Statistical Analysis and Data Mining: The ASA Data Science Journal

WILEY

RESEARCH ARTICLE OPEN ACCESS

An Efficient Filtering Approach for Model Estimation in Sparse Regression

Yanran Wei[1] | William Myers[2] | Xinwei Deng[1]

[1]Department of Statistics, Virginia Tech, Blacksburg, Virginia, USA | [2]Farmer School of Business, Miami University, Oxford, Ohio, USA

Correspondence: Xinwei Deng (xdeng@vt.edu)

Received: 31 March 2024 | **Revised:** 12 September 2024 | **Accepted:** 14 September 2024

Funding: This work was supported by a Research Collaboration Program between Procter & Gamble Co. and Virginia Tech.

Keywords: data reduction | data structure | filtering approach | sparse regression

ABSTRACT

As technology advances, the scale of data generated is growing exponentially, bringing huge challenges to data storage and computation. To facilitate the computational cost while maintaining model estimation accuracy, subdata selection become important. Conventional methods, such as LASSO and ridge regression, often focus on feature selection. In contrast, methods of subsampling aim at specifying data points to be extracted. However, these subsampling methods often overlook the full consideration of the role of the response variable and its relationship with predictor variables. In this work, we propose a so-called filtering approach for model estimation (FAME) method to perform subsampling in combination with feature screening. Compared with existing methods, the proposed method can result in the subdata being smaller in size both in terms of the number of features and observations, and also the computational complexity does not increase. The proposed method can be extended to situations when the predictor is binary, the response is binary, or both are binary. The performance of FAME is evaluated in both numerical studies and real data examples.

1 | Introduction

As modern technologies advance, the capability of collecting large datasets is often beyond the analysis ability of traditional statistical methods. For example, Walmart processes over 40 Petabytes of data with millions of rows per day, aiming to obtain valuable information on customer needs. Extracting valuable information from such large-scale data is challenging, and it is beyond the capability of conventional statistical methods to deal with these massive datasets directly. There is an emerging need to conduct data reduction for such large datasets and enable efficient data analysis.

Without loss of generality, this paper concerns the dataset with an $n \times 1$ response vector and an $n \times p$ predictor matrix. Here, a large dataset can be referred to as the situation that both n and p can be large. In the literature, there are two types of common approaches for data reduction. The first common method is subsampling of data points to get a subdata with n^* data points, where $n^* \ll n$. For example, Drineas et al. [1] developed algorithms based on statistical leverage scores of matrices. Ma, Mahoney, and Yu [2] proposed two new leveraging algorithms: shrinked leveraging estimator and unweighted leveraging estimator. The algorithmic leveraging method takes at least $O\left(np \log n/\epsilon^2\right)$ ($\epsilon \in (0,\ 0.5]$) time, and $O\left(np^2\right)$ for appropriate parameter setting. Besides subsampling-based methods, Wang, Yang, and Stufken [3] developed information-based optimal subdata selection (IBOSS) method selecting data points based on a certain optimal design criterion. Deldossi and Tommasi [4] proposed the method "Optimal Design Based" for observation selection based on any optimality criterion. Xie, Bai, and Ma

[5] proposed data reduction methods on D-optimality improving the computational efficiency of the online analysis. Meng et al. [6] used orthogonal Latin hypercube designs for robust regression estimation even when the underlying linear model is misspecified. Singh and Stufken [7] proposed the Combining Lasso and Subdata Selection method combining multiple LASSO and IBOSS for subdata selection. More work is extended based on IBOSS, like Cheng, Wang, and Yang [8] applied subdata selection on logistic regression models and Ai et al. [9] proposed the subsampling method for quantile regression.

Considering the data with a large dimension of predictor variables, the second common method for data reduction is to get a subdata with p^* predictors where $p^* \ll p$. Various variable selection methods have been proposed, including LASSO [10], SCAD [11], and Dantzig [12], among many others. In particular, Fan and Lv [13] proposed the sure independence screening (SIS) method to reduce the dimensionality of predictor variables with fast computation. Wang [14] proposed FP-SIS for ultrahigh-dimensional variable selection by factor analysis. Zhao et al. [15] extended SIS to a so-called preconditioned profiled independence screening (PPIS) method that can perform consistent model selection for spiked populations. Note that the variable selection approaches do not focus on the reduction of data points, while the subsampling approaches often overlook the problem of the large dimensionality of predictor variables. Moreover, the subsampling approaches based on optimal design criteria often fail to take full advantage of the information on the response variables.

In this work, we proposed a so-called filtering approach for model estimation (FAME), which efficiently selects subdata for estimating regression models. The key idea of the proposed method is to consider data reduction on both predictor variables and data points with fast computation. Specifically, we adopt the sure independence screening to effectively reduce the number of predictor variables in the subdata, and then use the information-based optimal subdata selection to reduce the number of data points in the subdata. The proposed FAME approach has several advantages over other methods. First, the dataset selected by the proposed FAME method is much smaller in both size and dimension than the original data. While the model estimation accuracy based on the reduced data is comparable to that based on the original data. Second, the computation time of the proposed FAME method is in the order of $O(np)$. It implies that there is no extra computation cost compared with other methods. Third, we consider both the response variable and predictor variable in the reduction of data points, which is different from the optimal design-based subdata selection. Finally, although this paper focuses on both the response variable and predictor variable to be continuous, the proposed FAME method can be easily extended to the case when the response variable or predictor variable is discrete.

The remainder of the paper is organized as follows. In Section 2, we present the proposed FAME method for continuous data and derive the statistical properties of the proposed FAME estimators. We extend it to situations with binary predictor variables or binary response variables. The proposed FAME method is evaluated using simulation data in Section 3 and is also examined in real data in Section 4. In Section 5, we conclude the paper with discussions on further research directions.

2 | The Proposed Methodology

In this section, we will detail the proposed method. Suppose that there are p predictor variables x_j, $j = 1, \dots, p$ and the corresponding response variable y. Let $\boldsymbol{D} = (\boldsymbol{y}, \boldsymbol{X})$ denote the full dataset, where $\boldsymbol{y} = (y_1, y_2, \dots, y_n)^T$ is an n-vector continuous response variable, and $\boldsymbol{X} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)^T$ is an n by p input matrix, where $\boldsymbol{x}_i \in \mathcal{R}^p$ for $i = 1, \dots, n$. We consider the linear regression model for $\boldsymbol{y}$ and $\boldsymbol{X}$ as

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{1}$$

where $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_p)$ is the parameter vector, and $\boldsymbol{\epsilon} = (\epsilon_1, \dots, \epsilon_n)^T$ is the error vector with $\mathrm{Var}(\boldsymbol{\epsilon}) = \sigma^2 I$. When $n > p$, the estimation of parameter $\boldsymbol{\beta}$ depends on the matrix inversion $(\boldsymbol{X}^T\boldsymbol{X})^{-1}$, which can be expensive when the dimensionality p is large. In the high-dimensional setting with $p > n$, the regularization methods, such as LASSO, are often used for model estimation. It is known that the computational complexity of LASSO [16] is closely related to both the sample size n and dimensionality p. Moreover, when the sample size n is large, the data could contain redundant information in rows, which may not contribute much to the inference and estimation of $\boldsymbol{\beta}$. To overcome the above challenges, we consider an efficient data reduction in terms of reducing both the rows and columns of the original data for model estimation in sparse regression. We will start with linear models with continuous inputs in Section 2.1 and extend the proposed method for generalized linear models in Section 2.3.

2.1 | The FAME Method

To efficiently conduct data reduction, we propose an FAME in the sparse regression setting. The key idea behind the proposed FAME method is to select a subset of columns and a subset of rows based on data filtering to form the reduced dataset for model estimation. Here, the data filtering aims to filter out (i.e., remove) relatively unimportant columns or rows such that the reduced dataset contains key information with a smaller size than the original dataset.

Specifically, we first conduct a feature screening to select important columns. Denote $\boldsymbol{x}^{(j)}$ to be the jth column of input matrix $\boldsymbol{X}$ and recall that the response vector is $\boldsymbol{y}$. Without loss of generality, we assume each column of $\boldsymbol{X}$ is standardized. Motivated by the sure independent screening in Fan and Lv [13], we consider to filter out features based on each individual column's index score as $w_j = g(\boldsymbol{x}^{(j)}, \boldsymbol{y})$. Thus, we can keep a subset of columns with the highest scores. Here, the score index measures the importance of feature $\boldsymbol{x}^{(j)}$ by the degree of correlation between $\boldsymbol{x}^{(j)}$ and the response variable y for $j = 1, \dots, p$. For example, the index score w_j can be the absolute value of the regression coefficient derived from the marginal regression of $\boldsymbol{y}$ on $\boldsymbol{x}^{(j)}$ as shown in Fan and Lv [13]. Then, we can define a subset of predictors as

$$\boldsymbol{M}_S = \{1 \le q \le h \le p : w_q \text{ is among the } h \text{ largest values in } \boldsymbol{w} = (w_1, \dots, w_p)^T\} \tag{2}$$

Correspondingly, the reduced input matrix consists of $\boldsymbol{X}_S = (\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(q)}, \dots, \boldsymbol{x}^{(h)})$, where $\boldsymbol{x}^{(q)}$ is the column whose score

index belongs to $\boldsymbol{M}_S$. Then, the complete data $\boldsymbol{D}$ are reduced to a subset with h columns as $\boldsymbol{D}_S = (\boldsymbol{X}_S, \boldsymbol{y})$.

