



# Core-elements for large-scale least squares estimation

Mengyu Li<sup>1</sup> · Jun Yu<sup>2</sup> · Tao Li<sup>1</sup> · Cheng Meng<sup>1</sup>

Received: 11 May 2024 / Accepted: 15 September 2024

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024

## Abstract

The coresets approach, also called subsampling or subset selection, aims to select a subsample as a surrogate for the observed sample and has found extensive application in large-scale data analysis. Existing coresets methods construct the subsample using a subset of rows from the predictor matrix. Such methods can be significantly inefficient when the predictor matrix is sparse or numerically sparse. To overcome this limitation, we develop a novel element-wise subset selection approach, called core-elements, for large-scale least squares estimation. We provide a deterministic algorithm to construct the core-elements estimator, only requiring an  $O(\text{nnz}(X) + rp^2)$  computational cost, where  $X$  is an  $n \times p$  predictor matrix,  $r$  is the number of elements selected from each column of  $X$ , and  $\text{nnz}(\cdot)$  denotes the number of non-zero elements. Theoretically, we show that the proposed estimator is unbiased and approximately minimizes an upper bound of the estimation variance. We also provide an approximation guarantee by deriving a coresets-like finite sample bound for the proposed estimator. To handle potential outliers in the data, we further combine core-elements with the median-of-means procedure, resulting in an efficient and robust estimator with theoretical consistency guarantees. Numerical studies on various synthetic and real-world datasets demonstrate the proposed method's superior performance compared to mainstream competitors.

**Keywords** Coresets · Linear model · Sparse matrix · Subset selection

## 1 Introduction

Sparse matrices are matrices in which most of the elements are zero. Such matrices are common in various areas, including medical research, bioinformatics, privacy-preserving analysis, and distributed computing (Davis and Hu 2011; Nguyen et al. 2023; Liu et al. 2021; Kairouz et al. 2021). In these areas, data are usually of high sparsity due to technical noises, data privacy concerns, and transmission cost,

among others (Konečný et al. 2016; Andrews et al. 2021). One example is the single-cell RNA-sequencing (scRNA-seq) data containing information about the gene expression level of single cells. Owing to technical noises and intrinsic biological variability, scRNA-seq data expressed in count matrices always possess significant sparsity, known as the zero-inflation phenomenon (Nguyen et al. 2023). Another example is the word occurrence matrix, whose elements are calculated by multiplying two metrics, i.e., how many times a word appears in a document, and the inverse document frequency of the word across a set of documents. Such matrices are also highly sparse, especially for a short document and a large language model that contains millions of words (Qaiser and Ali 2018).

In reality, many sparse matrices also exist due to missing values, i.e., the elements are not fully observed. Numerous methods have been developed to deal with the missing values in such cases, and most of these methods aim to fill the sparse matrix with some estimated non-zero values (Cai et al. 2010; Van Buuren and Groothuis-Oudshoorn 2011; Hastie et al. 2015; Muzellec et al. 2020). In this paper, however, we are less concerned with missing values and instead focus on the case that the sparse matrix itself is fully observed.

Jun Yu: Joint first author.

✉ Cheng Meng  
chengmeng@ruc.edu.cn

Mengyu Li  
limengyu516@ruc.edu.cn

Jun Yu  
yujunbeta@bit.edu.cn

Tao Li  
2019000153lt@ruc.edu.cn

<sup>1</sup> Center for Applied Statistics, Institute of Statistics and Big Data, Renmin University of China, Beijing 100872, China

<sup>2</sup> School of Mathematics and Statistics, Beijing Institute of Technology, Beijing 100081, China

We consider large-scale data analysis where the predictor matrix is highly sparse. One widely-used technique for large-scale data analysis is the coresets method, also called subsampling or subset selection. These methods select a subsample as a surrogate for the observed sample. Recently, such methods have been used pervasively in data reduction, measurement-constrained analysis, and active learning (Li and Meng 2021; Meng et al. 2021; Settles 2012). Various coresets methods have been proposed for linear regression (Dasgupta et al. 2009; Boutsidis et al. 2013; Ma and Sun 2015; Meng et al. 2017; Dereziński et al. 2018; Ma et al. 2020; Wang et al. 2021), generalized linear regression (Wang et al. 2018; Ai et al. 2020, 2021; Yu et al. 2022), streaming time series (Xie et al. 2019; Li et al. 2019), large-scale matrix approximation (Wang and Zhang 2013; Alaoui and Mahoney 2015; Wang et al. 2019), nonparametric regression (Ma et al. 2015; Meng et al. 2020; Sun et al. 2021; Meng et al. 2022; Dai et al. 2023), among others. Another avenue for handling large-scale data is data averaging (Wang et al. 2023), which is beyond the scope of this paper.

Despite the wide application, most existing coresets methods mainly focus on dense predictor matrices, and may be inefficient when the predictor matrix is of high sparsity. In particular, most of these methods construct the subsample using certain rows from the observed sample. When the observed predictor matrix is sparse, the selected subsample matrix also tends to be sparse for the subsampling methods that preserve the empirical distribution of the full data (Mak and Joseph 2018; Joseph and Vakayil 2022; Vakayil and Joseph 2022). Such a subsample thus may lead to inefficient results, since the selected zero-valued elements have almost no impact on the down-streaming analysis, such as model estimation, prediction, and inference. Another category of subsampling approaches is designed for prediction purposes (Joseph and Mak 2021; Chang 2022; Dai et al. 2023), whose resulting subsample may not be statistically similar to the full data, thereby potentially overcoming the inefficiency discussed above. Nevertheless, this class of methods still encounters the probability of selecting a sparse subsample matrix when the full predictor matrix has an extremely high sparsity. Consequently, more efficient statistical tools suitable for sparse matrices are still meager.

In this paper, we bridge this gap by developing a novel element-wise subset selection method, called core-elements, for large-scale least squares estimation. Different from existing coresets methods that aim to select  $r$  rows from the predictor matrix  $X \in \mathbb{R}^{n \times p}$  ( $n \gg p$ ), we aim to construct a sparser subdata matrix  $X^* \in \mathbb{R}^{n \times p}$  by keeping  $rp$  elements of  $X$  and zeroing out the remaining elements. Loosely speaking, our approach generalizes the existing coresets methods by getting rid of the requirement that the selected  $rp$  elements have to be located in  $r$  rows. Our major contributions are three-fold as follows.

- (1) We provide a deterministic algorithm to construct the core-elements estimator for linear regression. Utilizing such an estimator, we can approximate the least squares estimation within  $O(\text{nnz}(X) + rp^2)$  computational time, where  $\text{nnz}(\cdot)$  denotes the number of non-zero elements. Theoretical analysis demonstrates that our proposed estimator is unbiased and approximately minimizes an upper bound on estimation variance.
- (2) We establish a coresets-like finite sample bound for the proposed estimator with approximation guarantees. In particular, we show that to achieve an  $(1 + \epsilon)$ -relative error, the proposed estimator requires a subdata matrix  $X^*$  for the predictor matrix  $X$  such that the ratio  $\|X - X^*\|_2 / \|X\|_2$  is  $O(\epsilon^{1/2})$ . Intuitively, such a result indicates that when  $X$  gets (numerically) sparser, fewer elements are required in  $X^*$  to achieve the  $(1 + \epsilon)$ -relative error.
- (3) To handle potential outliers in the data, we develop a robust variant of the core-elements method by integrating it with the widely adopted median-of-means procedure (Lugosi and Mendelson 2019; Lecué and Lerasle 2020; Huang and Lederer 2023). We further propose an algorithm to construct the robust estimator within  $O(\text{nnz}(X) + rp^2)$  time. Theoretically, we show that the robust estimator is consistent with the true coefficient under certain regularity conditions.

To evaluate the empirical performance and computational efficiency of the proposed strategy, we compare it with mainstream competitors through extensive synthetic and real-world datasets, including uncorrupted and corrupted data with dense or sparse predictor matrices. Interestingly, although its primary design aims at sparse matrices, numerical findings reveal that the proposed estimator significantly outperforms its competitors regarding both estimation accuracy and CPU time, even when the predictor matrix is dense.

The remainder of this paper is organized as follows. We start in Sect. 2 by introducing the linear model and the state-of-the-art subsampling methods. In Sect. 3, we develop the core-elements estimator and present its theoretical properties. The robust version of the core-elements estimator is introduced in Sect. 4. We examine the performance of the proposed estimators through extensive simulations and a real-world example in Sects. 5 and 6, respectively. Technical proofs, additional experimental results, and implementation code are provided in supplementary materials.

## 2 Background

We adopt the common convention of using uppercase letters for matrices and lowercase letters for vectors or scalars. We denote the Euclidean norm and  $\ell_1$  norm of a vector  $x$  as  $\|x\|$  and  $\|x\|_1$ , respectively. For a matrix  $X$ , we represent its

**Table 1** Comparison of mainstream subsampling methods on computational cost, method type, and sampling type

Method	Computational cost*	Method type	Sampling type
Full sample	$O(np^2)$	–	–
Uniform subsampling	$O(rp^2)$	Randomized	Row-wise
Doubly sketching (Hou-Liu and Browne 2023)	$O(rp^2)$	Randomized	Row-wise
Leverage score subsampling (Ma and Sun 2015)	$O(np^2 + rp^2)^\dagger$	Randomized	Row-wise
IBOSS (Wang et al. 2019)	$O(np + rp^2)$	Deterministic	Row-wise
OSS (Wang et al. 2021)	$O(np \log r + rp^2)$	Deterministic	Row-wise
D-optimal subsampling (Reuter and Schwabe 2024)	$O(np^2 + rp^2)^\ddagger$	Deterministic	Row-wise
Core-elements (proposed)	$O(\text{nnz}(X) + rp^2)$	Deterministic	Element-wise

\* Here, each method subsamples  $r$  rows or  $s = rp$  elements from an  $n \times p$  predictor matrix

† The  $O(np^2)$  component can be reduced to  $O(np \log n)$  by involving some random projection-based approximation methods (Drineas et al. 2012)

‡ The  $O(np^2)$  component can be reduced to  $O(np)$  by using a simplified approximation that ignores non-diagonal elements of the covariance matrix

spectral norm (i.e., largest singular value) as  $\|X\|_2$  and its Frobenius norm as  $\|X\|_F$ . The condition number of  $X$ , i.e., the ratio of its largest and smallest singular values, is denoted as  $\kappa(X)$ . Additionally, we use the notations  $E(\cdot)$ ,  $\text{pr}(\cdot)$ , and  $\text{tr}(\cdot)$  for mathematical expectation, probability measure, and trace, respectively.

### 2.1 Subsampling methods for least squares estimation

Consider the linear regression model,

$$y_i = x_i^\top \beta + \varepsilon_i, \quad i = 1, \dots, n. \tag{1}$$

Here  $\{y_i\}_{i=1}^n \subset \mathbb{R}$  are the responses,  $\{x_i\}_{i=1}^n \subset \mathbb{R}^p$  are the observations,  $\beta \in \mathbb{R}^p$  is a vector of unknown coefficients, and  $\{\varepsilon_i\}_{i=1}^n$  are independent and identically distributed (i.i.d.) error terms with zero mean and constant variance  $\sigma^2$ . Let  $y = (y_1, \dots, y_n) \in \mathbb{R}^n$  be the response vector,  $X = (x_1, \dots, x_n)^\top \in \mathbb{R}^{n \times p}$  be the predictor matrix, and  $\varepsilon = (\varepsilon_1, \dots, \varepsilon_n) \in \mathbb{R}^n$  be the noise vector. In this study, we assume that  $n \gg p$ ,  $p$  is fixed, and  $X$  is of full column rank. We focus on the estimation of slope parameters and assume that the full data have been centralized. It is widely known that the ordinary least squares (OLS) estimator of  $\beta$  takes the form

$$\widehat{\beta}_{\text{OLS}} = (X^\top X)^{-1} X^\top y. \tag{2}$$

In practice, the calculation of the least squares problem may suffer from high computational costs. Specifically, standard computation of the formulation (2) requires  $O(np^2)$  computational time, which can be considerable when both  $n$  and  $p$  are large. To tackle the computational burden, various subsampling methods have been proposed. The main idea of subsampling methods can be described as follows:

given a predictor matrix  $X \in \mathbb{R}^{n \times p}$ , subsample  $r$  rows (i.e.,  $r$  observations) from  $X$  to construct a much smaller matrix  $\widehat{X} \in \mathbb{R}^{r \times p}$ , and then use it as a surrogate for  $X$  in downstream analysis.

