Received: 9 December 2020 | Accepted: 22 December 2020
DOI: 10.1002/sta4.352

WILEY

ORIGINAL ARTICLE

MuSP: A multistep screening procedure for sparse recovery

Yuehan Yang[1] | Ji Zhu[2] | Edward I. George[3]

[1]Department of Statistics, Central University of Finance and Economics, Beijing, 100081, China
[2]Department of Statistics, University of Michigan, Ann Arbor, 48109, Michigan, USA
[3]Department of Statistics, The Wharton School, University of Pennsylvania, Philadelphia, 19104, Pennsylvania, USA

Correspondence
Yuehan Yang, Department of Statistics, Central University of Finance and Economics, Beijing, China.
Email: yyh@cufe.edu.cn

Funding information
Center for Hierarchical Manufacturing, National Science Foundation, Grant/Award Number: DMS-1916245; National Natural Science Foundation of China, Grant/Award Number: 12001557

We propose a multistep screening procedure (MuSP) for the recovery of sparse linear models in high-dimensional data. This method is based on a repeated small penalty strategy that quickly converges to an estimate within a few iterations. Specifically, in each iteration, an adaptive lasso regression with a small penalty is fit within the reduced feature space obtained from the previous step, rendering its computational complexity roughly comparable with the Lasso. MuSP is shown to select the true model under complex correlation structures among the predictors and response, even when the irrepresentable condition fails. Further, under suitable regularity conditions, MuSP achieves the optimal minimax rate $(q \log n/n)^{1/2}$ for the upper bound of l_2-norm error. Numerical comparisons show that the method works effectively both in model selection and estimation, and the MuSP fitted model is stable over a range of small tuning parameter values, eliminating the need to choose the tuning parameter by cross-validation. We also apply MuSP to financial data and show that MuSP is successful in asset allocation selection.

KEYWORDS
high-dimensional data, iterative algorithm, Lasso, multistep method

1 | INTRODUCTION

Sparse recovery is of paramount interest in high-dimensional statistical problems where many predictors are available yet the regression function is well approximated by a few relevant covariates. A seminal contribution to this endeavour, the Lasso (Tibshirani, 1996) simultaneously performs model selection and parameter estimation through regularization with a convex penalty. Now widely used for sparse recovery in practice, further extensions of the Lasso have enhanced its applicability and offered some theoretical guarantees; for example, see Bühlmann and Van De Geer (2011), Efron et al. (2004), Fan and Lv (2010), Friedman et al. (2010), Hastie et al. (2015), Meinshausen and Bühlmann (2006), Zhao and Yu (2006), and Zou (2006).

Although convex regularization methods such as the Lasso are computationally attractive and enjoy great performance in prediction, they also lead to biased estimates and require rather restrictive conditions on the design matrix to obtain model selection consistency. Nonconvex penalization procedures such as SCAD (Fan & Li, 2001), MCP (Zhang, 2010a) and the Spike-and-Slab Lasso (SSL) (Ročková & George, 2018) have been proposed to lessen the bias. Multistep methods do this too, including Zhang (2010b) who proposed the Capped-l_1 regularization, leading to a multistep convex relaxation scheme, which is shown to obtain the correct feature set after a certain number of iterations. Zou and Li (2008) proposed a unified algorithm based on the local linear approximation (LLA) for maximizing the penalized likelihood, presenting a one-step low-dimensional asymptotic analysis for justification. Fan et al. (2014) provided a unified theory to show how to obtain the oracle solution via LLA. The theoretical properties of LLA highly rely on the initial estimates. Bühlmann and Meier (2008) proposed a method called multistep adaptive lasso (MSA-Lasso), which updates the adaptive weights and re-estimates *the entire set of regression coefficients* at each iteration until convergence. Huang and Zhang (2012) showed that, under certain conditions, the multistep framework can improve the solution quality. Further work focusing on multistep methods includes Liu et al. (2016), Wang et al. (2013), and Zhang and Zhang (2012).

In spite of the fact that most nonconvex penalties do not require the irrepresentable condition to achieve model selection consistency (Fan & Li, 2001; Zhang, 2010a), identifying the relevant predictors in the presence of highly collinear predictors may still present numerical challenges, as shown in Sections 4 and 5. Indeed, nonconvex penalties can introduce numerical difficulties in fitting models, becoming less computationally efficient than convex optimization problems.

