regression: biased estimation for nonorthogonal problems, Technometrics 12 (1) (1970) 55–67, http://dx.doi.org/10.1080/00401706.1970.10488634.

[2] A.N. Tihonov, Solution of incorrectly formulated problems and the regularization method, Soviet Math. Dokl. 5 (1963) 1035–1038.

[3] C. Saunders, A. Gammerman, V. Vovk, Ridge regression learning algorithm in dual variables, in: Proceedings of the 15th International Conference on Machine Learning, 1998, pp. 515–521.

[4] Y. Lu, P.S. Dhillon, D.P. Foster, L.H. Ungar, Faster ridge regression via the subsampled randomized hadamard transform, in: Proceedings of the 26th Annual Conference on Advances in Neural Information Processing Systems, Vol. 26, 2013, pp. 369–377.

[5] S. Chen, Y. Liu, M.R. Lyu, I. King, S. Zhang, Fast relative-error approximation algorithm for ridge regression, in: Proceedings of the 31st Conference on Uncertainty in Artificial Intelligence, 2015, pp. 201–210.

[6] H. Avron, K.L. Clarkson, D.P. Woodruff, Sharper bounds for regularized data fitting, in: Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, Vol. 81, 2017, pp. 27 : 1–27 : 22.

[7] J. Wang, J.D. Lee, M. Mahdavi, M. Kolar, N. Srebro, Sketching meets random projection in the dual: a provable recovery algorithm for big and high-dimensional data, Electron. J. Stat. 11 (2) (2017) 4896–4944, http://dx.doi.org/10.1214/17-EJS1334SI.

[8] A. Chowdhury, J. Yang, P. Drineas, An iterative, sketching-based framework for ridge regression, in: Proceedings of the 35th International Conference on Machine Learning, Vol. 80, 2018, pp. 989–998.

[9] J. Lacotte, M. Pilanci, Adaptive and oblivious randomized subspace methods for high-dimensional optimization: sharp analysis and lower bounds, 2020, arXiv preprint arXiv:2012.07054.

[10] L. Zhang, M. Mahdavi, R. Jin, T. Yang, S. Zhu, Recovering the optimal solution by dual random projection, in: Proceedings of the 26th Annual Conference on Learning Theory, Vol. 30, 2013, pp. 135–157.

[11] L. Zhang, M. Mahdavi, R. Jin, T. Yang, S. Zhu, Random projections for classification: a recovery approach, IEEE Trans. Inform. Theory 60 (11) (2014) 7300–7316, http://dx.doi.org/10.1109/TIT.2014.2359204.

[12] O.-A. Maillard, R. Munos, Compressed least-squares regression, in: Proceedings of Advances in Neural Information Processing Systems, Vancouver, Canada, 2009, pp. 1213–1221.

[13] M.M. Fard, Y. Grinberg, J. Pineau, D. Precup, Compressed least-squares regression on sparse spaces, in: Proceedings of the 26th AAAI Conference on Artificial Intelligence, Vol. 26, No. 1, 2012, pp. 1054–1060.

[14] A. Kabán, A new look at compressed ordinary least squares, in: 2013 IEEE 13th International Conference on Data Mining Workshops, 2013, pp. 482–488.

[15] A. Kabán, New bounds on compressive linear least squares regression, in: Proceedings of the 17th International Conference on Artificial Intelligence and Statistics, Vol. 33, 2014, pp. 448–456.

[16] G.-A. Thanei, C. Heinze, N. Meinshausen, Random projections for large-scale regression, in: Big and Complex Data Analysis, 2017, pp. 51–68.

[17] M. Slawski, Compressed least squares regression revisited, in: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Vol. 54, 2017, pp. 1207–1215.

[18] M. Slawski, On principal components regression, random projections, and column subsampling, Electron. J. Statist. 12 (2) (2018) 3673–3712, http://dx.doi.org/10.1214/18-EJS1486.

[19] L. Mor-Yosef, H. Avron, Sketching for principal component regression, SIAM J. Matrix Anal. Appl. 40 (2) (2019) 454–485, http://dx.doi.org/10.1137/18M1188860.

[20] P. Drineas, M. Magdon-Ismail, M.W. Mahoney, D.P. Woodruff, Fast approximation of matrix coherence and statistical leverage, J. Mach. Learn. Res. 13 (1) (2012) 3475–3506.

[21] R. Zhu, P. Ma, M.W. Mahoney, B. Yu, Optimal subsampling approaches for large sample linear regression, 2015, arXiv preprint arXiv:1509.05111.

[22] H. Wang, R. Zhu, P. Ma, Optimal subsampling for large sample logistic regression, J. Amer. Statist. Assoc. 113 (522) (2018) 829–844, http://dx.doi.org/10.1080/01621459.2017.1292914.

[23] P. Ma, X. Zhang, X. Xing, J. Ma, M. Mahoney, Asymptotic analysis of sampling estimators for randomized numerical linear algebra algorithms, in: Proceedings of the 23nd International Conference on Artificial Intelligence and Statistics, Vol. 108, 2020, pp. 1026–1035.

[24] H. Zhang, H. Wang, Distributed subdata selection for big data via sampling-based approach, Comput. Statist. Data Anal. 153 (2021) 107072, http://dx.doi.org/10.1016/j.csda.2020.107072.

[25] Y. Yao, H. Wang, Optimal subsampling for softmax regression, Statist. Papers 60 (2) (2019) 585–599, http://dx.doi.org/10.1007/s00362-018-01068-6.

[26] M. Ai, J. Yu, H. Zhang, H. Wang, Optimal subsampling algorithms for big data regressions, Statist. Sinica 31 (2021) 749–772.

[27] J. Yu, H. Wang, M. Ai, H. Zhang, Optimal distributed subsampling for maximum quasi-likelihood estimators with massive data, J. Amer. Statist. Assoc. 117 (537) (2022) 265–276, http://dx.doi.org/10.1080/01621459.2020.1773832.

[28] H. Wang, Y. Ma, Optimal subsampling for quantile regression in big data, Biometrika 108 (1) (2021) 99–112, http://dx.doi.org/10.1093/biomet/asaa043.

[29] Y. Chen, N. Zhang, Optimal subsampling for large sample ridge regression, 2022, arXiv preprint arXiv:2204.04776.

[30] A. van der Vaart, Asymptotic Statistics, Cambridge University Press, London, 1998.

[31] F. Pukelsheim, Optimal Design of Experiments, Wiley, New York, 1993.

[32] R.A. Horn, C.R. Johnson, Matrix Analysis, Cambridge University Press, 2012.

[33] V.V. Buldygin, Y.V. Kozachenko, Sub-Gaussian random variables, Ukrainian Math. J. 32 (6) (1980) 483–489, http://dx.doi.org/10.1007/BF01087176.

[34] J. Fan, J. Lv, Sure independence screening for ultrahigh dimensional feature space, J. R. Stat. Soc. Ser. B Stat. Methodol. 70 (5) (2008) 849–911, http://dx.doi.org/10.1111/j.1467-9868.2008.00674.x.

[35] M. Pilanci, M.J. Wainwright, Newton sketch: a near linear-time optimization algorithm with linear-quadratic convergence, SIAM J. Optim. 27 (1) (2017) 205–245, http://dx.doi.org/10.1137/15M1021106.

[36] A. Quarteroni, A. Valli, Numerical Approximation of Partial Differential Equations, Springer-Verlag, Berlin, 2008.

[37] T.S. Ferguson, A Course in Large Sample Theory, Chapman and Hall, London, 1996.