Although the reduction in the number of predictors will address the computational challenges of model estimation to some extent, there may be redundant information among data points for the big dataset with large size of rows. Thus, in the second step, we filter out relatively unimportant rows in the dataset $\boldsymbol{D}_S$. Specifically, the proposed FAME method selects data points with the aim of maximizing the Fisher information matrix based on the model in Equation (1). It is known that the Fisher information matrix based on the model in Equation (1) for the reduced dataset $\boldsymbol{D}_S$ can be written as $\boldsymbol{I}(\delta) = \frac{1}{\sigma^2}\sum_{i=1}^{n}\delta_i \boldsymbol{x}_{is}\boldsymbol{x}_{is}^T$, where $\boldsymbol{x}_{is} \in \mathcal{R}^h$ is the data point in $\boldsymbol{D}_S$ and $\delta_i = 1$ means the ith row in $\boldsymbol{D}_S$ is selected, otherwise not. Therefore, one could consider to select a subset of rows with size k based on the following criteria:

$$\max_{\delta=(\delta_1,\ldots,\delta_n),\delta_i\in\{0,1\}} \left|\sum_{i=1}^{n}\delta_i \boldsymbol{x}_{is}\boldsymbol{x}_{is}^T\right|, \quad \text{s.t.} \quad \sum_{i=1}^{n}\delta_i = k \tag{3}$$

which is also known as the D-optimality criterion in the design of experiment literature Kiefer [17]. However, the optimization in Equation (3) is nontrivial due to the discrete nature of the optimization. To address this issue, we consider an upper bound of the determinant of $\boldsymbol{I}(\delta)$ such that the selection of rows becomes computationally efficient. Motivated by Wang, Yang, and Stufken [3], we consider one reasonable upper bound for $|I(\delta)|$ to be,

$$|\boldsymbol{I}(\delta)| \le C\frac{k^{h+1}l^{2h}}{4^h\sigma^{2(h+1)}}\prod_{q=1}^{h}\left(\overline{x}_{(u)q} - \overline{x}_{(v)q}\right)^2, \quad q = 1, \ldots, h \tag{4}$$

where $\overline{x}_{(u)q} = \frac{1}{l}\left(x_{(n-l+1)q} + \cdots + x_{(n)q}\right)$ is the average of the l largest order statistic in $\boldsymbol{x}^{(q)}$. Here, $x_{(1)q} \le x_{(2)q} \le \cdots \le x_{(n)q}$ is the order statistic of the qth column $\boldsymbol{x}^{(q)}$. Similarly, $\overline{x}_{(v)q} = \frac{1}{l}\left(x_{(1)q} + \cdots + x_{(l)q}\right)$ is the average of the l smallest order statistic in $\boldsymbol{x}^{(q)}$. Note that the original upper bound in Wang, Yang, and Stufken [3] is a special case of the upper bound in Equation (4). By considering the average of the upper-order statistics and the average of the lower-order statistic of $\boldsymbol{x}^{(q)}$, it would help the selection of data points robust against the potential outliers and uncertainty on the extreme observations. Thus, we consider the proposed FAME method of selecting a subset of rows as

$$\max_{\delta=(\delta_1,\ldots,\delta_n),\delta_i\in\{0,1\}} \sum_{q=1}^{h}\delta_i \log\left[\overline{x}_{(u)q} - \overline{x}_{(v)q}\right], \quad \text{s.t.} \quad \sum_{i=1}^{n}\delta_i = k \tag{5}$$

It is seen that the above optimization can be decomposed into the selection of rows based on each column, which is computationally fast. For practical implementation, we sort the predictor values on the qth column $\boldsymbol{x}^{(q)}$ and select the $2l$ data points having the l largest and the l smallest order statistics. Repeat such a procedure for each column in $\boldsymbol{D}_S$, we compose the selected subset of data as $\tilde{\boldsymbol{D}}_S = (\tilde{\boldsymbol{X}}_S, \tilde{\boldsymbol{y}})$. Note that the reduced input matrix $\tilde{\boldsymbol{X}}_S$ is a $k \times h$ matrix. The choice of l and h is closely related to the size of the filtered data, $k = 2lh$, which will be discussed at the end of this subsection.

With the filtered data $\tilde{\boldsymbol{D}}_S$, we consider the model estimation by using the LASSO method [10] as

$$\tilde{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}}(\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{X}}_S\boldsymbol{\beta})^T(\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{X}}_S\boldsymbol{\beta}) + \lambda\|\boldsymbol{\beta}\|_1 \tag{6}$$

where $\lambda \ge 0$ is a tuning parameter and $\|\boldsymbol{\beta}\|_1 = \sum_{j=1}^{h}|\boldsymbol{\beta}_j|$ is the l_1 penalty. The optimal value of λ is chosen using cross-validation. Algorithm 1 summarizes the proposed FAME algorithm for sparse regression as follows.

ALGORITHM 1 | (FAME for sparse regression).

Step 1: For $1 \le j \le p$, calculate the score index $w_j = g(\boldsymbol{x}^{(j)}, \boldsymbol{y})$;

Step 2: Form $\boldsymbol{D}_S = (\boldsymbol{X}_S, \boldsymbol{y})$, where $\boldsymbol{X}_S$ includes the h columns with the h largest w_j values;

Step 3: For $q = 1, \ldots, h$, based on column $x^{(q)}$ in $\boldsymbol{X}_S$, form $\tilde{\boldsymbol{D}}_S = (\tilde{\boldsymbol{X}}_S, \tilde{\boldsymbol{y}})$ by including $2l$ data points with the l smallest values and l largest values but excluding data points already been selected;

Step 4: Obtain parameter estimation from Equation (6) using the data $\tilde{\boldsymbol{D}}_S$.

One can see that the computation complexity of Algorithm 1 is on the order of $O(np)$, which can be much faster than the Lasso using the original dataset. There are several remarks on the specification of Algorithm 1. First, the index score w_j measures the relationship between $\boldsymbol{y}$ and $\boldsymbol{x}^{(j)}$. It can be the Pearson correlation, partial correlation, or other marginal statistics, like t-statistics and p value. Based on the empirical study, we recommend using t-statistic providing robust results from Algorithm 1 compared with Pearson correlation and p value. Details will be discussed in the experimental section.

Here, we would like to remark that the choice of k, the size of reduced data, is often constrained by the practical consideration. Considering the selected columns have the most strong score index explaining the response variables, and the rule of thumb based on our empirical study (see a numerical study in Section 3) is to select 10–30 points from each column.

For the choice of h, the number of important features to be retained in the filtered data, one may follow the suggestion given by Fan and Lv [13] with $h = n/\log(n)$ when $n < p$ and $h = p/\log(p)$ when $n > p$. However, when both n and p are large, such a choice of h can also be large. Alternatively, one can determine h by using the change-point detection method based on the score index values $w_j = g(\boldsymbol{x}^{(j)}, \boldsymbol{y})$, $j = 1, \ldots, p$. The change-point detection method is to find the first point when the distribution of the descending-ordered score index w_j changes. Here, we adopt the pruned exact linear time (PELT) change-point detection algorithm in Killick, Fearnhead, and Eckley [18]. Specifically, suppose that there are a sorted score index $(w^{(1)}, \ldots, w^{(p)})$ and τ change-points splitting the data into $\tau + 1$ segments. Denote the τ change-point locations as $L = (l_1, \ldots, l_\tau)$. Then, the jth segment contains all score index between location $l_{j-1} + 1$ and l_j, denoted as $\boldsymbol{w}^{(l_{j-1}+1):l_j}$. A change point is detected if we can detect a point location

j that satisfies,

$$C\left(\boldsymbol{w}^{(t+1):j}\right) + C\left(\boldsymbol{w}^{(j+1):T}\right) + K \leq C\left(\boldsymbol{w}^{(t+1):T}\right), \quad \text{for all } t < j < T \tag{7}$$

where $C(\cdot)$ is a cost function for a segment, and K is a constant. The settings of $C(\cdot)$ and K follow the algorithm in Killick, Fearnhead, and Eckley [18] and the corresponding R package *changepoint*.

2.2 | Theoretical Properties

In this section, we investigate the theoretical properties of the FAME algorithm. These properties evaluate our proposed method and give bounds on variances of estimators as well as asymptotic results.

Theorem 1. *Let $\tilde{\boldsymbol{D}}_S$ be the subdata selected by the FAME algorithm including h columns and k data entries in total. Denote $\lambda_{min}(\boldsymbol{R})$ as the smallest eigenvalues of the sample correlation matrix $\boldsymbol{R}$ of $\tilde{\boldsymbol{D}}_S$. The determinant $|\tilde{\boldsymbol{X}}_{\boldsymbol{S}}^T\tilde{\boldsymbol{X}}_{\boldsymbol{S}}|$ satisfies*

$$\frac{|\tilde{\boldsymbol{X}}_{\boldsymbol{S}}^T\tilde{\boldsymbol{X}}_{\boldsymbol{S}}|}{\frac{k^{h+1}}{4^h}\prod_{q=1}^{h}\left(\overline{x}_{(u)q} - \overline{x}_{(v)q}\right)^2} \geq \frac{\lambda_{\min}^h(\boldsymbol{R})}{h^h} \times \prod_{q=1}^{h}\left(\frac{x_{(n-l+1)q} - x_{(l)q}}{\overline{x}_{(u)q} - \overline{x}_{(v)q}}\right)^2 \tag{8}$$

where $x_{(n-l+1)q}$ is the lst largest order statistic and $x_{(l)q}$ is the lst smallest order statistics in $\boldsymbol{x}^{(q)}$. $\overline{x}_{(u)q}$ is the average of the top l order statistic, and $\overline{x}_{(v)q}$ is the l smallest order statistic.

When $\lim_{n\to\infty}\ \lambda_{min}(\boldsymbol{R}) > 0$, then under suitable assumptions, the lower bound of Equation (8) will not converge to 0 as $n \to \infty$. $|\tilde{\boldsymbol{X}}_{\boldsymbol{S}}^T\tilde{\boldsymbol{X}}_{\boldsymbol{S}}|$ is of the same order as the upper bound for $|\boldsymbol{I}(\delta)|$ in Equation (4), although the upper bound is hard to achieve.