The main thrust of this paper is to propose a multistep screening procedure (MuSP), a simple multistep method with the following characteristics:

(1) MuSP applies a single small penalty parameter, which remains fixed throughout, to minimize bias at each iteration. At each step, the active set is shrunk by deleting the "useless" variables whose coefficients have been thresholded to 0. When dealing with high-dimensional data, this strategy will start off with a large model with many possibly incorrect variables and iteratively distinguish the nonzeros from zeros.
(2) This backward deletion strategy of MuSP significantly reduces the execution time of the multistep method. As will be seen in the simulations, the computational complexity of MuSP is roughly comparable with the solution path of Lasso.
(3) With an inherently small estimation error bound, MuSP successfully recovers the true underlying sparse model even when the irrepresentable condition is relaxed. Indeed, MuSP remains effective even when the irrelevant variables are strongly correlated with the relevant variables. Note that although many nonconvex methods do not require restrictive conditions on the design matrix in theory, they may still have difficulty in selecting the right model with finite samples. MuSP is much better able to deal with such data.
(4) It is seen in simulations that the MuSP fitted model is stable over a range of small tuning parameter values, eliminating the need to choose the tuning parameter by cross-validation. The solution of this method is both sparse and stable.

This paper is organized as follows. Section 2 presents the method and discusses its relationship to other methods. Section 3 shows its theoretical properties. The simulations in Section 4 and the application in Section 5 assess the performance of the proposed method and compare it with several existing methods. Technical details are provided in the Supporting Information.

2 | METHOD

In this section, we present the details of the MuSP algorithm and compare it with existing methods. We consider the linear regression problem:

$$y = X\beta + \epsilon,$$

where y is an n response vector, X is an $n \times p$ matrix, β is a vector of regression coefficients, and ϵ is the error vector. We are particularly interested in the case where the number of parameters greatly exceeds the number of observations ($n \ll p$). We consider the q-sparse model, where β has at most q nonzero elements. Components of the error vector ϵ are independently distributed from $N(0, \sigma^2)$. The data and coefficients are allowed to change as n grows; meanwhile, p and q are allowed to grow with n. For notational simplicity, we do not index them with n.

Recall that the Lasso estimator (Tibshirani, 1996) minimizes the squared error loss regularized with the l_1-penalty. Compared with least squares, Lasso shrinks a particular set of coefficients to zero while shrinking the others towards zero. These two effects, model selection and shrinkage estimation, are controlled only by a single tuning parameter, leading to its well-known estimation bias. Although Zhao and Yu (2006) and Meinshausen and Bühlmann (2006) proved that the Lasso is model selection consistent under an irrepresentable condition, the condition is, however, quite restrictive. To mitigate these drawbacks, we propose MuSP with two goals in mind: (1) recovery of the true sparse model when the irrepresentable condition fails and (2) "almost unbiased estimation" by lowering the influence of the shrinkage penalty.

The essential idea behind MuSP is to provide more precise estimation through iterated penalization with a smaller tuning parameter that is less influential at each step. More precisely, the MuSP algorithm proceeds as follows.

- Initialize $k = 1$. Obtain a lasso solution $\hat{\beta}^{[1]}(\lambda_0)$:

$$\hat{\beta}^{[1]} := \arg\min \left\{ \frac{1}{2}\|y - X\beta\|_2^2 + \lambda_0 \|\beta\| \right\},$$

  and let $\mathcal{A}^{[1]}$ be the nonzero index set of $\hat{\beta}^{[1]}$; that is, $\mathcal{A}^{[1]} = \{j \in \{1, \ldots, p\} : \hat{\beta}_j^{[1]} \neq 0\}$.
- Repeat the following steps until convergence:

$$k \leftarrow k + 1,$$
$$\hat{\beta}^{[k]} := \underset{\beta_{(\mathcal{A}^{[k-1]})^c}=0}{\arg\min} \left\{ \frac{1}{2}\|y - X\beta\|_2^2 + \lambda \sum_{j \in \mathcal{A}^{[k-1]}} |\beta_j / \hat{\beta}_j^{[k-1]}| \right\},$$

  where the active set $\mathcal{A}^{[k]}$ is updated in every step; that is, $\mathcal{A}^{[k]} = \{j \in \{1, \ldots, p\} : \hat{\beta}_j^{[k]} \neq 0\}$.