Most of the existing subsampling methods can be divided into two classes, i.e., the randomized subsampling approach and the design-based subsampling approach. The former class aims to carefully design a data-dependent non-uniform sampling probability distribution such that more informative data points will be selected with larger sampling weights (Ma et al. 2015; Meng et al. 2017; Knight 2018; Ma et al. 2020; Ai et al. 2021). In contrast, the latter class aims to construct the most effective subsample estimator based on certain optimality criteria developed in the design of experiments (Wang et al. 2019; Meng et al. 2021; Wang et al. 2021, 2022; Chasiotis and Karlis 2024). Despite the numerous subsampling algorithms proposed, most of them rely on row-wise sampling, which can be less effective when dealing with (numerically) sparse predictor matrices, as discussed in Sect. 1. In contrast, element-wise sampling is more adept at exploiting the inherent sparsity of data, motivating the development of the core-elements method. See Table 1 for a comprehensive comparison. We also refer readers to Li and Meng (2021) and Yu et al. (2023) for recent reviews.

Closely related to the subsampling methods are the core-sets methods. These methods aim to select a subsample in a deterministic way, such that a loss function  $\mathcal{L}$  based on the subsample estimator is bounded by the loss function based on the full-sample estimator multiplying a constant  $(1 + \epsilon)$  (Boutsidis et al. 2013; Munteanu et al. 2018; Feldman et al. 2020; Maalouf et al. 2022). In linear models, a subsample  $\widehat{X}$  is called an  $(1 + \epsilon)$ -coreset ( $\epsilon > 0$ ), if there exists an estimator  $\widetilde{\beta}$  constructed by  $\widehat{X}$ , such that

$$\|y - X\widehat{\beta}_{\text{OLS}}\|^2 \leq \|y - X\widetilde{\beta}\|^2 \leq (1 + \epsilon)\|y - X\widehat{\beta}_{\text{OLS}}\|^2.$$

## 2.2 Sparse and numerically sparse matrices

Recall that sparse matrices are matrices in which most elements are zero. Commonly, the sparsity is measured by using the  $\ell_0$  norm (i.e., the number of non-zero elements). Such a procedure, however, may not accurately reflect the simplicity of nearly sparse instances with a large number of small but non-zero elements. To combat the obstacle, existing literature also considers the so-called numerically sparse matrices (Gupta and Sidford 2018; Braverman et al. 2021). Intuitively, numerically sparse is a weaker condition than sparse in which we do not require most elements to be zero, only that most elements are small enough to be ignored. Examples of numerically sparse are widely encountered in practice, including but not limited to linear programming constraints of the form  $x_1 \geq \sum_{i=1}^n x_i/n$ , and physical models whose strength of interaction decays with distance (Carmon et al. 2020).

## 2.3 Element-wise sampling on sparse data matrices

Consider the scenario that the observed predictor matrix  $X \in \mathbb{R}^{n \times p}$  is a (numerically) sparse matrix in which most of the elements are (nearly) zero. Oftentimes, of interest is to find a sparser matrix  $X^+ \in \mathbb{R}^{n \times p}$  that is a good proxy for  $X$ . Here,  $X^+$  is also a sparse matrix, from which the non-zero elements are a subset of the non-zero elements with respect to (w.r.t.)  $X$ . The problem of finding such a matrix  $X^+$  has many applications in eigenvector approximation (Arora et al. 2006; Achlioptas and Mcsherry 2007; El Karoui and d'Aspremont 2010; Kundu et al. 2017; Gupta and Sidford 2018), semi-definite programming (Arora et al. 2005; d'Aspremont 2011; Garber and Hazan 2016), matrix completion (Candès and Recht 2009; Candès and Tao 2010; Chen et al. 2014), matrix multiplication (Mahoney 2016; Li et al. 2023), etc.

To construct such a matrix  $X^+$ , most existing methods aim to design a good sampling probability distribution  $p_{ij}$  ( $i = 1, \dots, n; j = 1, \dots, p$ ), such that the approximation error  $\|X - X^+\|_2$  is as small as possible given a sampling budget  $s$ . Achlioptas and Mcsherry (2007) proposed the so-called  $\ell_2$  sampling such that  $p_{ij} \propto x_{ij}^2$ , which was later refined by Drineas and Zouzias (2011). Alternatively, Arora et al. (2006) proposed  $\ell_1$  sampling, where  $p_{ij} \propto |x_{ij}|$ . Later, Achlioptas et al. (2013) proposed a near-optimal probability distribution using matrix-Bernstein inequality. Their approach can be regarded as a combination of two  $\ell_1$ -based distributions:  $p_{ij} \propto |x_{ij}|$  when  $s$  is small, and  $p_{ij} \propto |x_{ij}| \cdot \|x_i\|_1$  as  $s$  grows. More recently, Kundu et al. (2017) developed hybrid- $(\ell_1, \ell_2)$  sampling, which is also a convex combination of probabilities previously proposed.

In general, existing literature on element-wise sampling mainly focus on the algorithmic perspective, such that the applications of interest are usually compressed sensing and matrix recovery. Nevertheless, element-wise sampling with

a statistical perspective, such that the applications involve providing effective and efficient solutions for large-scale statistical modeling and inference, is still a blank field and remains further studied.

## 3 Core-elements

In this section, we present our main algorithm and the theoretical properties of the proposed estimator. We first introduce the definition of core-elements and then construct an unbiased core-elements estimator. Following this, we derive an upper bound for the variance of such an estimator by utilizing the matrix-form Taylor expansion. Subsequently, we provide an algorithm for core-elements selection, resulting in an estimator that approximately minimizes this upper bound. Furthermore, a coresets-like finite sample bound is provided to quantify the approximation error for the proposed estimator.

### 3.1 Problem setup

We consider the problem of large-scale least squares estimation when the predictor matrix  $X$  is sparse or numerically sparse. To utilize the sparse structure, we propose to construct a sparser subdata matrix  $X^* \in \mathbb{R}^{n \times p}$  from  $X$  carefully, and use  $X^*$  to construct an efficient least squares approximation. In particular, given a positive integer  $s (< np)$ , let  $S$  be an  $n \times p$  matrix such that its elements involve  $s$  ones and  $np - s$  zeros. The subdata matrix  $X^*$  then can be formulated as  $X^* = S \odot X$ , where  $\odot$  represents the Hadamard product, i.e., element-wise product. In other words,  $X^*$  is produced by keeping  $s$  elements of  $X$  and zeroing out the remaining elements. We then propose a general estimator that takes the form

$$\tilde{\beta}(D) = DX^{*\top}y, \quad (3)$$

where  $D \in \mathbb{R}^{p \times p}$  is a scaling matrix to be determined. For simplicity, suppose that both  $X^\top X$  and  $X^{*\top} X$  are invertible in context; otherwise, their inverse operation should be replaced by a generalized inverse.

A natural question arises: given a fixed budget  $s$ , among all the estimators that take the form (3), how to construct the scaling matrix  $D$  and the sparse subdata matrix  $X^*$ , such that (i) the estimator  $\tilde{\beta}(D)$  is unbiased and (ii) its estimation variance is as small as possible? For the first part of this question, one can see that when  $D = (X^{*\top} X)^{-1}$ ,  $E\{\tilde{\beta}(D)\} = (X^{*\top} X)^{-1} X^{*\top} \{X\beta + E(\varepsilon)\} = \beta$ , indicating that  $\tilde{\beta}(D)$  is unbiased to  $\beta$ . Such a finding motivates us to focus on the unbiased estimator

$$\tilde{\beta} = (X^{*\top} X)^{-1} X^{*\top} y.$$

Consider the classical subsample-based estimator

$$\tilde{\beta}' = (\widehat{X}^\top \widehat{X})^{-1} \widehat{X}^\top \widehat{y},$$

where  $\widehat{X} \in \mathbb{R}^{r \times p}$  represents a subsample (of full column rank) consisting of  $r$  rows from  $X$ , and  $\widehat{y} \in \mathbb{R}^r$  contains the corresponding elements from  $y$ . When the selected  $s = rp$  elements are located in  $r$  rows, i.e., when the element-wise subset selection degenerates to the row selection, it can be shown that  $\tilde{\beta} = \tilde{\beta}'$ . While in general cases, these two estimators are different.

Next, we consider the variance of the estimator  $\tilde{\beta}$ . Recall that  $\sigma^2$  represents the variance of random errors in model (1). Our goal is to find a subdata matrix  $X^*$  that minimizes the estimation variance, which has a closed form

$$\begin{aligned} E(\|\tilde{\beta} - \beta\|^2 | X) &= E\{y^\top X^* (X^\top X^*)^{-1} (X^{*\top} X)^{-1} X^{*\top} y | X\} - \beta^\top \beta \\ &= \beta^\top \beta + \sigma^2 \text{tr}\{X^* (X^\top X^*)^{-1} (X^{*\top} X)^{-1} X^{*\top}\} - \beta^\top \beta \\ &= \sigma^2 \|(X^{*\top} X)^{-1} X^{*\top}\|_F^2. \end{aligned} \tag{4}$$

However, this variance term is challenging to minimize directly. To overcome the obstacle, we provide an upper bound for (4) and aim to minimize the upper bound instead. The upper bound is derived by utilizing the matrix-form Taylor expansion (see Chapter 1, Higham (2008)), detailed in Lemma 1.

**Lemma 1** *Let  $L = X - X^*$ . A Taylor expansion of  $E(\|\tilde{\beta} - \beta\|^2 | X)$  around the point  $X^* = X$  yields the following upper bound,*

$$E(\|\tilde{\beta} - \beta\|^2 | X) \leq \sigma^2 [\text{tr}\{(X^\top X)^{-1}\} + \|(X^\top X)^{-1}\|_2^2 \|L\|_F^2 \{1 + O(\lambda_0)\}]. \tag{5}$$

Here, assume the spectral radius  $\lambda_0 = \|(X^\top X)^{-1} L^\top X\|_2 < 1$  to ensure the convergence of the matrix series.

When the Taylor expansion in Lemma 1 is valid, the inequality (5) indicates that the upper bound of the estimation variance decreases as  $\|L\|_F$  and the remainder  $\lambda_0$  decreases. Considering  $\lambda_0$ , we have

$$\begin{aligned} \lambda_0 &\leq \|(X^\top X)^{-1}\|_2 \|X\|_2 \|L\|_2 \\ &\leq \|(X^\top X)^{-1}\|_2 \|X\|_2 \left( p \max_{j \in \{1, \dots, p\}} l^{(j)\top} l^{(j)} \right)^{1/2}, \end{aligned}$$

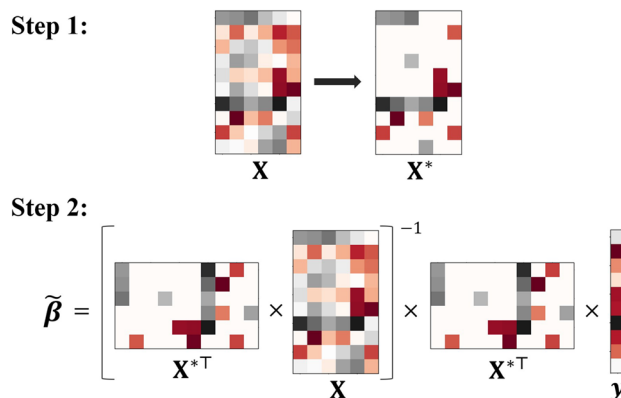
where  $l^{(j)}$  denotes the  $j$ th column of  $L$ . Such an inequality indicates that a smaller value of the maximum column norm of  $L$  is associated with a smaller  $\lambda_0$ . As a result, to minimize the upper bound in Lemma 1, we need to keep both  $\|L\|_F$  and the column norms of  $L$  as small as possible.