Following Tibshirani [10], one can have approximations of the lasso estimator from Equation (6) as

$$\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S \approx \left[\tilde{\boldsymbol{X}}_{\boldsymbol{S}}^T\tilde{\boldsymbol{X}}_{\boldsymbol{S}} + \lambda\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right]^{-1}\tilde{\boldsymbol{X}}_{\boldsymbol{S}}^T\tilde{\boldsymbol{y}} \text{ with}$$

$$\left\{\boldsymbol{\Psi}(\tilde{\boldsymbol{\beta}}(\lambda)\right\} = \text{diag}(\psi_1, \psi_2, \ldots, \psi_h),$$

$$\text{where } \psi_j = \begin{cases} \frac{1}{|\tilde{\beta}_j(\lambda)|} & \text{if } \tilde{\beta}_j(\lambda) \neq 0 \\ 0 & \text{otherwise} \end{cases}, \quad \text{for } j = 1, \ldots, h.$$

Then, we can also establish the properties on the variance of the approximated $\tilde{\beta}$ under the proposed FAME framework.

Theorem 2. *Use the same notation as in Theorem 1, when $\lambda_{min}(\boldsymbol{R}) > 0$ and $\tilde{\boldsymbol{D}}_S$ is the reduced dataset, then the variance of the approximated $\tilde{\beta}$ of the lasso regression satisfies*

$$\text{Var}\left(\tilde{\beta}_q|\tilde{\boldsymbol{D}}_S\right) \leq \frac{4h\sigma^2}{k\lambda_{\min}(\boldsymbol{R})\left(\overline{x}_{(u)q} - \overline{x}_{(v)q}\right)^2}, \quad \text{for } q = 1, \ldots, h \tag{9}$$

Theorem 2 is a finite sample property of $\tilde{\beta}$ from the subdata selected via the FAME algorithm. In the cases when $\lambda = 0$ and all variables are selected,

$$V\left(\tilde{\boldsymbol{\beta}}_{\boldsymbol{q}}|\tilde{\boldsymbol{D}}_S\right) \geq \frac{4\sigma^2}{k\lambda_{\max}(\boldsymbol{R})\left(x_{nq} - x_{1q}\right)^2} \tag{10}$$

where x_{nq} is the largest order statistic and x_{1q} is the smallest order statistics in $\boldsymbol{x}^{(q)}$. Assume that $\lim_{n\to\infty}\ \lambda_{\min}^h(\boldsymbol{R}) > 0$. As $n \to \infty$, $\tilde{\beta}$ obtained from $\tilde{\boldsymbol{D}}_S$ has the following property:

$$\text{Var}\left(\tilde{\beta}_q|\tilde{\boldsymbol{D}}_S\right) = O_p\left(\frac{h}{k\left(\overline{x}_{(u)q} - \overline{x}_{(v)q}\right)^2}\right), q = 1, \ldots, h$$

This theorem holds for any values of n, r, and h and can also be used to obtain results when one or more of these values go to infinity.

Theorem 3. *Denote $\boldsymbol{D}$ as the full data and the estimator of linear model using $\boldsymbol{D}$ as $\widehat{\beta}$. Denote the upper and lower bound of $\text{Var}\left(\tilde{\beta}_q|\tilde{\boldsymbol{D}}_S\right)$ as $U_{\tilde{\beta}_q|\tilde{\boldsymbol{D}}_S}$ and $L_{\tilde{\beta}_q|\tilde{\boldsymbol{D}}_S}$. Similarly, the upper and lower bound of $\text{Var}\left(\widehat{\beta}_q|\boldsymbol{D}\right)$ are $U_{\widehat{\beta}_q|\boldsymbol{D}}$ and $L_{\widehat{\beta}_q|\boldsymbol{D}}$. Then*

$$U_{\tilde{\beta}_q|\boldsymbol{D}_S} \leq U_{\widehat{\beta}_q|\boldsymbol{D}} \tag{11}$$

$$L_{\tilde{\beta}_q|\boldsymbol{D}_S} \geq L_{\widehat{\beta}_q|\boldsymbol{D}} \tag{12}$$

Theorem 3 shows that the estimator based on $\tilde{\boldsymbol{D}}_S$ can have a smaller variance range compared to the estimator based on the full data $\boldsymbol{D}$.

2.3 | Extension of Binary Data

We can also extend the proposed FAME method for noncontinuous responses or noncontinuous predictors. Here, we focus on binary response or binary predictors with two levels.

When the response variable $\boldsymbol{y}$ is binary as $y_i \in \{-1, 1\}$ for data point i and the input matrix is continuous, we can use the main effect in the experimental design to construct the score index w_j. That is, the score index w_j can be calculated as the difference of the average input between two levels of the response variable.

$$w_j = \overline{\boldsymbol{x}}^{(j)}(y_i+) - \overline{\boldsymbol{x}}^{(j)}(y_i-) = \frac{1}{n^{(+)}}\sum_{y_i=1} x_{ij} - \frac{1}{n^{(-)}}\sum_{y_i=-1} x_{ij}, \quad 1 \leq j \leq p, 1 \leq i \leq n; \tag{13}$$

where $\overline{\boldsymbol{x}}^{(j)}(y_i+)$ is the average of $\boldsymbol{x}^{(j)}$ values observed at the positive response level and $\overline{\boldsymbol{x}}^{(j)}(y_i-)$ is the average of $\boldsymbol{x}^{(j)}$ values observed at the negative response level. Here, $n^{(+)}$ and $n^{(-)}$ are the number of observations with response label $y = 1$ and $y = -1$, respectively.

Since the response variable $\boldsymbol{y}$ is binary, we consider the logistic regression for model estimation. Denote $\boldsymbol{x}_{m\tilde{s}} \in \mathcal{R}^h$ is the data point and $y_m \in \{-1, 1\}$ is the response variable in $\tilde{\boldsymbol{D}}_S$, where $m = 1, \ldots, 2l$. Then, $\Pr(\tilde{\boldsymbol{y}}_{\boldsymbol{m}} = 1|\boldsymbol{x}_{m\tilde{s}}) = \frac{\exp\left(\boldsymbol{x}_{m\tilde{s}}^T\boldsymbol{\beta}\right)}{1+\exp\left(\boldsymbol{x}_{m\tilde{s}}^T\boldsymbol{\beta}\right)}$, where $\boldsymbol{\beta}$ is an $h \times 1$ vector of regression coefficients. A penalized likelihood

estimation can be used for parameter estimation of $\boldsymbol{\beta}$ by minimizing the negative log-likelihood function.

$$\tilde{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}}\{-\ell(\boldsymbol{\beta}) + \lambda\|\boldsymbol{\beta}\|_1\} \tag{14}$$

where $\ell(\boldsymbol{\beta}) = \frac{1}{2l}\sum_{m=1}^{2l}\log\left[1+\exp\left(-y_m \boldsymbol{x}_{m\tilde{S}}^T\boldsymbol{\beta}\right)\right]$. Here, $\lambda \geq 0$ is a tuning parameter. Algorithm 2 summarizes the proposed FAME algorithm for the binary response variable.

ALGORITHM 2 | (FAME for binary response).

Step 1: For $1 \leq j \leq p$, calculate the score index $w_j = g\left(\boldsymbol{x}^{(j)}, \boldsymbol{y}\right)$ in Equation (13);

Step 2: Form $\boldsymbol{D}_S = (\boldsymbol{X}_S, \boldsymbol{y})$, where $\boldsymbol{X}_S$ includes h columns with the h largest w_j values;

Step 3: For $p = 1, \ldots, h$, based on column $x^{(p)}$ in $\boldsymbol{X}_S$, form $\tilde{\boldsymbol{D}}_S = (\tilde{\boldsymbol{X}}_S, \tilde{\boldsymbol{y}})$ by including $2l$ data points with the l smallest values and l largest values but excluding data points already been selected;

Step 4: Obtain model estimation from Equation (14).

The second extension is to consider the FAME method for data with binary predictor variables with two levels as "+" and "−." For the selection of columns, we then can define the score index based on the difference in the average response between two levels of the predictor variable. That is,

$$w_j = \bar{y}_i\left(\boldsymbol{x}^{(j)}+\right) - \bar{y}_i\left(\boldsymbol{x}^{(j)}-\right) = \frac{1}{n^{(+)}}\sum_{x_{ij}=1} y_i - \frac{1}{n^{(-)}}\sum_{x_{ij}=-1} y_i,$$
$$1 \leq j \leq p, 1 \leq i \leq n \tag{15}$$

where $\bar{y}_i\left(\boldsymbol{x}^{(j)}+\right)$ is the average of y_i values observed at the positive predictor level and $\bar{y}_i\left(\boldsymbol{x}^{(j)}-\right)$ is the average of y_i values observed at the negative predictor level for the jth covariate. Here, $n^{(+)}$ and $n^{(-)}$ are the number of observations with the predictor $\boldsymbol{x}^{(j)} = 1$ and $\boldsymbol{x}^{(j)} = -1$, respectively.