At convergence, denote the active set by $\mathcal{A}$ and the solution by $\hat{\beta}$. Note that the active sets obtained during the iterations are nested; that is,

$$\mathcal{A}^{[1]} \supseteq \mathcal{A}^{[2]} \supseteq \ldots \supseteq \mathcal{A}^{[k]} \supseteq \ldots \supseteq \mathcal{A},$$

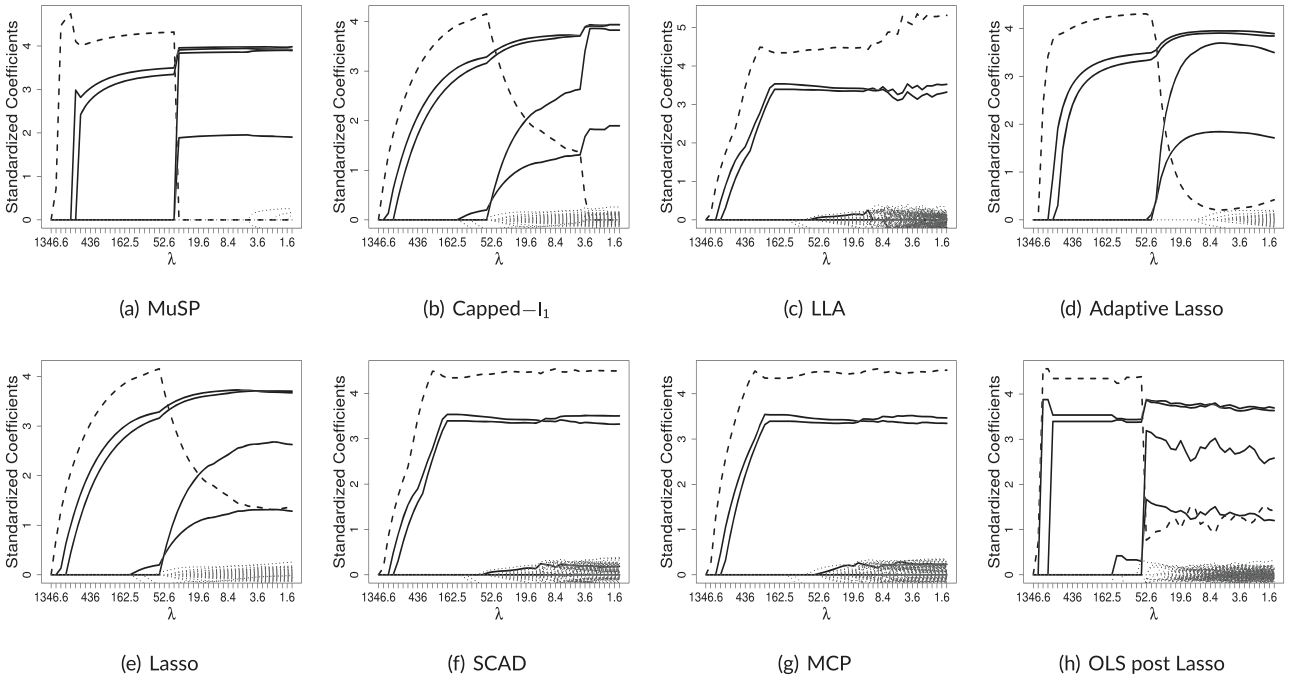


FIGURE 1 An example to illustrate eight methods' (in)consistency in model selection. The solid lines stand for the relevant variables; the dashed line stands for the variable that is irrelevant but highly correlated with the relevant variables; the dotted lines stand for other irrelevant variables

as in each iteration an adaptive lasso is fit using only the features selected by the previous step. This is key to control the computational time of the algorithm as well as to maintain a rather small tuning parameter λ. We will provide more details on the choice of this small tuning parameter in the theoretical results.