Motivated by this, we propose to construct  $X^*$  by keeping the elements with the top largest absolute values w.r.t. each column of  $X$  and zeroing out the remaining. Intuitively, for a fixed number of selected elements, such  $L$  has the approximately minimum column norm respecting every column. Thus, both the values of  $\|L\|_F$  and  $\|L\|_2$  will be approximately minimized, resulting in a relatively small upper bound of the estimation variance in Lemma 1. Moreover, the column-wise process can prevent any entire column of  $X$  from being discarded, thus avoiding producing a singular  $X^*$  and ensuring the estimability of coefficients. We call such a procedure ‘‘core-elements’’ since the idea behind it is analogous to ‘‘coresets’’, except that what we select are elements instead of rows.

### 3.2 Main algorithm

Without loss of generality, we assume the number of selected elements  $s = rp$ , where  $r$  is an integer. Algorithm 1 summarizes the construction of core-elements and the proposed estimator, which are illustrated in Fig. 1.

Consider the computational cost of Algorithm 1. Constructing the matrix  $X^*$  using a partition-based selection algorithm requires  $O(\text{nnz}(X))$  time (Musser 1997; Martinez 2004; Wang et al. 2019). Each column of  $X^*$  contains at most  $r$  non-zero elements, thus calculating  $X^{*\top} X$  takes  $O(rp^2)$  time by using sparse matrix representations and operations. The computing time for  $\tilde{\beta}$  is thus at the order of  $O(rp^2 + p^3)$ . Therefore, the overall computational cost of Algorithm 1 is  $O(\text{nnz}(X) + rp^2)$ , which becomes  $O(\text{nnz}(X))$  when  $n \gg r$ . In the large-sample scenario such that  $n \gg p$ , such an algorithm is much faster than the popular leverage-based subsampling methods (Ma et al. 2015). This is because the



**Fig. 1** Illustration for Algorithm 1. Each element of a matrix is labeled with a different color, such that larger positive and smaller negative values are labeled with more red and more black colors, respectively. Step 1 illustrates the core-elements selection, where  $r = 3$  elements with the largest absolute values w.r.t. each column are selected. Step 2 illustrates the proposed estimator

leverage-based methods involve the singular value decomposition of the full predictor matrix, requiring a cost of the order  $O(np^2)$ .

**Algorithm 1** CORE- ELEMENTS( $X, y, r$ )

- 1: **Input:**  $X = (x_{ij}) \in \mathbb{R}^{n \times p}, y \in \mathbb{R}^n, r \in \mathbb{Z}_+$
- 2: Initialize  $S = (0) \in \mathbb{R}^{n \times p}$
- 3: **for**  $j = 1, \dots, p$  **do**
- 4:   Let  $\mathcal{I} = \{i_1, \dots, i_r\}$  be an index set, such that  $\{|x_{i_q j}|\}_{q=1}^r$  are the  $r$  largest ones among  $\{|x_{i_j}|\}_{i=1}^n$
- 5:   Let  $s_{i_q j} = 1, q = 1, \dots, r$
- 6:  $X^* = S \odot X$ , where  $\odot$  represents the element-wise product
- 7: **Return**  $\tilde{\beta} = (X^{*\top} X)^{-1} X^{*\top} y$

**3.3 Theoretical properties**

We now present our main theorem, indicating that the selected elements are  $(1 + \epsilon)$ -core-elements for least squares estimation. In other words, under some regularity conditions, the proposed estimate achieves the  $(1 + \epsilon)$ -relative error w.r.t. the  $\ell_2$  loss. Technical proofs are provided in the supplementary material.

**Theorem 1** *Given  $X \in \mathbb{R}^{n \times p}, y \in \mathbb{R}^n$  and  $\epsilon > 0$ , let  $X^*$  be the subdata matrix and  $\tilde{\beta}$  be the estimate that calculated by Algorithm 1. When  $X^*$  satisfies  $\|X - X^*\|_2 \leq \epsilon' \|X\|_2$  with*

$$0 < \epsilon' \leq \frac{1}{\kappa^2(X)} \left[ 1 + \frac{\{\kappa^2(X) + 1\} \|y\|}{\epsilon^{1/2} \|y - X\widehat{\beta}_{OLS}\|} \right]^{-1}, \tag{6}$$

we have

$$\|y - X\widehat{\beta}_{OLS}\|^2 \leq \|y - X\tilde{\beta}\|^2 \leq (1 + \epsilon) \|y - X\widehat{\beta}_{OLS}\|^2. \tag{7}$$

Theorem 1 indicates that to achieve the  $(1 + \epsilon)$ -relative error in inequality (7), Algorithm 1 requires a subdata matrix  $X^*$  such that the ratio  $\|X - X^*\|_2 / \|X\|_2$  is  $O(\epsilon^{1/2})$ . This rate is supported by empirical results in Sect. 5. Intuitively, such a result also indicates that when the predictor matrix  $X$  gets (numerically) sparser, fewer elements are required in  $X^*$  to achieve the same relative error w.r.t. the  $\ell_2$  loss.

In addition, the value of  $\epsilon'$  also depends on the condition number  $\kappa(X)$  and the relative sum of squares error (SSE)  $\|y - X\widehat{\beta}_{OLS}\|^2 / \|y\|^2$ . Specifically, a larger  $\epsilon'$  is admitted to achieve the  $(1 + \epsilon)$ -relative error if the condition number  $\kappa(X)$  decreases and the SSE increases. The following remark discusses the relationship between  $r$  and  $\epsilon'$ .

**Remark 1** Suppose  $X$  is a sparse covariate matrix with  $\alpha \times 100\%$  non-zero elements ( $0 < \alpha \leq 1$ ), and each column has the same number of non-zero elements. Further, suppose the

non-zero elements of  $X$  are i.i.d. from a continuous cumulative distribution function  $F$ . We consider two specific cases of the distribution  $F$  as follows.

- *Case 1* uniform distribution over  $(-1, 1)$ .  
If the subsample parameter  $r$  in Algorithm 1 satisfies  $r < \alpha n$  and

$$\frac{r}{n} \geq \alpha - \frac{(\alpha \epsilon' \|X\|_2)^{2/3}}{(2np)^{1/3}},$$

then the subdata matrix  $X^*$  achieves the condition  $\|X - X^*\|_2 \leq \epsilon' \|X\|_2$  in Theorem 1 with high probability for a relatively large  $n$ . Under a general condition  $\|X\|_2 = O((np)^{1/2})$ , which can be satisfied as long as all the elements in  $X$  are bounded, such a result indicates that Algorithm 1 needs to select around  $\{\alpha - (c\alpha\epsilon')^{2/3}\} \times 100\%$  elements to achieve the  $(1 + \epsilon)$ -relative error for a constant  $c > 0$ .

- *Case 2* standard normal distribution on  $\mathbb{R}$ .  
If  $r < \alpha n$  satisfies

$$\frac{r}{n} \geq \alpha - \min \left\{ \alpha\phi, \frac{(\epsilon' \|X\|_2)^2}{2G^{-1}(\phi)np} \right\}, \tag{8}$$

where  $0 < \phi < 1$  and  $G$  is the cumulative distribution function of the chi-squared distribution with 1 degree of freedom, then the condition  $\|X - X^*\|_2 \leq \epsilon' \|X\|_2$  holds with high probability when  $n$  is sufficiently large. To achieve the  $(1 + \epsilon)$ -relative error, (8) indicates us to select  $[\alpha - \min\{\alpha\phi, (c\epsilon')^2 / G^{-1}(\phi)\}] \times 100\%$  elements under the condition  $\|X\|_2 = O((np)^{1/2})$ , where  $c > 0$  is a constant.

**4 MOM core-elements**

To guarantee the robustness of the core-elements approach, we modify Algorithm 1 by combining it with the popular median-of-means (MOM) procedure (Hsu and Sabato 2014; Lugosi and Mendelson 2019; Lecu e and Lerasle 2019, 2020; Mathieu 2021; Huang and Lederer 2023) to provide a robust version of core-elements, called ‘‘MOM core-elements’’. We first propose an algorithm in the divide-and-conquer framework to obtain the MOM core-elements estimator, and then we establish consistency of the proposed estimator under regularity conditions.

**4.1 Proposed algorithm**

Due to the existence of outliers, we relax the standard i.i.d. setup to the  $\mathcal{I} \cup \mathcal{O}$  framework following the work of Lecu e and Lerasle (2020). Specifically, we assume that data are

partitioned into two (unknown) sets,  $\mathcal{I}$  and  $\mathcal{O}$ , such that  $\mathcal{I} \cup \mathcal{O} = \{1, \dots, n\}$  and  $\mathcal{I} \cap \mathcal{O} = \emptyset$ . Data  $\{(x_i, y_i)\}_{i \in \mathcal{I}}$  are i.i.d. informative data, and data  $\{(x_i, y_i)\}_{i \in \mathcal{O}}$  are outliers on which no assumption is granted.

Given a random partition of the full data  $(X, y) = \{(x_i, y_i)\}_{i=1}^n$  into blocks of equal sizes, the principle of MOM estimator is that we first obtain an estimation on each block independently, and then we aggregate the results from these blocks by taking the median. Let  $k$  be the number of blocks and  $n_l$  ( $l = 1, \dots, k$ ) be the size of each block. Without loss of generality, we assume that both  $n$  and  $r$  are divisible by  $k$ . Then  $n_l \equiv n/k$  for  $l = 1, \dots, k$ . Denote  $(X^{(l)}, y^{(l)}) \in \mathbb{R}^{n_l \times p} \times \mathbb{R}^{n_l}$  be the data partitioned into the  $l$ th block. For the  $l$ th block, we construct the core-elements matrix  $X^{(l)*}$  containing  $r_l p$  non-zero elements, where  $r_l \equiv r/k$  for  $l = 1, \dots, k$ . Next, we obtain the corresponding estimator  $\tilde{\beta}^{(l)} = (X^{(l)*\top} X^{(l)})^{-1} X^{(l)*\top} y^{(l)}$ . Then, the MOM core-elements estimator is defined as

$$\tilde{\beta}_{\text{MOM}} = \text{med}(\tilde{\beta}^{(1)}, \dots, \tilde{\beta}^{(k)}), \tag{9}$$

where  $\text{med}(\cdot)$  denotes the coordinate-wise median. Algorithm 2 details the MOM core-elements procedure.

---

**Algorithm 2** MOM CORE-ELEMENTS( $X, y, r, k$ )

---

- 1: **Input:**  $X = (x_{ij}) \in \mathbb{R}^{n \times p}$ ,  $y \in \mathbb{R}^n$ ,  $r, k \in \mathbb{Z}_+$
  - 2: Partition  $(X, y)$  into  $k$  blocks  $\{(X^{(l)}, y^{(l)})\}_{l=1}^k$  randomly and evenly
  - 3: **for**  $l = 1, \dots, k$  **do**
  - 4:     Compute  $\tilde{\beta}^{(l)} = \text{CORE-ELEMENTS}(X^{(l)}, y^{(l)}, r/k)$
  - 5: **Return**  $\tilde{\beta}_{\text{MOM}} = \text{med}(\tilde{\beta}^{(1)}, \dots, \tilde{\beta}^{(k)})$
- 

In Algorithm 2, the number of blocks  $k$  indicates the robustness of our proposed estimator. A popular measure to quantify robustness is the breakdown point (Hampel 1968; Donoho and Huber 1983; Donoho and Gasko 1992; Lecué and Lerasle 2020), defined as the smallest proportion of corrupted observations needed to push an estimator to infinity. The breakdown point of Algorithm 2 is  $\lfloor k/2 \rfloor/n$ , because less than  $\lfloor k/2 \rfloor$  outliers may corrupt at most  $\lfloor k/2 \rfloor$  blocks, leaving the median in (9) equal to the estimation on a single block with uncorrupted data. Remark that Algorithm 1 is a special case of Algorithm 2 when  $k = 1$ , which is applicable to datasets without extreme outliers.