For the selection of rows, however, the use of criterion in Equation (4) become improper since

$$|\boldsymbol{I}(\delta)| \leq C\frac{k^{h+1}l^{2h}}{4^h\sigma^{2(h+1)}} \tag{16}$$

where C is a positive constant. Clearly, the order statistics of each column in the input matrix do not work for selecting subdata to maximize the information matrix. To overcome this issue, for each filtered column $\boldsymbol{x}^{(q)}$, sorting predictor values and denote the location of the value closest to $\bar{y}_i\left(\boldsymbol{x}^{(q)}+\right)$ as $\gamma^{(q)}(+)$ and the location of the value closest to $\bar{y}_i\left(\boldsymbol{x}^{(q)}-\right)$ as $\gamma^{(q)}(-)$, denoting [ ] as the floor division operator. When l is odd, selecting the l data points with response variable closest to $\bar{y}_i\left(\boldsymbol{x}^{(q)}+\right)$ denoted as $y_{\gamma^{(q)}(+)-\left[\frac{l}{2}\right]:\gamma^{(q)}(+)+\left[\frac{l}{2}\right]}\left(x_q+\right)$ and l data points with response variable closest to $\bar{y}_i\left(\boldsymbol{x}^{(q)}-\right)$ denoted as $y_{\gamma^{(q)}(-)-\left[\frac{l}{2}\right]:\gamma^{(q)}(-)+\left[\frac{l}{2}\right]}\left(x_q-\right)$. When l is even, selecting the l data points $y_{\gamma^{(q)}(+)-\left[\frac{l}{2}\right]+1:\gamma^{(q)}(+)+\left[\frac{l}{2}\right]}\left(x_q+\right)$ and l data points $y_{\gamma^{(q)}(-)-\left[\frac{l}{2}\right]+1:\gamma^{(q)}(-)+\left[\frac{l}{2}\right]}\left(x_q-\right)$.

Algorithm 3 summarizes the proposed FAME algorithm for binary predictor variable. It is easy to see that the computing time is the same as in Algorithm 1, which is $O(np)$.

ALGORITHM 3 | (FAME for binary predictor variables).

Step 1: For $1 \leq j \leq p$, calculate the score index $w_j = g\left(\boldsymbol{x}^{(j)}, \boldsymbol{y}\right)$ in Equation (15);

Step 2: Form $\boldsymbol{D}_S = (\boldsymbol{X}_S, \boldsymbol{y})$, where $\boldsymbol{X}_S$ includes the h columns with the h largest w_j values;

Step 3: For $q = 1, \ldots, h$, based on column $x^{(q)}$ in $\boldsymbol{X}_S$, form $\tilde{\boldsymbol{D}}_S = (\tilde{\boldsymbol{X}}_S, \tilde{\boldsymbol{y}})$ by including $2l$ data points with the l values around mean of y_i when x_{iq} is positive and l values around mean of y_i when x_{iq} is negative but excluding data points already been selected;

Step 4: Obtain coefficient estimation from Equation (6).

Corollary 1. *For the subdata selected by the FAME algorithm with binary predictor variables, denote* $\boldsymbol{R}$ *as the sample correlation matrix of* $\tilde{\boldsymbol{D}}_S$, *the determinant* $\left|\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right|$ *satisfies*

$$\left|\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right| \geq Ck\left(\frac{k\lambda_{min}(\boldsymbol{R})}{h}\right)^h \tag{17}$$

where C is a constant.

The lower bound of the information matrix of subdata $\tilde{\boldsymbol{D}}_S$ is determined by the number of columns selected in the subdata as well as the smallest eigenvalue of the correlation matrix of the subdata. Based on Theorem 3, we can know that this bound is larger than the lower bound of the information matrix of the full data and will not converge to 0 as $n \to \infty$ when $\lim_{n\to\infty}\lambda_{\min}(\boldsymbol{R}) > 0$.

Corollary 2. *When* $\lambda_{min}(\boldsymbol{R}) \geq 0$, *for the reduced dataset* $\tilde{\boldsymbol{D}}_S$ *with binary predictors*,

$$\mathrm{Var}\left(\tilde{\beta}_q|\tilde{\boldsymbol{D}}_S\right) \leq \frac{Ch\sigma^2}{k\lambda_{\min}(\boldsymbol{R})}, \text{for } q = 1, \ldots, h \tag{18}$$

The upper bound of $\mathrm{Var}\left(\tilde{\beta}_h\right)$ is not related to the binary predictors as in the situation of continuous predictors.

Corollary 3. *As* $n \to \infty$, *assuming* $lim_{n\to\infty}\lambda_{min}(\boldsymbol{R}) > 0$,

$$\mathrm{Var}\left(\tilde{\beta}_q|\tilde{\boldsymbol{D}}_S\right) = O_p\left(\frac{h}{k}\right), q = 1, \ldots, j \tag{19}$$

When h and k are fixed, it is seen that $\mathrm{Var}\left(\tilde{\beta}_q|\tilde{\boldsymbol{D}}_S\right) = O_p(1)$.

3 | Simulation

In this section, we will evaluate the performance of the proposed FAME method using simulated data with continuous predictors or discrete predictors. To estimate the coefficient parameter $\tilde{\boldsymbol{\beta}}$, we compare the proposed FAME method with five benchmark methods: IBOSS, IBOSS_LASSO, SIS, SIS_LASSO, and LASSO.

1. FAME: The proposed FAME algorithm;
2. IBOSS: Benchmark method selecting predictors randomly and selecting features using the IBOSS method [18]. The subdata are fitted with multivariate linear regression;

3. IBOSS_LASSO: Benchmark method selecting predictors randomly and selecting features using IBOSS method [18]. The subdata are fitted with regularized regression;
4. SIS: Benchmark method selecting predictors using the SIS method [13] with all data points. The subdata are fitted with multivariate linear regression;
5. SIS_LASSO: Benchmark method selecting predictors using the SIS method [13] with all data points. The subdata are fitted with regularized regression;
6. LASSO: Benchmark method on the original data fitted with regularized regression.

The subdata selected by each method are shown in Figure 1. The light color in the plot represents selected columns or rows, while dark color represents selected subdata.

Subdata are split into 70% training data and 30% testing data. Parameters are estimated on training data and the performance of methods is evaluated by empirical mean squared errors (MSEs) and signal-to-noise ratio (SNR) on testing data. MSE $= \frac{1}{n_t S}\sum_{s=1}^{S}\left\|\boldsymbol{y}^{(s)}-\hat{\boldsymbol{y}}^{(s)}\right\|^2$, where S is the number of simulation and n_t is the number of data points in the test dataset. $\boldsymbol{y}^{(s)}$ is the true response value, and $\hat{\boldsymbol{y}}^{(s)}$ is the predicted response value in each simulation. SNR $= \frac{1}{n}\sum_{s=1}^{S}\frac{\mathrm{var}\left(\hat{\boldsymbol{y}}^{(s)}\right)}{\hat{\sigma}^{(s)}}$, where $\hat{\sigma}^{(s)}$ is the variance of the predicted residuals.

3.1 | Cases With Continuous Predictors

Data are generated based on Equation (1) with 2000 data points, 5 true predictors among 1000 features and $\sigma^2 = 16$. Covariates matrix $\boldsymbol{X}$ is generated from multivariate normal distribution $\boldsymbol{X} \sim N(\boldsymbol{0}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}$ is a covariance matrix with $\Sigma_{ij} = 0.5^{\mathbb{I}(i\neq j)}$ under dependent situation and $\boldsymbol{\Sigma} = \boldsymbol{I}$ under independent situation. $\mathbb{I}()$ is the indicator function. The simulation is repeated $S = 50$ times, and each column is standardized. The selected subdata has 200 data points with five columns or 25 columns. Besides MSE, methods are also compared in terms of the standard deviation of MSE, true positive rate (TPR), true negative rate (TNR), and computational time (TIME), where TPR = true positive/(true positive + false negative) and TNR = true negative/(true negative + false positive).

Table 1 summarizes the estimation results of selecting 200 data points from five columns. For both independent and dependent covariates cases, the proposed FAME method beats the benchmark methods with smaller MSE. Compared with IBOSS with MSE, the proposed methods selected the most important features instead of randomly selecting, which contributes to decreasing the prediction MSE from 19.49 to 16.90. Compared with SIS, the proposed FAME method uses t-statistic as the score index instead of componentwise regression. The MSE of SIS_LASSO is 37.10 which is much larger than the MSE of FAME. This big difference in prediction accuracy shows that the proposed FAME method has higher accuracy in selecting the most informative features

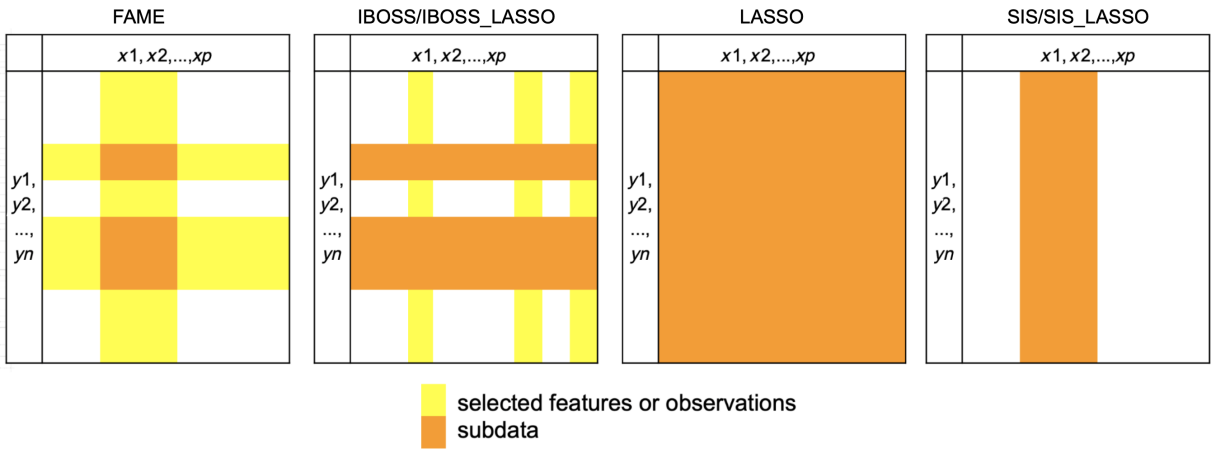

FIGURE 1 | An illustration of the proposed method and three benchmark methods.