We use a simple example to demonstrate that many existing methods may not work as well as MuSP when irrelevant variables are highly correlated with the relevant variables. Set $(n, p) = (200, 400)$ and β with four nonzero entries. In this example, there exists a variable that is irrelevant but highly correlated with the relevant variables; hence, the irrepresentable condition fails. Figure 1 shows eight methods' (in)consistency in model selection: MuSP, Lasso (Tibshirani, 1996), LLA (Fan et al. 2014; Zou & Li, 2008), MCP (Zhang, 2010a), SCAD (Fan & Li, 2001), adaptive Lasso (Zou, 2006), OLS post Lasso (Belloni & Chernozhukov, 2013), and Capped−l_1 (Zhang, 2010b). As shown in Figure 1, except for MuSP, all other methods pick up this irrelevant variable first and never shrink it back to zero. MuSP performs similarly when λ is large, but when λ is small, MuSP obtains stable and accurate estimates and selects the right model. More details of this data example with further comparisons can be found in the simulation studies in Section 4.

2.1 | Relationship to other methods

There are many widely used methods that estimate regression coefficients for sparse linear models well. In this section, we analyse the MuSP solution and describe how our approach differs from these methods, more specifically, how MuSP can select the right model when there exist strong correlations between the irrelevant and relevant variables.

Normally, λ is the key to control the amount of regularization, but the proposed method intends to use the iterations to do the controlling rather than using λ. Note for any y, X, and λ, the solution of the kth iteration of MuSP is given by

$$\hat{\beta}_{(\mathcal{A}^{[k]})^c} = 0 \text{ and } \hat{\beta}_{\mathcal{A}^{[k]}} = \left(X_{\mathcal{A}^{[k]}}^{\mathsf{T}} X_{\mathcal{A}^{[k]}}\right)^{-1} \left(X_{\mathcal{A}^{[k]}}^{\mathsf{T}} y - \frac{\lambda s_{\mathcal{A}^{[k]}}}{\hat{\beta}_{\mathcal{A}^{[k]}}^{[k-1]}}\right),$$

where $s_{\mathcal{A}^{[k]}}$ is the vector of signs of $\hat{\beta}_{\mathcal{A}^{[k]}}$, and the equation on the right may be expressed as

$$\hat{\beta}_{\mathcal{A}^{[k]}} = \hat{\beta}_{\mathcal{A}^{[k]}}^{\text{ols}} - \frac{\lambda\left(X_{\mathcal{A}^{[k]}}^{\mathsf{T}} X_{\mathcal{A}^{[k]}}\right)^{-1} s_{\mathcal{A}^{[k]}}}{\hat{\beta}_{\mathcal{A}^{[k]}}^{[k-1]}}, \tag{1}$$

where $\hat{\beta}^{\text{ols}}_{\mathcal{A}^{[k]}}$ is the OLS estimator on the set $\mathcal{A}^{[k]}$. For $j = 1, \ldots, p$, we make the following notes on the relevant and the irrelevant predictors, respectively:

- Assume $\beta_j = 0$ and the first step of the MuSP algorithm (i.e., Lasso) did not shrink its estimate to zero. Reviewing the estimation properties of the Lasso (Meinshausen & Yu, 2009; Negahban et al. 2012), under the restricted eigenvalue condition and the proper choice of λ, this first iteration estimate is bounded by

$$|\hat{\beta}^{[1]}_j| = O((\log p/n)^{1/2})$$

  with high probability. At the same time, it is not difficult to verify that $|\hat{\beta}^{\text{ols}}_j|$ has the same bound. According to (1), given λ, the penalty term for the jth variable increases at the rate $(n/\log p)^{1/2}$ while $\hat{\beta}^{\text{ols}}_j$ is bounded by $M(\log p/n)^{1/2}$ with some positive constant M. Thus, the associated variable will be deleted from the active set in a finite number of steps, which we have found is typically few.
- Assume $\beta_j \neq 0$ satisfying the Beta-min condition for the nonzero coefficients (see 2 in the next section). Following the above argument, there will be a gap between its estimate and 0. Because $\hat{\beta}^{\text{ols}}_j$ is bounded away from zero and the penalty term will change little after several iterations, the algorithm will stabilize when all the irrelevant variables have been deleted.

To explain the difference between our method and others, we consider two examples, MCP (Zhang, 2010a) and LLA (Fan et al. 2014; Zou & Li, 2008) for illustration. For the MCP, the method essentially uses a large penalty for variables whose estimated coefficients are close to zero and no penalty when the estimated coefficients are large. In the high-dimensional setting, however, by chance there often exist a few irrelevant variables whose estimated coefficients are not close to zero, and this is especially the case when the irrelevant variables are strongly correlated with the relevant variables. In practice, under such situations, a one-step procedure is often not sufficient to remove all irrelevant variables while keeping all relevant variables. See Figure 1. LLA, on the other hand, is an iterative method, but in each iteration, the method deals with the entire set of predictors, and since the number of irrelevant variables is always much larger than that of relevant variables, the iteration does not really help in terms of choosing an appropriate value of the tuning parameter in comparison with one-step methods, especially when irrelevant variables are strongly correlated with relevant variables.