Consider the computation time of Algorithm 2. Constructing the core-elements estimator on the  $l$ th block requires  $O(\text{nnz}(X^{(l)}) + rp^2/k + p^3)$  time, and the aggregation step needs  $O(kp)$  time. Thus, the total computational cost of Algorithm 2 is at the order of  $O(\text{nnz}(X) + rp^2 + kp^3 + kp) = O(\text{nnz}(X) + rp^2)$ , where the equation holds because  $r/k$  should be at least of the same order as  $p$  for ensuring the well-definedness of  $\tilde{\beta}^{(1)}, \dots, \tilde{\beta}^{(k)}$ . Such cost is the same as that of Algorithm 1.

## 4.2 Theoretical properties

Now we provide the convergence of the MOM core-elements estimator. To begin with, we introduce the following regularity conditions.

- (H1) Denote the Fisher information matrix  $I^{(l)} = n_l^{-1} X^{(l)\top} X^{(l)}$  on the  $l$ th block for  $l = 1, \dots, k$ . Assume that  $c < \inf_l \lambda_{\min}(I^{(l)}) \leq \sup_l \lambda_{\max}(I^{(l)}) < \infty$  for some constant  $c > 0$ , where  $\lambda_{\max}(\cdot)$  and  $\lambda_{\min}(\cdot)$  respectively stand for the maximum and minimum eigenvalues.
- (H2) Let  $L^{(l)} = X^{(l)} - X^{(l)*}$  for  $l = 1, \dots, k$ . Assume that  $\sup_l \|L^{(l)}\|_F^2/n_l^2 \rightarrow 0$  as  $n_l \rightarrow \infty$ .
- (H3) Assume that the spectral radius  $\lambda_0^{(l)} = \|(X^{(l)\top} X^{(l)})^{-1} L^{(l)\top} X^{(l)}\|_2 < 1$  for  $l = 1, \dots, k$ .
- (H4) Suppose that  $k > 2|\mathcal{B}_{\mathcal{O}}| + 1$ , where  $\mathcal{B}_{\mathcal{O}}$  is the set of blocks containing at least one outlier and  $|\cdot|$  denotes the cardinal number. Further, assume that  $\mathcal{B}_{\mathcal{O}}$  contains finite elements.

**Theorem 2** Suppose the conditions (H1)–(H4) hold almost surely. As  $k, n_l \rightarrow \infty$  ( $l = 1, \dots, k$ ),  $\tilde{\beta}_{\text{MOM}}$  converges to  $\beta$  in probability, i.e., for any given  $\epsilon > 0$ , it holds that

$$\text{pr}(\|\tilde{\beta}_{\text{MOM}} - \beta\| > \epsilon) \rightarrow 0.$$

In Theorem 2, condition (H1) is commonly assumed in the literature. Conditions (H2) and (H3) bound the error of the local core-elements estimator on each block by using Lemma 1. Condition (H4) guarantees the effectiveness of the MOM procedure according to its breakdown point discussed above.

## 5 Simulation studies

In this section, we first evaluate the performance of core-elements (i.e., Algorithm 1) in estimating  $\beta$  and predicting  $y$  on uncorrupted synthetic datasets. Subsequently, we provide empirical evidence to support the error bound in Theorem 1. Next, we consider corrupted datasets to show the effectiveness and robustness of MOM core-elements (i.e., Algorithm 2). Finally, we demonstrate the advantage of the proposed strategy over other subsampling methods w.r.t. computational efficiency.

### 5.1 Performance on uncorrupted data

We use CORE to refer to the estimator in Algorithm 1. For comparison, we consider the full sample OLS estimation (FULL) and several state-of-the-art subsampling methods mentioned in Table 1, including uniform subsampling (UNIF), doubly sketching (DOUBLY) (Hou-Liu and

Browne 2023), basic leverage subsampling (BLEV) (Drineas et al. 2006; Ma et al. 2015), shrinkage leverage subsampling (SLEV) with shrinkage parameter being 0.9 (Ma et al. 2015), information-based optimal subset selection (IBOSS) (Wang et al. 2019), orthogonal subsampling (OSS) (Wang et al. 2021), and D-optimal subsampling (DOPT) (Reuter and Schwabe 2024).

For uncorrupted data, the predictor matrix  $X$  is generated from different kinds of widely-used distributions:

- (D1) Multivariate normal distribution,  $N(0_p, \Sigma)$ ;
- (D2) Multivariate log-normal distribution,  $LN(0_p, \Sigma)$ ;
- (D3) Multivariate t-distribution with 3 degrees of freedom,  $t_3(0_p, \Sigma)$ ,

where  $\Sigma = (\sigma_{ij}) \in \mathbb{R}^{p \times p}$  is a covariance matrix with  $\sigma_{ij} = 0.6^{|i-j|}$  for  $i, j = 1, \dots, p$ .

To introduce sparsity, after generating the predictor matrix and centering it, we randomly zero out their elements with a sparsity ratio  $\alpha$ . Specifically, we randomly select  $\alpha \times 100\%$  of the elements and set these elements to be zero. Consider  $\alpha = \{0, 0.2, 0.4, 0.6, 0.8\}$ , referred to as (R1)–(R5), respectively. Next, we add a small random perturbation following  $U(-0.1, 0.1)$  to each zero element of the predictor matrix to obtain a numerically sparse matrix. Then, (R1) corresponds to a completely dense matrix, and (R5) corresponds to a highly numerically sparse matrix. We then generate the response  $y$  from the linear model (1). The true coefficient  $\beta$  is a  $p$ -dimensional vector of ones, and the signal-to-noise ratio, defined as  $SNR = \text{var}(X\beta)/\sigma^2$ , is set to be 4. The simulations in misspecified linear models and alternative choices of true  $\beta$  are relegated to the supplementary material. Let the sample size  $n = 10^4$  and dimension  $p = 10^2$ . For the row-wise subsampling methods, we select  $r \in \{2p, 4p, 6p, 8p, 10p\}$  rows for each of these methods. For a fair comparison, we select  $s = rp$  elements for the proposed core-elements method. We calculate the empirical mean squared error (MSE) for each of the estimators based on one hundred replications, i.e.,

$$MSE = \frac{1}{100} \sum_{i=1}^{100} \frac{\|\widehat{\beta}^{(i)} - \beta\|^2}{\|\beta\|^2}, \tag{10}$$

where  $\widehat{\beta}^{(i)}$  represents the estimator in the  $i$ th replication. We also consider the prediction MSE (PMSE). For the  $i$ th replication, we randomly split the observed sample into a training set  $(y_{\text{train}}^{(i)}, X_{\text{train}}^{(i)})$  of size  $\lfloor 0.7n \rfloor$  and a test set  $(y_{\text{test}}^{(i)}, X_{\text{test}}^{(i)})$  of size  $\lfloor 0.3n \rfloor$ . For each subsampling method, we select a subset from the training set leading to an estimator  $\widehat{\beta}_{\text{train}}^{(i)}$ , and then use it to predict the response  $y_{\text{test}}$  in the test set. In this way, PMSE is calculated as

$$PMSE = \frac{1}{100} \sum_{i=1}^{100} \frac{\|X_{\text{test}}^{(i)} \widehat{\beta}_{\text{train}}^{(i)} - y_{\text{test}}^{(i)}\|^2}{\|y_{\text{test}}^{(i)}\|^2}. \tag{11}$$

The results of  $\log(\text{MSE})$  and  $\log(\text{PMSE})$  versus different subsample sizes are shown in Figs. 2 and 3, respectively.

In Figs. 2 and 3, we observe that both MSE and PMSE w.r.t. all estimators decreases as  $r$  increases. We also observe that CORE consistently outperforms other subsampling approaches under all circumstances. The advantage becomes more apparent when the level of sparsity increases, i.e., from (R1) to (R5). Such an observation indicates the proposed estimator can provide a more effective estimate than the competitors, especially when the predictor matrix is of high sparsity. Such success can be attributed to the fact that the proposed core-elements approach can effectively utilize the sparsity structure of the predictor matrix, and the proposed estimator is unbiased and has an approximately minimized estimation variance.

While other deterministic subsampling methods (i.e., IBOSS, OSS, and DOPT) are also competitive, their performance varies significantly across different data distributions. Specifically, the OSS method performs well with normally distributed data (D1), but it loses efficacy with asymmetric log-normal (D2) and heavy-tailed t-distributions (D3), performing worse than uniform sampling. This behavior aligns with the findings presented in Table 3 of Yu et al. (2023). In contrast, while IBOSS and DOPT are comparable to CORE in handling heavy-tailed distributions, they fall short in normal distributions. Overall, the proposed core-elements approach demonstrates both generality and superiority across a variety of data distributions.

In order to verify the theoretical error bound provided in Theorem 1, we compare the empirical and theoretical values of  $\epsilon$  under different values of  $\epsilon'$ , as shown in Fig. 4. Specifically, given a small  $\epsilon'$ , the empirical value of  $\epsilon$  is calculated as

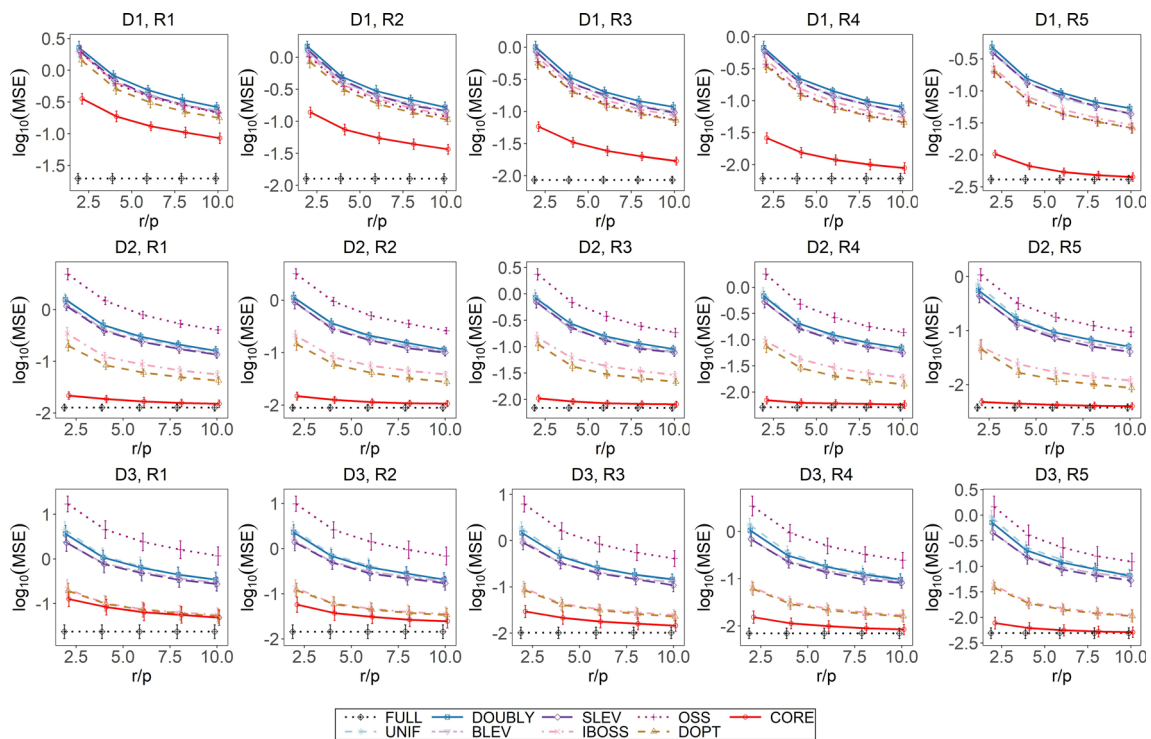
$$\frac{\|y - X\widetilde{\beta}\|^2}{\|y - X\widehat{\beta}_{\text{OLS}}\|^2} - 1$$