TABLE 1 | Results of filtering 200 data points from five columns.

		FAME	IBOSS	IBOSS_LASSO	SIS	SIS_LASSO	LASSO
Independent covariates	MSE	16.90	1.35e + 07	19.49	37.10	37.10	16.43
	TPR	0.86	1.00	0.90	0.64	0.63	0.96
	TNR	1.00	0.80	0.97	1.00	1.00	0.98
	TIME	0.05	0.05	0.26	0.002	0.06	2.49
Dependent covariates	MSE	18.19	6.64e + 04	18.85	44.81	44.85	16.34
	TPR	0.82	1.00	0.90	0.63	0.61	0.96
	TNR	1.00	0.80	0.97	1.00	1.00	0.97
	TIME	0.05	0.05	0.26	0.002	0.06	1.79

compared with SIS. LASSO used all the data for prediction and has the smallest MSE of all six methods. However, the MSE of FAME is similar to that of LASSO with less number of both predictors and data points.

IBOSS, IBOSS_LASSO, and LASSO used all features for parameter estimation while FAME only utilized five features. The TPR of FAME is smaller and TNR is higher than those three methods. SIS already selected several features before regularized regression, which reduces the number of features, so the MSE of SIS_LASSO is similar to that of SIS which is 37.10. However, the regularized regression improves the IBOSS methods by reducing the MSE from 1.35e + 07 to 19.49, leading to a decrease in TPR from 1 to 0.9.

The proposed FAME method generates a subdata with the smallest size by reducing both the number of features and data points. The computation time is also small compared to LASSO using all data and is similar to other benchmark methods. SIS has the smallest computation time because of the small number of features and no parameter tuning for regularized regression.

The result is similar when selecting 25 features as shown in Table 2. The proposed FAME method has MSE similar to that of LASSO and smaller than other benchmark methods. The prediction accuracy is high, the size of the subdata generated by FAME is the smallest among all methods and the computation costs is also low. For the simulated dataset, only five features out of 1000 are true covariates, so compared with Table 1, the MSE of FAME does not improve, while SIS has a smaller MSE because more important features are selected.

Table 3 summarizes the variance of the estimation of selecting 200 data points from 25 or five columns. When 25 columns are selected from 1000 independent predictors, the predicted variance of all methods is smaller than that of the true variance besides the IBOSS method. The proposed FAME method has the largest SNR besides the LASSO method. The above phenomenon can also be found when the predictors are dependent or when five columns are selected from 1000 predictors. The proposed FAME method has a prediction variance close to the true variance and an SNR close to that of LASSO. The subdata generated by FAME are the smallest, and the computation time is small.

To determine the optimal number of columns in the subdata, we apply the change-point detection method described in Section 2. As shown in Figure 2, the first location of the change point detected is on the fifth column. One would recommend the number of features included in the subdata to be five, which is the number of true coefficients when generating the original data. The performance of the proposed FAME algorithm does not seem to improve much when more than five columns are included in the subdata.

Note that the number of data points contained in the subdata is 200 in this simulation study, where 40 points from each column are selected. We also examine the performance of the prediction MSE concerning the number of points in the subdata in Figure 3. As shown in Figure 3, for the dependent case, the

TABLE 2 | Results of filtering 200 data points from 25 columns.

		FAME	IBOSS	IBOSS_LASSO	SIS	SIS_LASSO	LASSO
Independent covariates	MSE	17.19	1.60e + 08	19.52	28.38	28.15	16.43
	TPR	0.90	1.00	0.90	0.72	0.72	0.96
	TNR	1.00	0.80	0.97	0.98	1.00	0.98
	TIME	0.06	0.05	0.26	0.004	0.07	2.44
Dependent covariates	MSE	17.63	1.05e + 06	18.75	28.91	28.83	16.35
	TPR	0.87	1.00	0.91	0.73	0.73	0.96
	TNR	0.99	0.80	0.97	0.98	0.99	0.97
	TIME	0.05	0.05	0.23	0.004	0.07	1.78

TABLE 3 | Results of filtering simulated subdata on prediction variance.

	25/1000						5/1000					
	Independent			Dependent			Independent			Dependent		
Method	Prediction variance	True variance	SNR	Prediction variance	True variance	SNR	Prediction variance	True variance	SNR	Prediction variance	True variance	SNR
FAME	5.40		12.94	6.95		59.70	5.45		13.78	6.96		56.58
IBOSS	18.88		1.05	13.85		1.35	16.42		1.07	11.11		1.27
IBOSS_LASSO	5.30	5.53	10.35	6.94	6.99	55.69	5.30	5.53	10.38	6.94	6.99	55.26
SIS	5.41		10.32	6.96		45.98	5.37		8.74	6.94		35.49
SIS_LASSO	5.38		10.14	6.94		45.51	5.35		8.62	6.93		34.94
LASSO	5.40		13.58	6.96		64.75	5.40		13.58	6.96		64.75

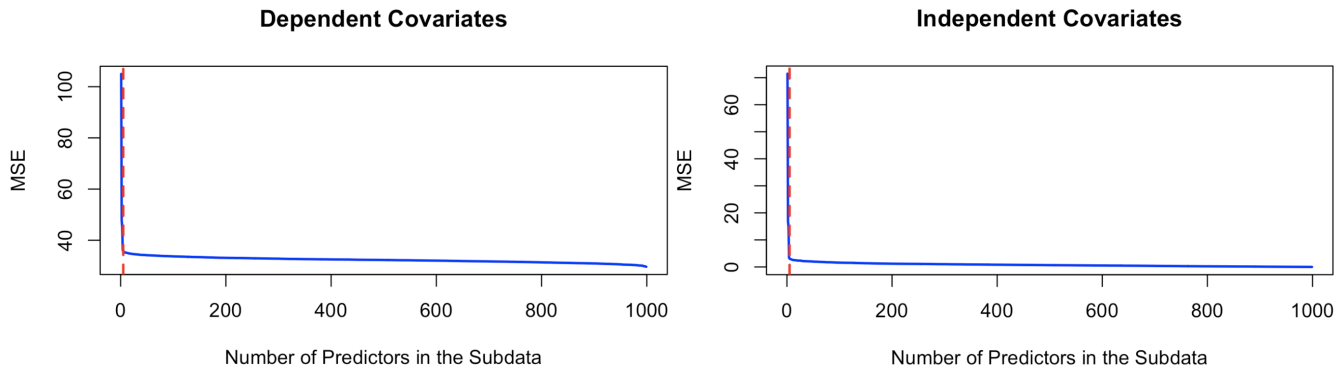

FIGURE 2 | Change-point detection using t-statistics for the simulated dataset.

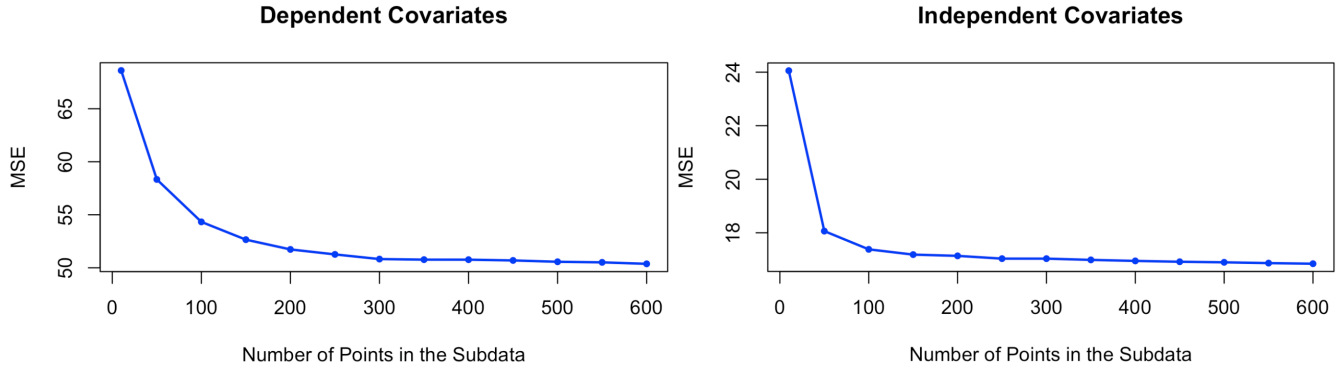

FIGURE 3 | MSE and number of points in the subdata.

MSE drops significantly when the number of data points in the subdata increases from 10 to 200. For the independent case, the turning point of MSE occurs when the subdata have around 100 data points. It is seen that the MSE does not improve much when we continuously increase the size of the subdata. Without loss of generality, one would recommend that the rule of thumb is to choose at least 10 points from each selected column to form the subdata.

3.2 | Cases With Binary Predictors

The simulated dataset has 700 observations and 700 predictors, of which five are true predictors generated from the normal distribution. The predictors came from a binomial distribution with a probability between a range of 0.3 to 0.7, representing that the covariates can be imbalanced. The response variable is generated based on Equation (1), where ϵ is the distributed normally with a mean of 0 and a standard deviation of 0.3. The simulation is repeated 50 times, and empirical mean squared error (MSE) is the measurement metric.

The simulation result is shown in Table 4. The proposed method has the smallest MSE compared with all benchmark methods, besides LASSO using all data. It shows the superior performance of feature selection and data points selection in FAME. Compared with IBOSS and IBOSS_Lasso, FAME selected the most important features and data points. Compared with SIS and SIS_Lasso, the main effect is more effective in selecting the most important features. The TPR of SIS and SIS_LASSO is only around 0.16, while that for FAME is around 0.8. IBOSS and LASSO used all the data, so the TPR of these two methods is almost 100% accurate. The subdata generated by FAME are the smallest among all methods with low computation cost.