3 | THEORETICAL RESULTS

We first define some notation. Without loss of generality, we assume the columns of X are standardized: $X^T\mathbb{1} = 0$ and $X_j^T X_j/n = 1$ for $j = 1, \ldots, p$. Let $S \equiv \{j \in \{1, \ldots, p\} : \beta_j \neq 0\}$, $|S| = q$; let $C = X^T X/n$, $C_{SS} = X_S^T X_S/n$ and $C_{S^cS} = X_{S^c}^T X_S/n$. To state our theoretical results, we need the following assumption.

(C.1) Restricted eigenvalue (RE) condition: there exists a positive constant K_2 that

$$v^T C v \geqslant K_2 \|v\|_2^2,$$

for all $v \in G(S)$ where $G(S) := \{v \in \mathbb{R}^p : \|v_{S^c}\|_1 \leqslant 3\|v_S\|_1\}$.

(C.1) is usually used to bound the l_2-error between the coefficients and the estimates (Meinshausen & Yu, 2009; Negahban et al. 2012) and is also the least restrictive condition of similar types, for example, the restricted isometry property (Candes & Tao, 2007) and the partial Riesz condition (Zhang & Huang, 2008). It has been proved that (C.1) holds with high probability for quite general classes of Gaussian matrices for which the predictors may be highly correlated, in which case the irrepresentable condition or the restricted isometry condition may be violated with high probability (Raskutti et al. 2010).

We consider the following dimensions, in particular $p = O(\exp(n^{c_1}))$ and $q = O(n^{c_2})$ where $c_2 < 1/3$ and $0 \leqslant c_1 + c_2 < 1$. As a preparatory result, the following proposition shows that the first step of MuSP selects an active set $\mathcal{A}^{[1]}$ containing the true set with high probability.

Proposition 1. *Suppose (C.1) holds. Set $\lambda_0 = 4\sigma(n \log p)^{1/2}$. Assume there exists a positive constant K_1 such that*

$$\min_{j \in S} |\beta_j| > K_1 \sqrt{q} \lambda_0/\sqrt{n}. \tag{2}$$

Considering the first step of MuSP, $\hat{\beta}^{[1]}(\lambda_0)$ and the corresponding set $\mathcal{A}^{[1]}$, we have

$$P(S \subseteq \mathcal{A}^{[1]}) \geqslant 1 - 1/p. \tag{3}$$

Remark 1. Note that (2) requires a small gap between β_S and 0. It allows $|\beta_j| \to 0$ when $n \to \infty$ but at a rate that can be distinguished. This is a condition that has been frequently used in the literature for proving model selection consistency, for example, Lasso (Zhao & Yu, 2006), Capped-l_1 (Zhang, 2010a), and LLA for sparse linear regression (Fan et al. 2014).

Remark 2. Because the first-step estimator of MuSP is the Lasso, Proposition 1 can be seen as proving a property for the Lasso estimator. We obtain this result by using the bound of l_2-norm error between β and $\hat{\beta}^{[1]}$, which is known from past works, for example, Meinshausen and Yu (2009) and Negahban et al. (2012). Proposition 1 supports the backward deletion strategy of MuSP, which removes the variables that do not belong to $\mathcal{A}^{[1]}$ as they are irrelevant with high probability.

Remark 3. We set $\lambda_0 = 4\sigma(n\log p)^{1/2}$, which is the same as that for Lasso in order to achieve the error bound, while Lasso's model selection consistency requires a larger tuning parameter, that is, $Kn^{(1+c_4)/2}$ where $c_1 < c_4$. Hence, when n is large, the estimation accuracy and selection consistency cannot hold at the same time for Lasso. We solve this problem using an iterative strategy.

Now we show results on the error bound and the sign consistency of MuSP.