according to (7), and the theoretical value of  $\epsilon$  is calculated as

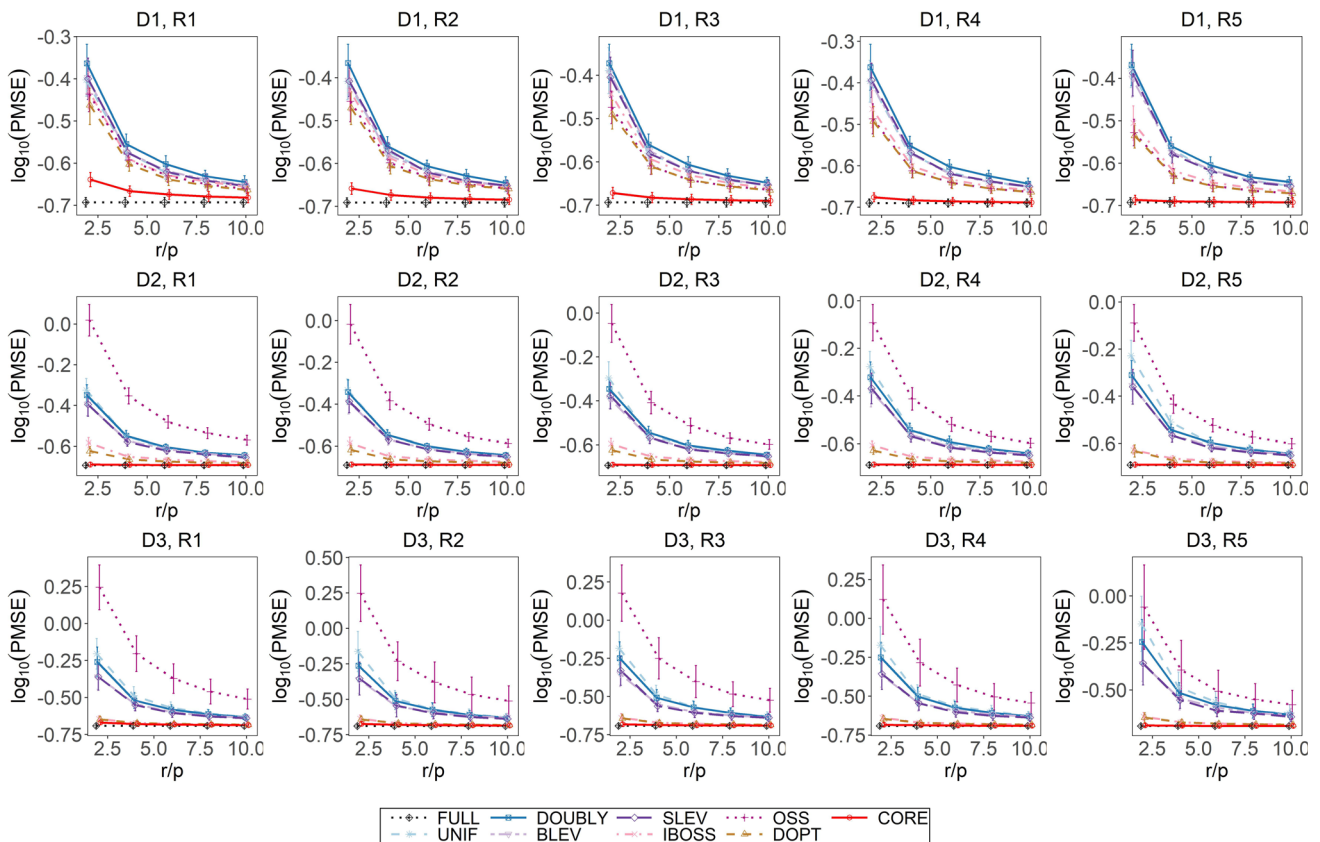
$$\left[ \frac{\epsilon' \kappa^2(X) \{\kappa^2(X) + 1\} \|y\|}{\{1 - \epsilon' \kappa^2(X)\} \|y - X\widehat{\beta}_{\text{OLS}}\|} \right]^2$$

according to (6). Both  $\epsilon$  and  $\epsilon'$  have been made logarithmic transformation in Fig. 4. We can observe that although the empirical and theoretical values of  $\epsilon$  differ, their growth trends have an apparent parallel pattern. This observation indicates that our proposed error bound and the empirical value are of the same order. Their difference is up to a constant under the log transformation.

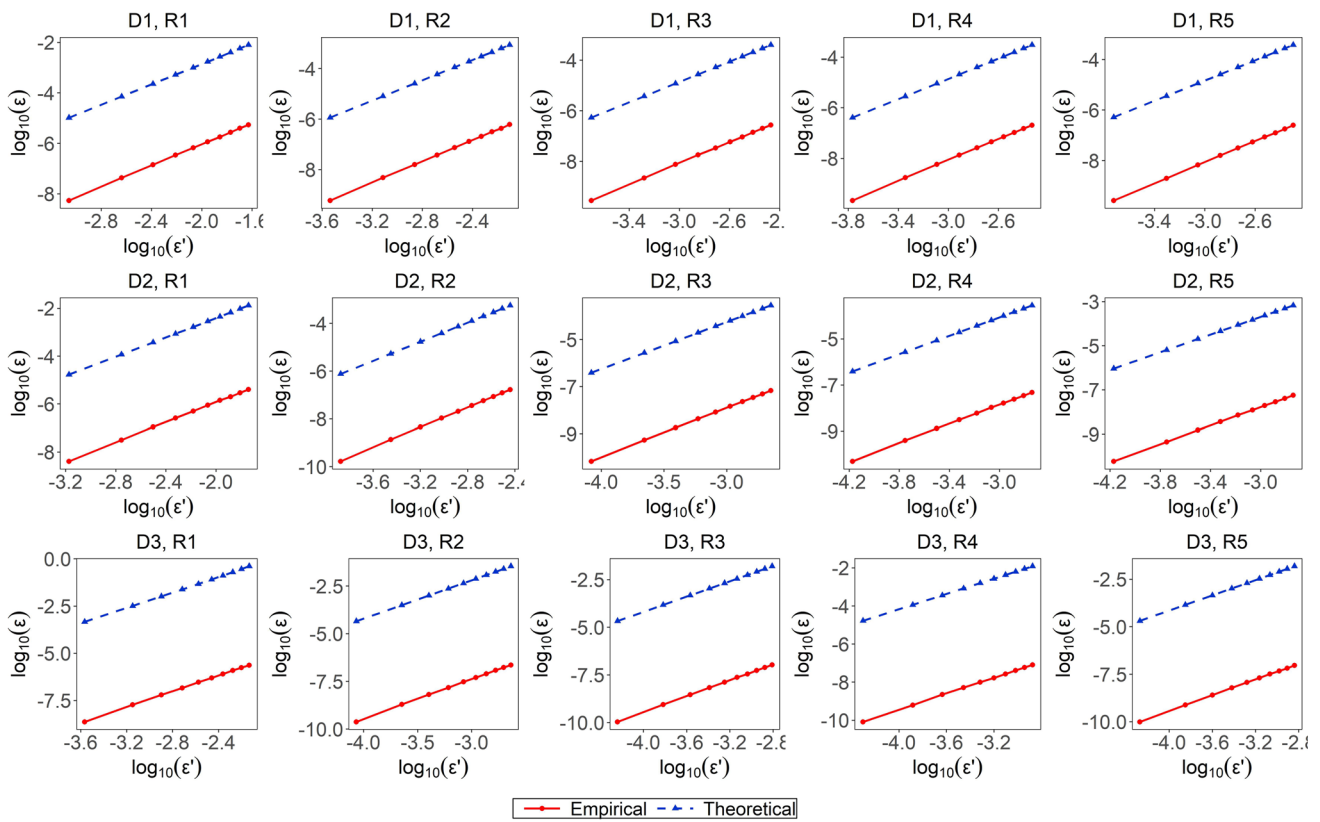




**Fig. 2** Comparison of different estimators w.r.t. MSE. Each row represents a particular data distribution, i.e., (D1)–(D3), and each column represents a different sparsity ratio, i.e., (R1)–(R5). Vertical bars are the standard errors



**Fig. 3** Comparison of different estimators w.r.t. PMSE. Each row represents a particular data distribution, i.e., (D1)–(D3), and each column represents a different sparsity ratio, i.e., (R1)–(R5). Vertical bars are the standard errors



**Fig. 4** Comparison of the empirical value and the theoretical value of the error term  $\epsilon$ . Each row represents a particular data distribution, i.e., (D1)–(D3), and each column represents a different sparsity ratio, i.e., (R1)–(R5)

### 5.2 Performance on corrupted data

We compare the proposed MOM core-elements approach (i.e., Algorithm 2), referred to as MOM-CORE, with the full sample OLS estimation and the subsampling methods mentioned above in the presence of outliers. To ensure fairness, all competing methods are equipped with the MOM procedure.

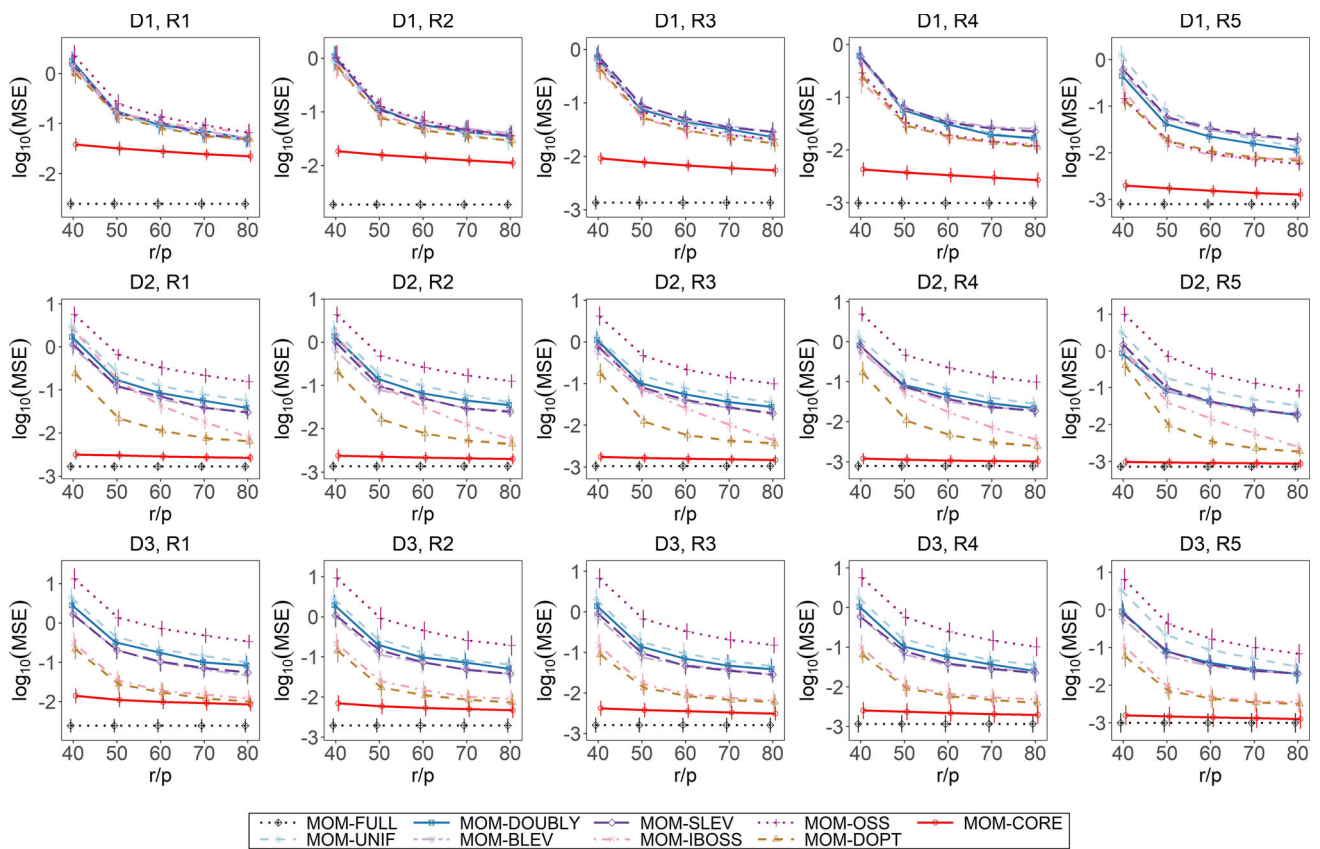
The corrupted data consist of informative data  $\{(x_i, y_i)\}_{i \in \mathcal{I}}$  and various types of outliers  $\{(x_i, y_i)\}_{i \in \mathcal{O}}$  with  $\mathcal{O} = \mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{O}_3 \cup \mathcal{O}_4$ , such that  $|\mathcal{O}| = n_o$  and  $|\mathcal{I}| = n - n_o$ .  $\{(x_i, y_i)\}_{i \in \mathcal{I}}$  is generated in the same way as in the previous section, and  $\{(x_i, y_i)\}_{i \in \mathcal{O}}$  is constructed following the setup in Lecué and Lerasle (2020). More precisely,