4 | Case Studies

4.1 | ARCENE Data in Cancer Study

The ARCENE dataset is obtained from the UCI Machine Learning Repository, which was published by the National Cancer Institute (NCI) and the Eastern Virginia Medical School (EVMS). The objective of this study is to distinguish cancer based on patterns which are continuous input variables. It is a classification problem with binary output. Among the 10,000 predictors, 7000 are real variables while the other 3000 variables are random probes. Both the training and testing dataset has 100 observations. The estimation result is measured by classification error f/n, where f is the number of sample cases incorrectly classified, and n is the total number of sample cases.

Table 5 summarizes the estimation results of the proposed and benchmark methods. When the subdata has five columns, the classification error of the proposed FAME method is the smallest among all methods. Compared with IBOSS and IBOSS_LASSO, the proposed method selected the most several important features. Compared with SIS and SIS_LASSO, the main effect performs well in ranking feature importance. Although LASSO used all features, its MSE is larger than that of FAME.

TABLE 4 | Estimating result for selecting five columns and 50 data points for dataset with binary predictors.

	FAME	IBOSS	IBOSS_LASSO	SIS	SIS_LASSO	LASSO
MSE	0.23	199.73	0.24	0.97	0.96	0.10
TPR	0.78	1.00	0.76	0.17	0.16	0.96
TNR	1.00	0.94	0.97	1.00	1.00	0.96
TIME	0.04	0.01	0.12	0.001	0.05	0.33

TABLE 5 | Results of filtering subdata with 50 observations from five or 10 columns of ARCENE data.

	$c = 5$		$c = 10$	
	Classification error	Time	Classification error	Time
FAME	0.26	0.08	0.28	0.25
IBOSS	0.47	3.88	0.48	3.61
IBOSS_LASSO	0.36	0.82	0.37	0.74
SIS	0.35	0.002	0.33	0.004
SIS_LASSO	0.42	0.84	0.40	1.85
LASSO	0.33	2.81	0.33	2.57

TABLE 6 | Result of filtering subdata with 50 observations from five columns of P&G data.

	R068_1min				R068_5min			
	5/59		5/3009		5/59		5/3009	
Method	MSE (SD)	Time (SD)	MSE (SD)	Time (SD)	MSE (SD)	Time (SD)	MSE (SD)	Time (SD)
FAME	1.87	0.05	2.05	0.04	2.94	0.05	3.22	0.05
	(0.55)	(0.01)	(0.42)	(0.01)	(0.60)	(0.01)	(0.81)	(0.003)
IBOSS	2.39e + 07	0.002	4.04e + 08	0.03	1.24e + 05	0.002	8.13e + 05	0.07
	(1.69e + 08)	(0.002)	(1.97e + 09)	(0.005)	(4.88e + 05)	(0.00)	(4.29e + 06)	(0.03)
IBOSS LASSO	2.17	0.12	3.72	0.18	8.82	0.11	4.91	0.28
	(1.12)	(0.02)	(6.87)	(0.03)	(26.93)	(0.03)	(11.13)	(0.05)
SIS	2.03e + 04	0.002	9.16e + 03	0.002	10.37	0.003	5.58	0.003
	(8.24e + 04)	(0.00)	(4.89e + 04)	(0.002)	(23.91)	(0.00)	(1.26)	(0.00)
SIS LASSO	1.96	1.18	2.66	0.12	2.81	0.87	3.53	0.13
	(0.93)	(0.54)	(0.69)	(0.12)	(0.93)	(0.39)	(1.30)	(0.16)
LASSO	1.83	1.55	1.79	0.26	2.94	1.24	2.70	0.62
	(0.66)	(0.62)	(0.32)	(0.04)	(1.80)	(0.36)	(0.44)	(0.07)

Meanwhile, the size of the subdata generated by FAME is the smallest among all. A similar pattern can be observed when 10 features are selected in the subdata.

4.2 | Babycare Process Data in Manufacturing System

The P&G data are offered by Procter & Gamble with 161 continuous observations and 59 continuous features. The response variable is the number of rejects in the production line in a 1-min interval or 5-min interval. By permuting each feature with noise, the number of features increases from 59 to 3009. The subdata contain 50 observations and 50 features. The estimation result is evaluated using MSE and SNR.

The estimation result is shown in Table 6. For the production line with a 1-min interval, when the subdata are filtered from the original data with 59 features or 3009 features, the proposed FAME method performs the best besides LASSO, then is SIS_LASSO and IBOSS_LASSO. The superiority of FAME is more obvious when the number of features is large. When there are 59 features, the MSE of FAME is 1.87. Increasing the number of features to 3009, the MSE of FAME increases to 2.05 while that of SIS_LASSO increases from 1.96 to 2.66 and that of IBOSS_LASSO increases from 2.17 to 3.72. LASSO always has the smallest MSE because it

TABLE 7 | Results of filtering subdata with 50 observations from five columns of P&G data on prediction variance.

	R068_1min						R068_5min					
	5/59			5/3009			5/59			5/3009		
Method	Prediction variance	True variance	SNR	Prediction variance	True variance	SNR	Prediction variance	True variance	SNR	Prediction variance	True variance	SNR
FAME	0.28	1.80	0.15	0.47	1.80	0.24	0.45	2.67	0.15	0.56	2.67	0.17
IBOSS	2.39e + 07		1.00	4.04e + 08		1.00	1.20e + 05		0.99	7.87e + 05		1.00
IBOSS_LASSO	0.42		0.11	1.92		0.13	6.33		0.22	2.18		0.08
SIS	2.02e + 04		0.80	9.11e + 03		0.50	8.91		0.64	1.35		0.34
SIS_LASSO	0.33		0.10	0.54		0.27	0.29		0.08	0.35		0.11
LASSO	0.21		0.08	0.07		0.04	0.41		0.07	0.04		0.01

uses all original data. However, the proposed method generates subdata with the smallest size and low computational cost.

The superiority of the proposed method compared with benchmark methods, especially for large datasets, can be observed for the production line with 5-min interval. When the number of features is 59, the MSE of FAME is similar to LASSO, but both are larger than the mSE of SIS_LASSO. However, increasing the number of features to 3009, the MSE of FAME is the smallest besides LASSO.

The variance and SNR of the predicted estimator are shown in Table 7. For the production line with a 1-min interval, the variance of the predicted response variable of the proposed FAME method is the smallest besides LASSO. For the production line with a 5-min interval, although the variance of the predicted response variable of FAME is larger than SIS_LASSO, the SNR of FAME is larger than that of SIS_LASSO.

5 | Discussion

In this work, we proposed the FAME method for efficient data reduction for sparse regression. This proposed methodology conducts data reduction in terms of reducing both the number of rows and columns of the original data considering the correlation between predictors and response. Overall, the proposed FAME method is robust in parameter estimation and outperforms benchmark methods in scenarios including continuous response with continuous predictors, continuous response with binary predictors, and binary response with continuous predictors. The superiority is more obvious when the number of data points is much larger than the number of predictors. The advantage of the proposed method comes from selecting the most important features and data points while considering the relationship between the response variable and predictors in the procedure. The subdata generated from the proposed FAME method is always the smallest compared to the benchmark methods.

We would like to remark that the proposed FAME method may not be the best data reduction method but aims to strike a good balance between data reduction efficiency and model estimation accuracy. There are several aspects worth further investigation. Currently, we only discuss the scenarios when the predictors or response are binary. However, in practice, the non-continuous predictors or response variables usually can have more than two distinct categories. In some cases, both predictors and response can be discrete. This brings the challenge to the proposed FAME algorithm on how to select data points in each level of predictor variables. Besides, for datasets with discrete responses, it is possible that the dataset is highly imbalanced where the majority of classes dominate the whole dataset. It will be of future interest to improve the proposed FAME method to handle imbalanced data. Third, the current FAME method assumes a linear model for data reduction. However, the linear model assumption can be too strong or can be misspecified for some case studies in practice. It will be interesting to extend the proposed method to enable data reduction for nonlinear models or nonparametric models.

Author Contributions

Yanran Wei: conceptualization, methodology, implementation, writing and editing. **William Myers:** conceptualization, editing. **Xinwei Deng:** conceptualization, editing, supervision.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data that support the findings of the case study are available from the corresponding author upon reasonable request. Code is available on https://github.com/Echo-Wei/FAME-Subdata-Selection. It contains the proposed and benchmark methods as well as examples of implementing those functions in R software.

References

1. P. Drineas, M. W. Mahoney, S. Muthukrishnan, and T. Sarlós, "Faster Least Squares Approximation," *Numerische Mathematik* 117 (2011): 219–249.

2. P. Ma, M. W. Mahoney, and B. Yu, "A Statistical Perspective on Algorithmic Leveraging," *Journal of Machine Learning Research* 16 (2015): 861–911.

3. H. Wang, M. Yang, and J. Stufken, "Information-Based Optimal Subdata Selection for Big Data Linear Regression," *Journal of the American Statistical Association* 114 (2019): 393–405.

4. L. Deldossi and C. Tommasi, "Optimal Design Subsampling From Big Datasets," *Journal of Quality Technology* 54 (2022): 93–101.

5. R. Xie, S. Bai, and P. Ma, "Optimal Sampling Designs for Multi-Dimensional Streaming Time Series With Application to Power Grid Sensor Data," *Annals of Applied Statistics* 17, no. 4 (2023): 3195–3215.

6. C. Meng, R. Xie, A. Mandal, X. Zhang, W. Zhong, and P. Ma, "LowCon: A Design-Based Subsampling Approach in a Misspecified Linear Model," *Journal of Computational and Graphical Statistics* 30 (2020): 694–708.

7. R. Singh and J. Stufken, "Subdata Selection With a Large Number of Variables," *New England Journal of Statistics in Data Science* 1 (2023): 426–438.

8. Q. Cheng, H. Wang, and M. Yang, "Information-Based Optimal Subdata Selection for Big Data Logistic Regression," *Journal of Statistical Planning and Inference* 209 (2020): 112–122.

9. M. Ai, F. Wang, J. Yu, and H. Zhang, "Optimal Subsampling for Large-Scale Quantile Regression," *Journal of Complexity* 62 (2021): 101512.

10. R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society: Series B: Methodological* 58 (1996): 267–288.

11. J. Fan and R. Li, "Variable Selection via Nonconcave Penalized Likelihood and Its Oracle Properties," *Journal of the American Statistical Association* 96 (2001): 1348–1360.

12. E. Candes and T. Tao, "The Dantzig Selector: Statistical Estimation When p Is Much Larger Than n," *Annals of Statistics* 35 (2007): 2313–2351.

13. J. Fan and J. Lv, "Sure Independence Screening for Ultrahigh Dimensional Feature Space," *Journal of the Royal Statistical Society, Series B: Statistical Methodology* 70 (2008): 849–911.

14. H. Wang, "Factor Profiled Sure Independence Screening," *Biometrika* 99 (2012): 15–28.

15. N. Zhao, Q. Xu, M.-L. Tang, B. Jiang, Z. Chen, and H. Wang, "High-Dimensional Variable Screening Under Multicollinearity," *Stat* 9 (2020): e272.

16. B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least Angle Regression," *Annals of Statistics* 32 (2004): 407–499.

17. J. Kiefer, "Optimum Experimental Designs," *Journal of the Royal Statistical Society: Series B: Methodological* 21 (1959): 272–319.

18. R. Killick, P. Fearnhead, and I. A. Eckley, "Optimal Detection of Changepoints With a Linear Computational Cost," *Journal of the American Statistical Association* 107 (2012): 1590–1598.

19. S.-G. Hwang, "Cauchy's Interlace Theorem for Eigenvalues of Hermitian Matrices," *American Mathematical Monthly* 111 (2004): 157–159.

Appendix A

A.1 | Proof of Theorem 1

Proof. For the full data $\boldsymbol{D}$ with n rows and p columns, denote the subdata obtained after feature screening is $\boldsymbol{D}_S = (\boldsymbol{X}_S, \boldsymbol{y})$ and the one from the FAME algorithm as $\tilde{\boldsymbol{D}}_S = (\tilde{\boldsymbol{X}}_{\boldsymbol{S}}, \tilde{\boldsymbol{y}})$. $x_{(i)q}$ being the ith order statistic for $\boldsymbol{x}_q = \{x_{(1)q}, \cdots, x_{(n)q}\}$, where $q = 1, \cdots, h$ and $i = 1, \cdots, n$.

Let $\bar{\tilde{\boldsymbol{x}}}$ and $\mathrm{var}(\tilde{\boldsymbol{x}}_{\boldsymbol{q}})$ be the mean and variance for the covariate $\tilde{\boldsymbol{x}}_{\boldsymbol{q}}$ in $\tilde{\boldsymbol{D}}_S$. The information matrix of $\tilde{\boldsymbol{D}}_S$ is

$$\tilde{\boldsymbol{X}}_{\boldsymbol{S}}^T\tilde{\boldsymbol{X}}_{\boldsymbol{S}} = \boldsymbol{B}^{-1}\begin{bmatrix} k & \boldsymbol{0}^T \\ \boldsymbol{0} & (k-1)\boldsymbol{R} \end{bmatrix}\left(\boldsymbol{B}^T\right)^{-1} \tag{20}$$

where

$$\boldsymbol{B} = \begin{bmatrix} 1 & & \\ -\frac{\bar{\tilde{\boldsymbol{x}}}_1}{\sqrt{\mathrm{var}(\tilde{\boldsymbol{x}}_1)}} & \frac{1}{\sqrt{\mathrm{var}(\tilde{\boldsymbol{x}}_1)}} & \\ \cdots & & \ddots \\ -\frac{\bar{\tilde{\boldsymbol{x}}}_h}{\sqrt{\mathrm{var}(\tilde{\boldsymbol{x}}_h)}} & & \frac{1}{\sqrt{\mathrm{var}(\tilde{\boldsymbol{x}}_h)}} \end{bmatrix} \tag{21}$$

From Equations (20) and (21),

$$|\tilde{\boldsymbol{X}}_{\boldsymbol{S}}^T\tilde{\boldsymbol{X}}_{\boldsymbol{S}}| = k|(k-1)\boldsymbol{R}|\prod_{q=1}^{h}\mathrm{var}(\tilde{\boldsymbol{x}}_{\boldsymbol{q}}) \geq k(k-1)^h\lambda_{\min}^h(\boldsymbol{R})\prod_{q=1}^{h}\mathrm{var}(\tilde{\boldsymbol{x}}_{\boldsymbol{q}}) \tag{22}$$

For each sample variance,

$$(k-1)\mathrm{var}(\tilde{\boldsymbol{x}}_{\boldsymbol{q}}) = \sum_{i=1}^{k}\left(\tilde{x}_{iq} - \bar{\tilde{\boldsymbol{x}}}_q\right)^2 \tag{23}$$

$$\geq \left(\sum_{i=1}^{l} + \sum_{i=n-l+1}^{n}\right)\left(x_{(i)q} - \bar{\tilde{\boldsymbol{x}}}_q^*\right)^2 \tag{24}$$

$$= \sum_{i=1}^{l}\left(x_{(i)q} - \bar{x}_{(u)q}\right)^2 + \sum_{i=n-l+1}^{n})\left(x_{(i)q} - \bar{x}_{(v)q}\right)^2 + \frac{l}{2}\left(\bar{x}_{(u)q} - \bar{x}_{(v)q}\right)^2 \tag{25}$$

$$\geq \frac{l}{2}\left(\bar{x}_{(u)q} - \bar{x}_{(v)q}\right)^2 \tag{26}$$

$$\geq \frac{l}{2}\left(x_{(n-l+1)q} - x_{(l)q}\right)^2 \tag{27}$$

where $\bar{\tilde{\boldsymbol{x}}}_q^* = \left(\sum_{i=1}^{l} + \sum_{i=n-l+1}^{n}\right)x_{(i)q}/(2l)$, $\bar{x}_{(u)q} = \sum_{i=n-l+1}^{n}x_{(i)q}/l$ and $\bar{x}_{(v)q} = \sum_{i=1}^{l}x_{(i)q}/l$.From Equation (27),

$$\mathrm{var}(\tilde{\boldsymbol{x}}_{\boldsymbol{q}}) \geq \frac{l\left(\bar{x}_{(u)q} - \bar{x}_{(v)q}\right)^2}{2(k-1)}\left(\frac{x_{(n-l+1)q} - x_{(l)q}}{\bar{x}_{(u)q} - \bar{x}_{(v)q}}\right)^2 \tag{28}$$

Thus, one can have

$$|\tilde{\boldsymbol{X}}_{\boldsymbol{S}}^T\tilde{\boldsymbol{X}}_{\boldsymbol{S}}| \geq k(k-1)^h\lambda_{\min}^h(\boldsymbol{R})\prod_{q=1}^{h}\frac{l\left(\bar{x}_{(u)q} - \bar{x}_{(v)q}\right)^2}{2(k-1)}\left(\frac{x_{(n-l+1)q} - x_{(l)q}}{\bar{x}_{(u)q} - \bar{x}_{(v)q}}\right)^2 \tag{29}$$

$$= \frac{l^h}{2^h}k\lambda_{\min}^h(\boldsymbol{R})\prod_{q=1}^{h}\left(\bar{x}_{(u)q} - \bar{x}_{(v)q}\right)^2 \times \prod_{q=1}^{h}\left(\frac{x_{(n-l+1)q} - x_{(l)q}}{\bar{x}_{(u)q} - \bar{x}_{(v)q}}\right)^2 \tag{30}$$

This shows that

$$\frac{|\tilde{\boldsymbol{X}}_{\boldsymbol{S}}^T\tilde{\boldsymbol{X}}_{\boldsymbol{S}}|}{\frac{k^{h+1}}{4^h}\prod_{q=1}^{h}\left(\bar{x}_{(u)q} - \bar{x}_{(v)q}\right)^2} \geq \frac{\lambda_{\min}^h(\boldsymbol{R})}{h^h} \times \prod_{q=1}^{h}\left(\frac{x_{(n-l+1)q} - x_{(l)q}}{\bar{x}_{(u)q} - \bar{x}_{(v)q}}\right)^2 \tag{31}$$

□

A.2 | Proof of Theorem 2

Proof. Approximations of the lasso estimator of Equation (6) according to Tibshirani [10] is

$$\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S \approx \left[\tilde{\boldsymbol{X}}_{\boldsymbol{S}}^T\tilde{\boldsymbol{X}}_{\boldsymbol{S}} + \lambda\Psi(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S)\right]^{-1}\tilde{\boldsymbol{X}}_{\boldsymbol{S}}^T\tilde{\boldsymbol{y}} \tag{32}$$

$$\{\Psi(\tilde{\boldsymbol{\beta}}(\lambda)\} = \mathrm{diag}(\psi_1, \psi_2, \ldots, \psi_h) \tag{33}$$

$$\text{where } \psi_j = \begin{cases} \frac{1}{|\tilde{\beta}_j(\lambda)|} & \text{if } \tilde{\beta}_j(\lambda) \neq 0 \\ 0 & \text{otherwise} \end{cases}, \quad \text{for } j = 1, \ldots, h \tag{34}$$

The variance of the approximated lasso estimator is

$$\begin{aligned}&\mathrm{Var}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\\&\approx\sigma^2\left[\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S+\lambda\Psi\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right]^{-1}\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\left[\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S+\lambda\Psi\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right]^{-1}\end{aligned}\tag{35}$$