- For  $i \in \mathcal{O}_1$  with  $|\mathcal{O}_1| = \lceil n_o/4 \rceil$ ,  $y_i = 1000 + 10\zeta_1$  and  $x_i = -10 \times 1_p + \zeta_2$ , where  $\zeta_1 \in \mathbb{R}$  and  $\zeta_2 \in \mathbb{R}^p$  are noises following the (multivariate) standard normal distribution;
- For  $i \in \mathcal{O}_2$  with  $|\mathcal{O}_2| = \lceil n_o/4 \rceil$ ,  $y_i = -500 + 10\zeta_1$  and  $x_i = 10 \times 1_p + \zeta_2$ ;
- For  $i \in \mathcal{O}_3$  with  $|\mathcal{O}_3| = \lceil n_o/4 \rceil$ ,  $y_i$  is a 0 – 1-Bernoulli random variable and  $x_i$  is uniformly distributed over  $[0, 1]^p$ ;

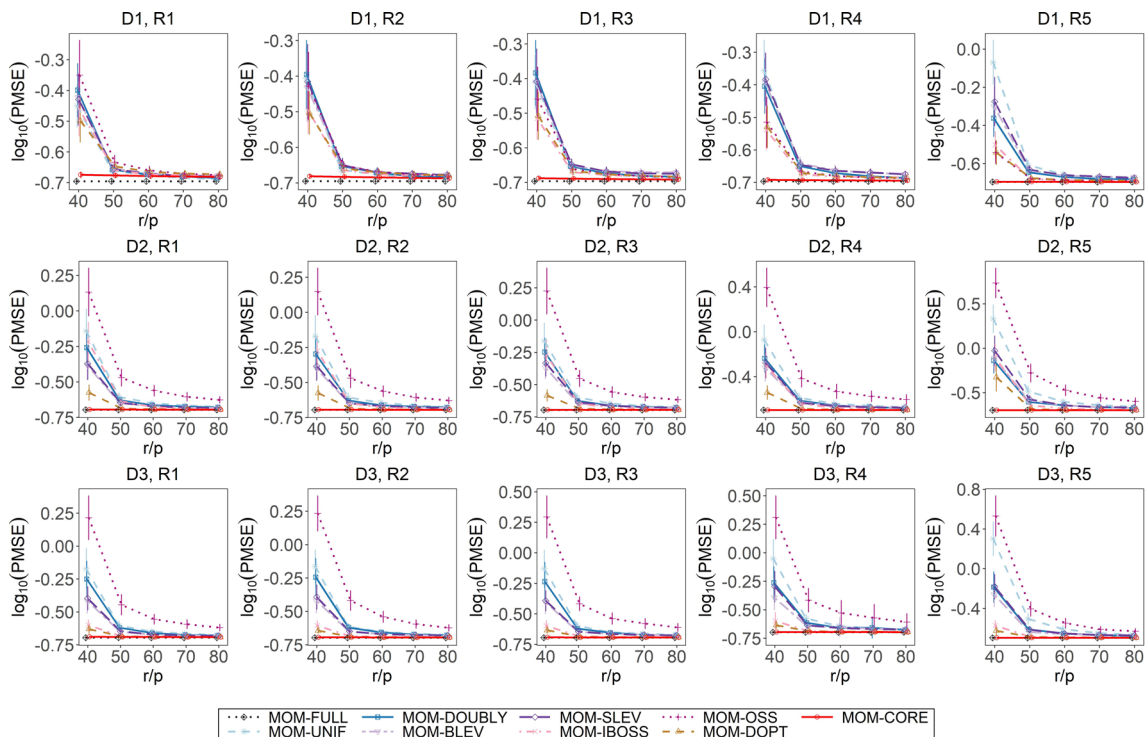
- For  $i \in \mathcal{O}_4$ ,  $(x_i, y_i)$  is generated from the linear model (1) with the same true parameter  $\beta = 1_p$  but for a different choice of design  $X$  and noise  $\epsilon$ . Here, we take the covariance matrix  $\Sigma$  as an identity matrix and  $\epsilon$  is a heavy-tailed noise following  $t_2$  distribution.

After generating the corrupted data above, observations in  $\mathcal{I}$  and  $\mathcal{O}$  are merged and shuffled before downstream operations. Let the sample size  $n = 5 \times 10^4$ , the dimension  $p = 20$ , and the number of outliers  $n_o = 19$ . We subsample  $r \in \{40p, 50p, 60p, 70p, 80p\}$  rows for row-sampling methods or  $s = rp$  elements for MOM-CORE. The number of blocks is set to be  $k = 40$  for the MOM procedure. Other settings are the same as those in the above section.

To evaluate the performance of different methods on corrupted data, we calculate MSE according to (10) and PMSE according to (11) for each method, and their results versus increasing subsample sizes are shown in Figs. 5 and 6, respectively. Remark that when predicting the responses, we first split the training set and test set on informative data  $\{(x_i, y_i)\}_{i \in \mathcal{I}}$ , and then add outliers  $\{(x_i, y_i)\}_{i \in \mathcal{O}}$  to the training set; that is, the test set is not corrupted by outliers.



**Fig. 5** Comparison of different estimators w.r.t. MSE on corrupted data. Each row represents a particular data distribution, i.e., (D1)–(D3), and each column represents a different sparsity ratio, i.e., (R1)–(R5). Vertical bars are the standard errors



**Fig. 6** Comparison of different estimators w.r.t. PMSE on corrupted data. Each row represents a particular data distribution, i.e., (D1)–(D3), and each column represents a different sparsity ratio, i.e., (R1)–(R5). Vertical bars are the standard errors

**Table 2** CPU time (in seconds) of estimating  $\beta$ , for different combinations of  $n$  and  $p$  under (D1), (R1)

Method	FULL	UNIF	DOUBLY	BLEV	SLEV	IBOSS	OSS	DOPT	CORE
(a) CPU time for different $p$ , with fixed $n = 10^5$									
$p = 50$	0.82	0.00	0.02	0.51	0.51	0.15	0.40	0.43	0.08
$p = 100$	3.19	0.01	0.08	2.02	2.03	0.43	2.26	1.89	0.35
$p = 500$	76.31	0.39	1.61	46.37	46.39	6.53	58.77	45.54	5.29
(b) CPU time for different $n$ , with fixed $p = 100$									
$n = 5 \times 10^4$	1.72	0.00	0.06	1.01	1.01	0.12	1.87	0.89	0.13
$n = 5 \times 10^5$	16.95	0.02	0.10	10.86	10.85	1.36	6.41	9.92	0.87
$n = 5 \times 10^6$	189.65	0.11	0.21	119.83	119.84	12.94	43.28	91.52	7.54

As shown in Figs. 5 and 6, MOM-CORE consistently achieves the smallest estimation and prediction errors among all these subsampling methods, and its advantage becomes more prominent with the increase of sparsity. Such an observation indicates that by integrating with the MOM procedure, our proposed MOM core-elements algorithm leads to an effective and robust estimator. Additionally, it is noteworthy that in both Figs. 3 and 6, the (MOM-)CORE prediction not only outperforms other subsampling approaches but also often delivers results that are nearly indistinguishable from full data.

### 5.3 Computing time

To compare the computational efficiency of these subsampling approaches, we present the CPU time (in seconds) for different combinations of the sample size  $n$  and the dimension  $p$  under the case of (D1), (R1) in Table 2. Here, data corruption and the MOM procedure are omitted to save space, as they hardly affect the computation time. We take the subsample parameter  $r = 10p$ . All computations are implemented using the R programming language on a desktop running Windows 10 with an Intel i5-10210U CPU and 16GB memory. The CPU time for using the full sample is also presented for comparison.

As can be observed from Table 2, all of these estimates are more efficient than the full sample OLS estimate. It is unsurprising that the random sampling-based methods, UNIF and DOUBLY, require the least computing time, as they avoid the additional step of calculating subsampling probabilities. Among these methods, BLEV, SLEV, and DOPT necessitate the calculation of singular value decomposition of the full predictor matrix, leading to relatively longer CPU times. As expected, OSS also requires considerable computing time due to its complexity of  $O(np \log r + rp^2)$ .

Notably, the proposed core-elements approach only requires a longer time than UNIF and DOUBLY, while being faster than other competitors in almost all circumstances. In addition, its superiority in computation becomes more promi-

nent when  $n$  is much larger than  $p$ , which is exactly the most suitable situation for taking advantage of subsampling.

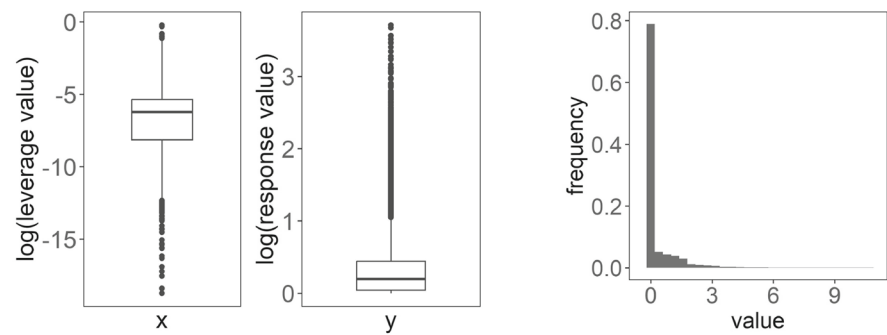
## 6 Real data example

The rapid development of the single-cell RNA sequencing (scRNA-seq) technique enables the gene expression profiling of single cells. ScRNA-seq data are often organized into a read count matrix, where rows are cells, columns represent genes, and the  $(i, j)$ th component is the observed expression level of the gene  $j$  in cell  $i$ . We consider a scRNA-seq dataset collected by Azizi et al. (2018), which includes CD45+ immune cells from eight breast carcinomas, as well as matched normal breast tissue, blood, and lymph node. The dataset is publicly available with the accession code GSE114725 in Gene Expression Omnibus (Edgar et al. 2002). Our goal is to find the relationship between the expression of the gene MT-RNR2 and other genes. As a critical neuroprotective factor, the MT-RNR2 gene encodes the Humanin polypeptide and protects against death in Alzheimer's disease.