Denote $\boldsymbol{\beta}$ as the OLS estimator of Equation (6) when $\lambda=0$. Denote $\boldsymbol{W}=\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\left[\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S+\lambda\Psi\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right]^{-1}$.

$$\begin{aligned}&\mathrm{Var}\left(\boldsymbol{\beta}|\tilde{\boldsymbol{D}}_S\right)-\mathrm{Var}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\\&=\sigma^2\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}-\sigma^2\boldsymbol{W}^T\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\boldsymbol{W}\\&=\sigma^2\left[\boldsymbol{W}^T\left(\boldsymbol{W}^T\right)^{-1}\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\boldsymbol{W}^{-1}\boldsymbol{W}-\boldsymbol{W}^T\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\boldsymbol{W}\right]\\&=\sigma^2\boldsymbol{W}^T\left[\left(\boldsymbol{W}^T\right)^{-1}\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\boldsymbol{W}^{-1}-\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\right]\boldsymbol{W}\\&=\sigma^2\boldsymbol{W}^T\left[\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\left[\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S+\lambda\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right]\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\right.\\&\quad\left.\left[\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S+\lambda\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right]\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}-\left(\tilde{\boldsymbol{X}}_S^T-\tilde{\boldsymbol{X}}_S\right)^{-1}\right]\boldsymbol{W}\\&=\sigma^2\boldsymbol{W}^T\left[2\lambda\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\right.\\&\quad\left.+\lambda^2\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\right]\boldsymbol{W}\\&=\sigma^2\left[\tilde{\boldsymbol{X}}_T^S\tilde{\boldsymbol{X}}_S+\lambda\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right]^{-1}\left[2\lambda\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right.\\&\quad\left.+\lambda^2\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right]\\&\quad\left[\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S+\lambda\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right]^{-1}.\end{aligned}\tag{36}$$

When $\lambda>0$, for any $\upsilon\neq0$, $z=\left[\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S+\lambda\Psi\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right]^{-1}\upsilon\neq0$.

$$\begin{aligned}&\upsilon^T\left(\mathrm{Var}\left(\boldsymbol{\beta}|\tilde{\boldsymbol{D}}_S\right)-\mathbf{Var}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right)\upsilon\\&=\sigma^2z^T\left[2\lambda\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)+\lambda^2\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\right]z\\&=2\sigma^2\lambda z^T\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)z+\sigma^2\lambda^2z^T\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)\left(\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S\right)^{-1}\boldsymbol{\Psi}\left(\tilde{\boldsymbol{\beta}}(\lambda)|\tilde{\boldsymbol{D}}_S\right)z\\&>0.\end{aligned}\tag{37}$$

Equation (36) is positive definite, and it proves that the variance of the lasso estimator decreases as λ increases. Then, from Equations (20) and (21),

$$\mathrm{Var}\left(\boldsymbol{\beta}|\tilde{\boldsymbol{D}}_S\right))=\sigma^2\tilde{\boldsymbol{X}}_S^T\tilde{\boldsymbol{X}}_S^{-1}=\sigma^2\boldsymbol{B}^T\begin{bmatrix}\frac{1}{k}&\boldsymbol{0}^T\\\boldsymbol{0}&\frac{1}{k-1}\boldsymbol{R}^{-1}\end{bmatrix}\boldsymbol{B}\tag{38}$$

$$\mathrm{Var}\left(\tilde{\boldsymbol{\beta}}_{\boldsymbol{q}}|\tilde{\boldsymbol{D}}_S\right))\leq V\left(\boldsymbol{\beta}_{\boldsymbol{q}}|\tilde{\boldsymbol{D}}_S\right))=\frac{\sigma^2}{k-1}\frac{\left(\boldsymbol{R}^{-1}\right)_{qq}}{var\left(\tilde{\boldsymbol{x}}_{\boldsymbol{q}}\right)}\tag{39}$$

where $\left(\boldsymbol{R}^{-1}\right)_{qq}$ is the qth diagonal element of $\boldsymbol{R}^{-1}$. Denote the spectral decomposition of $\boldsymbol{R}$ as $\boldsymbol{R}=\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^T$. Since $\Lambda^{-1}\leq\lambda_{\min}^{-1}(\boldsymbol{R})\boldsymbol{I}_h$, $\boldsymbol{R}^{-1}=\boldsymbol{V}\Lambda^{-1}\boldsymbol{V}^T\leq\boldsymbol{V}\lambda_{\min}^{-1}(\boldsymbol{R})\boldsymbol{I}_h\boldsymbol{V}^T=\lambda_{\min}^{-1}(\boldsymbol{R})\boldsymbol{I}_h^T$. Thus, $\boldsymbol{R}_{qq}^{-1}\leq\lambda_{\min}^{-1}(\boldsymbol{R})$ for all j. Based on Equations (27) and (38),

$$\begin{aligned}V\left(\tilde{\boldsymbol{\beta}}_{\boldsymbol{q}}|\tilde{\boldsymbol{D}}_S\right)&\leq\frac{\sigma^2}{k-1}\frac{\left(\boldsymbol{R}^{-1}\right)_{qq}}{\mathrm{var}\left(\tilde{\boldsymbol{x}}_{\boldsymbol{q}}\right)}\leq\frac{4h\sigma^2}{k\lambda_{\min}(\boldsymbol{R})\left(\overline{x}_{(u)q}-\overline{x}_{(v)q}\right)^2}\\&\leq\frac{4h\sigma^2}{k\lambda_{\min}(\boldsymbol{R})\left(x_{(n-l+1)q}-x_{(l)q}\right)^2}\end{aligned}\tag{40}$$

With the fact that sum of squared differences from the sample mean is smaller than the sum of squared differences from other values,

$$\mathrm{var}\left(\tilde{\boldsymbol{x}}_{\boldsymbol{q}}\right)\leq\frac{1}{k-1}\sum_{i=1}^{k}\left(\tilde{x}_{iq}-\frac{x_{nq}+x_{1q}}{2}\right)^2\leq\frac{k}{4(k-1)}\left(x_{nq}-x_{1q}\right)^2\tag{41}$$

$$V\left(\tilde{\boldsymbol{\beta}}_{\boldsymbol{q}}|\tilde{\boldsymbol{D}}_S\right)\geq\frac{4\sigma^2}{k\lambda_{\max}(\boldsymbol{R})\left(x_{nq}-x_{1q}\right)^2}\tag{42}$$

In the cases when $\lambda=0$ and all variables are selected. When $\lambda\to\infty$, the lower bound of $V\left(\tilde{\boldsymbol{\beta}}_{\boldsymbol{q}}|\tilde{\boldsymbol{D}}_S\right)$ is close to 0 that none of variables are selected. □

A.3 | Proof of Theorem 3

We will use the result in Hwang [19] to prove Theorem 3.

Lemma 1 (Cauchy Interlace Theorem). *Let $\boldsymbol{A}$ be a Hermitan matrix of order* n *and let $\boldsymbol{B}$ be a principal submatrix of $\boldsymbol{A}$ of order $n-1$. If $\lambda_n\leq\lambda_{n-1}\cdots\lambda_2\leq\lambda_1$ lists the eigenvalues of $\boldsymbol{A}$ and $\mu_n\leq\mu_{n-1}\leq\cdots\leq\mu_3\leq\mu_2$ lists the eigenvalues of $\boldsymbol{B}$, then $\lambda_n\leq\mu_n\leq\lambda_{n-1}\leq\mu_{n-1}\cdots\leq\lambda_2\leq\mu_2\leq\lambda_1$.*

Proof of Theorem 3. Denote $\boldsymbol{R}$ the correlation matrix of $\tilde{\boldsymbol{D}}_S$ selected by the FAME algorithm. $\boldsymbol{R}_{\text{full}}$ as the correlation matrix of the full data $\boldsymbol{D}$. Let $\lambda_{\min}(\boldsymbol{R})\leq\lambda_{q-1}(\boldsymbol{R})\cdots\leq\lambda_2(\boldsymbol{R})\leq\lambda_{\max}(\boldsymbol{R})$ lists the eigenvalues of $\boldsymbol{R}$, and $\mu_{\min}(\boldsymbol{R}_{\text{full}})\leq\mu_{j-1}(\boldsymbol{R}_{\text{full}})\cdots\leq\mu_2(\boldsymbol{R}_{\text{full}})\leq\mu_{\max}(\boldsymbol{R}_{\text{full}})$ lists the eigenvalues of $\boldsymbol{R}_{\text{full}}$, where $q=1,\cdots,h$ and $j=1,\cdots,p$. Then

$$\mu_{\min}(\boldsymbol{R}_{\text{full}})\leq\lambda_{\min}(\boldsymbol{R})\leq\lambda_{\max}(\boldsymbol{R})\leq\mu_{\max}(\boldsymbol{R}_{\text{full}})\tag{43}$$

Thus, we can have

$$\frac{4\sigma^2}{k\lambda_{\max}(R)\left(x_{(n)q}-x_{(1)q}\right)^2}\geq\frac{4\sigma^2}{k\mu_{\max}(R_{\text{full}})\left(x_{(n)q}-x_{(1)q}\right)^2}\tag{44}$$

$$\frac{4h\sigma^2}{k\lambda_{\min}(R)\left(\overline{\tilde{\boldsymbol{x}}}_q^u-\overline{\tilde{\boldsymbol{x}}}_q^v\right)^2}\leq\frac{4h\sigma^2}{k\mu_{\min}(R_{\text{full}})\left(\overline{\tilde{\boldsymbol{x}}}_q^u-\overline{\tilde{\boldsymbol{x}}}_q^v\right)^2}\tag{45}$$

□