To achieve the goal, we take the reads of this gene as the response and the reads for other genes as predictors. Following the data pre-processing steps in Huang et al. (2008), we first screen the genes as follows: (1) select the top 3000 genes with the highest expression levels; and (2) select the top 500 genes with the largest variances. We then standardize the predictors so that they have unit variance. The distribution of pre-processed data points is shown in Fig. 7a. The final sample contains  $n \approx 10^5$  cells and  $p = 500$  genes with over 75% zero elements, as illustrated through the histogram in Fig. 7b. Figure 7c presents the relationship between the response against ten randomly selected predictors of the scRNA-seq dataset, from which we can observe linear patterns. Therefore, it is reasonable to assume the data follow the linear model (1).

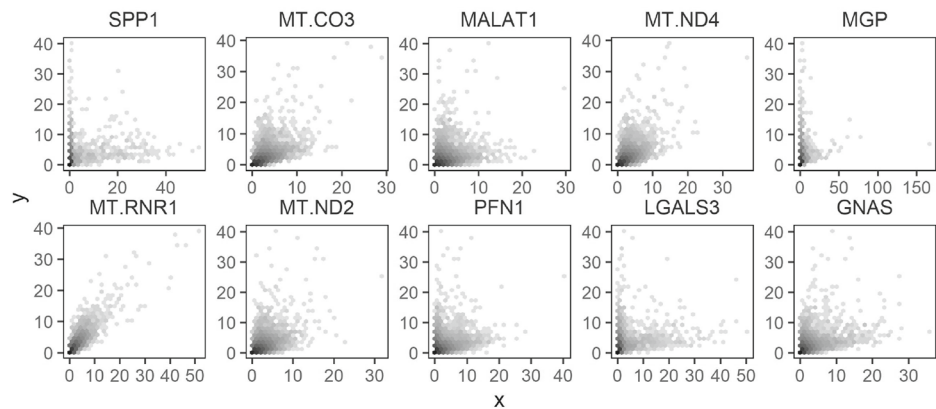
As the true coefficient vector  $\beta$  is unknown in real-world data analysis, we focus on comparing prediction performance by calculating the PMSE as defined in (11). This evalua-

**Fig. 7** Visualization of the scRNA-seq dataset’s distribution. For visual clarity, the leverage and response values are (shifted to the positive range and) log-transformed in the subfigure (a), and the predictor values are truncated at the 99.9% quantile in the subfigure (b)



(a) Box plots of predictor leverage values (left) and response values (right).

(b) Histogram of values in the predictor matrix.

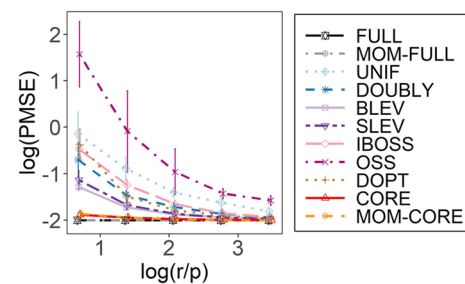


(c) Hexbin scatter plots (Carr et al., 2023) of the response against ten randomly selected predictors.

tion is based on one hundred bootstrap samples. The training and test sets are randomly partitioned according to the ratio of 7:3. The subsampling methods considered here are the same as those in Sect. 5, and the subsample size is set to be  $r \in \{2p, 2^2p, 2^3p, 2^4p, 2^5p\}$  rows or  $s = rp$  elements equivalently. We set the number of blocks for MOM to be  $k = 5$ .

As shown in Fig. 7, the data exhibits linear patterns between the response and predictors, with no apparent outliers. Consequently, the MOM variant performs similarly to its original estimator. For clarity, we present only the MOM-FULL and MOM-CORE, omitting the MOM variants of other competing subsampling methods.

Figure 8 displays the performance of different approaches for predicting  $y$  on the test set. We observe the baseline methods, FULL and MOM-FULL, achieve almost the same prediction accuracy, suggesting the absence of extreme outliers in the dataset. Similarly, the proposed CORE and MOM-CORE methods have nearly identical prediction accuracy, both of which consistently outperform other subsampling approaches. In addition, the proposed estimators perform



**Fig. 8** Comparison of different methods w.r.t. PMSE for the scRNA-seq dataset

almost the same as  $\hat{\beta}_{OLS}$  w.r.t. the PMSE, even when the selected number of elements is just  $s = 2p^2$ .

Table 3 shows the CPU time for these subsampling methods. Compared to the full sample estimate, all the subsample estimates require a shorter computational time, except for the OSS method as  $r$  increases. This exception can be attributed to the multiplicative order  $O(\log r)$  in its computational cost. Similar to the results in Table 2, we observe that the core-

**Table 3** CPU time (in seconds) of estimating  $\beta$  for the scRNA-seq dataset

Method	FULL	UNIF	DOUBLY	BLEV	SLEV	IBOSS	OSS	DOPT	CORE
$r = 2p$	–	0.19	0.23	46.98	46.96	8.27	12.13	43.29	3.24
$r = 2^3 p$	–	0.81	1.16	47.44	47.43	9.81	85.32	44.84	6.05
$r = 2^5 p$	–	3.39	6.43	50.06	50.08	15.33	438.91	47.50	8.53
$n$	80.24	–	–	–	–	–	–	–	–

elements approach has great advantages in computing. It is only slower than the random sampling-based UNIF and DOUBLY methods, while requiring nearly half the CPU time of the IBOSS method. The results in Fig. 8 and Table 3 indicate that the proposed strategy can provide a more effective estimate than the competitors, requiring a relatively short computational time.

## 7 Discussion

Realizing the gaps of element-wise subset selection methods in large-scale data analysis, we developed the core-elements method for least squares estimation in linear models. Theoretically, we showed that the proposed core-elements estimator approximately minimizes an upper bound of the estimation variance. We also provided a coresets-like finite-sample bound for the proposed estimator, which is supported by empirical results. To deal with data corruption, we introduced the median-of-means estimation to provide a robust version of core-elements and established consistency of the resultant estimator. Empirical studies suggest that the proposed method is not only suitable for (numerically) sparse matrices but also has a superior performance for more general dense cases in terms of both accuracy and time.

Considering that the predictor matrix defined by basis function evaluations usually enjoys the (numerically) sparse property, we plan to extend core-elements to nonparametric additive models for efficiently approximating the penalized least squares estimation in the future. More importantly, the core-elements approach not only facilitates computation, but also has excellent applications in preserving data privacy and improving communication efficiency in federated learning, which are also left to our future work.

**Supplementary Information** The online version contains supplementary material available at <https://doi.org/10.1007/s11222-024-10505-6>.

**Author Contributions** M.L. developed and implemented the algorithm; drafted the manuscript. J.Y. analyzed the theoretical properties; revised the manuscript. T.L. analyzed and interpreted the data. C.M. conceptualized and designed the work; revised the manuscript. All authors reviewed the manuscript and approved the version to be published.

**Funding** This work was supported by Beijing Municipal Natural Science Foundation Grant No. 1232019, National Natural Science Foun-

ation of China Grant No. 12101606, No. 12471244, No. 12271522, Renmin University of China research fund program for young scholars, and Beijing Institute of Technology research fund program for young scholars.

**Data availability** No datasets were generated or analysed during the current study.

## Declarations

**Conflict of interest** The authors declare no Conflict of interest.

## References

- Azizi, E., Carr, A.J., Plitas, G., Cornish, A.E., Konopacki, C., Prabhakaran, S., Nainys, J., Wu, K., Kiseliovas, V., Setty, M.: Single-cell map of diverse immune phenotypes in the breast tumor microenvironment. *Cell* **174**(5), 1293–1308 (2018)
- Arora, S., Hazan, E., Kale, S.: Fast algorithms for approximate semidefinite programming using the multiplicative weights update method. In: *46th Annual IEEE Symposium on Foundations of Computer Science*, pp. 339–348. IEEE (2005)
- Arora, S., Hazan, E., Kale, S.: A fast random sampling algorithm for sparsifying matrices. In: Dinur, I., Jansen, K., Naor, S. (eds.) *Approximation, randomization, and combinatorial optimization: algorithms and techniques*, pp. 272–279. Springer, Berlin (2006)
- Achlioptas, D., Karnin, Z.S., Liberty, E.: Near-optimal entrywise sampling for data matrices. *Adv. Neural. Inf. Process. Syst.* **26**, 1565–1573 (2013)
- Andrews, T.S., Kiselev, V.Y., McCarthy, D., Hemberg, M.: Tutorial: guidelines for the computational analysis of single-cell RNA sequencing data. *Nat. Protoc.* **16**(1), 1–9 (2021)
- Achlioptas, D., Mcsherry, F.: Fast computation of low-rank matrix approximations. *J. Assoc. Comput. Mach.* **54**(2), 1–19 (2007)
- Alaoui, A., Mahoney, M.W.: Fast randomized kernel ridge regression with statistical guarantees. *Adv. Neural. Inf. Process. Syst.* **28**, 775–783 (2015)
- Ai, M., Wang, F., Yu, J., Zhang, H.: Optimal subsampling for large-scale quantile regression. *J. Complex.* **62**, 101512 (2020)
- Ai, M., Yu, J., Zhang, H., Wang, H.: Optimal subsampling algorithms for big data regressions. *Stat. Sin.* **31**, 749–772 (2021)
- Boutsidis, C., Drineas, P., Magdon-Ismael, M.: Near-optimal coresets for least-squares regression. *IEEE Trans. Inf. Theory* **59**(10), 6880–6892 (2013)
- Braverman, V., Krauthgamer, R., Krishnan, A.R., Sapir, S.: Near-optimal entrywise sampling of numerically sparse matrices. In: *Proceedings of Thirty Fourth Conference on Learning Theory*, vol. 134, pp. 759–773. PMLR (2021)
- Chen, Y., Bhojanapalli, S., Sanghavi, S., Ward, R.: Coherent matrix completion. In: *International Conference on Machine Learning*, pp. 674–682. PMLR (2014)

- Cai, J.-F., Candès, E.J., Shen, Z.: A singular value thresholding algorithm for matrix completion. *SIAM J. Optim.* **20**(4), 1956–1982 (2010)
- Chang, M.-C.: Predictive subdata selection for computer models. *J. Comput. Graph. Stat.* **32**, 1–18 (2022)
- Carmon, Y., Jin, Y., Sidford, A., Tian, K.: Coordinate methods for matrix games. In: *2020 IEEE 61st Annual Symposium on Foundations of Computer Science*, pp. 283–293. IEEE (2020)
- Chasiotis, V., Karlis, D.: Subdata selection for big data regression: an improved approach. *J. Data Sci. Stat. Visual.* (2024). <https://doi.org/10.52933/jdssv.v4i3.78>
- Carr, D., Lewin-Koh, N., Maechler, M., Sarkar, D.: hexbin: Hexagonal binning routines. R package version 1.28.3 (2023)
- Candès, E.J., Recht, B.: Exact matrix completion via convex optimization. *Found. Comput. Math.* **9**(6), 717–772 (2009)
- Candès, E.J., Tao, T.: The power of convex relaxation: near-optimal matrix completion. *IEEE Trans. Inf. Theory* **56**(5), 2053–2080 (2010)
- Dasgupta, A., Drineas, P., Harb, B., Kumar, R., Mahoney, M.W.: Sampling algorithms and coresets for  $l_p$  regression. *SIAM J. Comput.* **38**(5), 2060–2078 (2009)
- Donoho, D.L., Gasko, M.: Breakdown properties of location estimates based on halfspace depth and projected outlyingness. *Ann. Stat.* **20**(4), 1803–1827 (1992)
- Donoho, D.L., Huber, P.J.: The notion of breakdown point. *A Festschrift for Erich L. Lehmann* (1983)
- Davis, T.A., Hu, Y.: The University of Florida sparse matrix collection. *ACM Trans. Math. Softw.* **38**(1), 1–25 (2011)
- Drineas, P., Kannan, R., Mahoney, M.W.: Fast Monte Carlo algorithms for matrices I: Approximating matrix multiplication. *SIAM J. Comput.* **36**(1), 132–157 (2006)
- Drineas, P., Magdon-Ismaïl, M., Mahoney, M.W., Woodruff, D.P.: Fast approximation of matrix coherence and statistical leverage. *J. Mach. Learn. Res.* **13**(1), 3475–3506 (2012)
- Dai, W., Song, Y., Wang, D.: A subsampling method for regression problems based on minimum energy criterion. *Technometrics* **65**(2), 192–205 (2023)
- Dereziński, M., Warmuth, M.K., Hsu, D.J.: Leveraged volume sampling for linear regression. *Adv. Neural. Inf. Process. Syst.* **31**, 2510–2519 (2018)
- Drineas, P., Zouzias, A.: A note on element-wise matrix sparsification via a matrix-valued Bernstein inequality. *Inf. Process. Lett.* **111**(8), 385–389 (2011)
- d’Aspremont, A.: Subsampling algorithms for semidefinite programming. *Stoch. Syst.* **1**(2), 274–305 (2011)
- Edgar, R., Domrachev, M., Lash, A.E.: Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Res.* **30**(1), 207–210 (2002)
- El Karoui, N., d’Aspremont, A.: Second order accurate distributed eigenvector computation for extremely large matrices. *Electron. J. Stat.* **4**, 1345–1385 (2010)
- Feldman, D., Schmidt, M., Sohler, C.: Turning big data into tiny data: Constant-size coresets for  $k$ -means, PCA, and projective clustering. *SIAM J. Comput.* **49**(3), 601–657 (2020)
- Garber, D., Hazan, E.: Sublinear time algorithms for approximate semidefinite programming. *Math. Program.* **158**(1), 329–361 (2016)
- Gupta, N., Sidford, A.: Exploiting numerical sparsity for efficient learning: faster eigenvector computation and regression. *Adv. Neural. Inf. Process. Syst.* **31**, 5274–5283 (2018)
- Hampel, F.R.: *Contribut. Theory Robust Estim.* University of California, Berkeley (1968)
- Higham, N.J.: *Functions of Matrices: Theory and Computation*. SIAM, Philadelphia (2008)
- Huang, S.-T., Lederer, J.: Deepmom: Robust deep learning with median-of-means. *J. Comput. Graph. Stat.* **32**(1), 181–195 (2023)
- Hou-Liu, J., Browne, R.P.: Generalized linear models for massive data via doubly-sketching. *Stat. Comput.* **33**(5), 105 (2023)
- Hastie, T., Mazumder, R., Lee, J.D., Zadeh, R.: Matrix completion and low-rank SVD via fast alternating least squares. *J. Mach. Learn. Res.* **16**(1), 3367–3402 (2015)
- Huang, J., Ma, S., Zhang, C.H.: Adaptive Lasso for sparse high-dimensional regression models. *Stat. Sin.* **18**(4), 1603–1618 (2008)
- Hsu, D., Sabato, S.: Heavy-tailed regression with a generalized median-of-means. In: *International Conference on Machine Learning*, pp. 37–45. PMLR (2014)
- Joseph, V.R., Mak, S.: Supervised compression of big data. *Stat. Anal. Data Min.: ASA Data Sci. J.* **14**(3), 217–229 (2021)
- Joseph, V.R., Vakayil, A.: SPlit: an optimal method for data splitting. *Technometrics* **64**(2), 166–176 (2022)
- Kundu, A., Drineas, P., Magdon-Ismaïl, M.: Recovering PCA and sparse PCA via hybrid- $(\ell_1, \ell_2)$  sparse sampling of data elements. *J. Mach. Learn. Res.* **18**(1), 2558–2591 (2017)
- Kairouz, P., McMahan, H.B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A.N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al.: Advances and open problems in federated learning. *Found. Trends Mach. Learn.* **14**(1–2), 1–210 (2021)
- Konečný, J., McMahan, H.B., Yu, F.X., Richtárik, P., Suresh, A.T., Bacon, D.: Federated learning: Strategies for improving communication efficiency. In: *NIPS Workshop on Private Multi-Party Machine Learning* (2016)
- Knight, K.: Subsampling least squares and elemental estimation. In: *2018 IEEE Data Science Workshop (DSW)*, pp. 91–94. IEEE (2018)
- Lecué, G., Lerasle, M.: Learning from MOM’s principles: Le Cam’s approach. *Stoch. Process. Appl.* **129**(11), 4385–4410 (2019)
- Lecué, G., Lerasle, M.: Robust machine learning by median-of-means: theory and practice. *Ann. Stat.* **48**(2), 906–931 (2020)
- Lugosi, G., Mendelson, S.: Regularization, sparse recovery, and median-of-means tournaments. *Bernoulli* **25**(3), 2075–2106 (2019)
- Li, T., Meng, C.: Modern subsampling methods for large-scale least squares regression. *Int. J. Cyber-Phys. Syst.* **2**(2), 1–28 (2021)
- Li, F., Xie, R., Wang, Z., Guo, L., Ye, J., Ma, P., Song, W.: Online distributed IoT security monitoring with multidimensional streaming big data. *IEEE Internet Things J.* **7**(5), 4387–4394 (2019)
- Li, M., Yu, J., Li, T., Meng, C.: Importance sparsification for Sinkhorn algorithm. *J. Mach. Learn. Res.* **24**, 1–44 (2023)
- Liu, F., Zhu, B., Yuan, S., Li, J., Xue, K.: Privacy-preserving truth discovery for sparse data in mobile crowdsensing systems. In: *2021 IEEE Global Communications Conference*, pp. 1–6. IEEE (2021)
- Mahoney, M.W.: Lecture notes on randomized linear algebra. arXiv preprint [arXiv:1608.04481](https://arxiv.org/abs/1608.04481) (2016)
- Martinez, C.: Partial quicksort. In: *Proc. 6th ACM-SIAM Workshop on Algorithm Engineering and Experiments and 1st ACM-SIAM Workshop on Analytic Algorithmics and Combinatorics*, pp. 224–228 (2004)
- Mathieu, T.: M-estimation and median of means applied to statistical learning. In: *PhD thesis*, Université Paris-Saclay (2021)
- Maalouf, A., Eini, G., Mussay, B., Feldman, D., Osadchy, M.: A unified approach to coreset learning. *IEEE Trans. Neural Netw. Learn. Syst.* **35**, 1–13 (2022)
- Ma, P., Huang, J.Z., Zhang, N.: Efficient computation of smoothing splines via adaptive basis sampling. *Biometrika* **102**(3), 631–645 (2015)
- Mak, S., Joseph, V.R.: Support points. *Ann. Stat.* **46**(6A), 2562–2592 (2018)
- Muzellec, B., Josse, J., Boyer, C., Cuturi, M.: Missing data imputation using optimal transport. In: *International Conference on Machine Learning*, pp. 7130–7140. PMLR (2020)
- Ma, P., Mahoney, M.W., Yu, B.: A statistical perspective on algorithmic leveraging. *J. Mach. Learn. Res.* **16**(1), 861–911 (2015)

- Ma, P., Sun, X.: Leveraging for big data regression. *Wiley Interdiscip. Rev.: Comput. Stat.* **7**(1), 70–76 (2015)
- Munteanu, A., Schwiegelshohn, C., Sohler, C., Woodruff, D.P.: On core-sets for logistic regression. *Adv. Neural. Inf. Process. Syst.* **31**, 6562–6571 (2018)
- Musser, D.R.: Introspective sorting and selection algorithms. *Softw.: Pract. Exp.* **27**(8), 983–993 (1997)
- Meng, C., Wang, Y., Zhang, X., Mandal, A., Zhong, W., Ma, P.: Effective statistical methods for big data analytics. In: *Handbook of Research on Applied Cybernetics and Systems Science*, pp. 280–299. IGI Global, (2017)
- Meng, C., Xie, R., Mandal, A., Zhang, X., Zhong, W., Ma, P.: Low-con: A design-based subsampling approach in a misspecified linear model. *J. Comput. Graph. Stat.* **30**(3), 694–708 (2021)
- Meng, C., Yu, J., Chen, Y., Zhong, W., Ma, P.: Smoothing splines approximation using Hilbert curve basis selection. *J. Comput. Graph. Stat.* **31**, 1–11 (2022)
- Ma, P., Zhang, X., Xing, X., Ma, J., Mahoney, M.: Asymptotic analysis of sampling estimators for randomized numerical linear algebra algorithms. In: *International Conference on Artificial Intelligence and Statistics*, pp. 1026–1035. PMLR (2020)
- Meng, C., Zhang, X., Zhang, J., Zhong, W., Ma, P.: More efficient approximation of smoothing splines via space-filling basis selection. *Biometrika* **107**, 723–735 (2020)
- Nguyen, T.K.H., Berge, K., Chiogna, M., Risso, D.: Structure learning for zero-inflated counts with an application to single-cell RNA sequencing data. *Ann. Appl. Stat.* **17**(3), 2555–2573 (2023)
- Qaiser, S., Ali, R.: Text mining: Use of TF-IDF to examine the relevance of words to documents. *Int. J. Comput. Appl.* **181**(1), 25–29 (2018)
- Reuter, T., Schwabe, R.: D-optimal subsampling design for massive data linear regression. *arXiv preprint arXiv:2307.02236* (2024)
- Settles, B.: Active learning. *Synth. Lect. Artif. Int. Mach. Learn.* **6**(1), 1–114 (2012)
- Sun, X., Zhong, W., Ma, P.: An asymptotic and empirical smoothing parameters selection method for smoothing spline anova models in large samples. *Biometrika* **108**(1), 149–166 (2021)
- Van Buuren, S., Groothuis-Oudshoorn, K.: MICE: Multivariate imputation by chained equations in R. *J. Stat. Softw.* **45**, 1–67 (2011)
- Vakayil, A., Joseph, V.R.: Data twinning. *Stat. Anal. Data Min.: ASA Data Sci. J.* **15**(5), 598–610 (2022)
- Wang, L., Elmstedt, J., Wong, W.K., Xu, H.: Orthogonal subsampling for big data linear regression. *Ann. Appl. Stat.* **15**(3), 1273–1290 (2021)
- Wang, S., Gittens, A., Mahoney, M.W.: Scalable kernel K-means clustering with Nyström approximation: relative-error bounds. *J. Mach. Learn. Res.* **20**(1), 431–479 (2019)
- Wang, R., Ouyang, Y., Panpan, Y., Xu, W.: A fast and accurate estimator for large scale linear model via data averaging. *Adv. Neural Inf. Process. Syst.* **36**, 34917–34927 (2023)
- Wang, H., Yang, M., Stufken, J.: Information-based optimal sub-data selection for big data linear regression. *J. Am. Stat. Assoc.* **114**(525), 393–405 (2019)
- Wang, S., Zhang, Z.: Improving CUR matrix decomposition and the Nyström approximation via adaptive sampling. *J. Mach. Learn. Res.* **14**(1), 2729–2769 (2013)
- Wang, H., Zhu, R., Ma, P.: Optimal subsampling for large sample logistic regression. *J. Am. Stat. Assoc.* **113**(522), 829–844 (2018)
- Wang, J., Zou, J., Wang, H.: Sampling with replacement vs Poisson sampling: a comparative study in optimal subsampling. *IEEE Trans. Inf. Theory* **68**(10), 6605–6630 (2022)
- Xie, R., Wang, Z., Bai, S., Ma, P., Zhong, W.: Online decentralized leverage score sampling for streaming multidimensional time series. In: *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 2301–2311 (2019)
- Yu, J., Ai, M., Ye, Z.: A review on design inspired subsampling for big data. *Stat. Papers* **65**, 1–44 (2023)
- Yu, J., Wang, H., Ai, M., Zhang, H.: Optimal distributed subsampling for maximum quasi-likelihood estimators with massive data. *J. Am. Stat. Assoc.* **117**(537), 265–276 (2022)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.