Theorem 1. *Under the same conditions of Proposition 1. Set $\lambda = 4\sigma(n\log n)^{1/2}$. For $c_1 + c_2 \leqslant c_3 < 1$ and $(1+3c_2)/2 < c_3$, with probability at least $1 - 1/n$, the following error bounds for the estimate $\hat{\beta}$ hold,*

$$\|\hat{\beta} - \beta\|_2 \leqslant \frac{8\sigma}{K_2 \cdot K_3}\left(\frac{q\log n}{n^{c_3}}\right)^{1/2},$$

$$\|\hat{\beta} - \beta\|_1 \leqslant \frac{32\sigma \cdot q}{K_2 \cdot K_3}\left(\frac{\log n}{n^{c_3}}\right)^{1/2},$$

where $K_3 < K_1$, K_1 and K_2 are defined in (2) and (C.1), respectively. Further, we have

$$P(sign(\hat{\beta}) = sign(\beta)) \geqslant 1 - 1/n.$$

Remark 4. Note that λ and λ_0 have different orders under the assumption that $p = O(\exp(n^{c_1}))$. If we consider another high-dimensional setting, where $p = O(n)$, by setting $\lambda_0 = \lambda = 4\sigma(n\log n)^{1/2}$, we would have the same result as in Theorem 1. For simplicity, we use the same value for λ_0 and λ in simulation studies and empirical analysis.

Remark 5. The error bound of MuSP in (1) is influenced by the adaptive penalty. We allow β_j to converge to 0 in (2); for example, there exists c_3 such that the lower bound of β_j is $n^{(c_3-1)/2}$ for $j \in S$. As a consequence, $n^{c_3/2}$ dominates the denominator of the error bound of MuSP rather than $n^{1/2}$. When c_3 is close to 1, the l_2-norm error bound is close to the rate $(q\log n/n)^{1/2}$.

Considering the following dimensions: $p = O(\exp(n^{c_1}))$ and $q = O(n^{c_2})$, where $0 \leqslant c_1 + c_2 < 1$ and $0 < c_1 < 1/3$, Corollary 1 shows that the l_1-error and the l_2-error of the MuSP estimator achieve the rate $q(\log n/n)^{1/2}$ and $(q\log n/n)^{1/2}$, respectively.

Corollary 1. *Suppose (C.1) holds. For the nonzero coefficients, let $c = \min_{j\in S}|\beta_j|$ and assume $1/c < \infty$. Set $\lambda = 4\sigma(n\log n)^{1/2}$. With probability $1 - 1/n$, the following error bounds hold for $\hat{\beta}$:*

$$\|\hat{\beta} - \beta\|_2 \leqslant \frac{8\sigma}{cK_2}\left(\frac{q\log n}{n}\right)^{1/2},$$

$$\|\hat{\beta} - \beta\|_1 \leqslant \frac{32\sigma \cdot q}{cK_2}\left(\frac{\log n}{n}\right)^{1/2}.$$

Further, we have

$$P(sign(\hat{\beta}) = sign(\beta)) \geqslant 1 - 1/n.$$

Remark 6. Corollary 1 sets a lower bound for β_S where c is allowed to be any positive constant. This condition has also appeared frequently in the literature, for example, Huang, Ma, and Zhang (2008).

Remark 7. Note the Gaussian assumption on the error term in the linear regression model can be relaxed by a sub-Gaussian assumption. Specifically, there exist constants $K, k > 0$ such that for $i = 1, \ldots, n$,

$$P(|\epsilon_i| \geqslant t) \leqslant Ke^{-kt^2}, \forall t \geqslant 0.$$

4 | SIMULATION STUDIES

In this section, we use simulation studies to demonstrate the performance of the proposed method: (1) the first part illustrates MuSP's consistency in model selection; (2) the second part compares the performance of the proposed method with those of several existing methods and also analyses the stability of MuSP with respect to the tuning parameter λ; and (3) the third part evaluates the computational time of different methods.

Of the existing methods that are compared with the proposed method, we choose three one-step methods, Lasso (Tibshirani, 1996), SCAD (Fan & Peng, 2004), and MCP (Zhang, 2010a), and four multi(two)-step methods, adaptive Lasso (Zou, 2006) (denoted as Alasso), OLS post Lasso (denoted as Plasso) (Belloni & Chernozhukov, 2013), capped-l_1 (Zhang, 2010b), and LLA (Fan et al. 2014; Zou & Li, 2008). In addition, we also compare with a Bayesian method, SSL (Ročková & George, 2018). We use the R package SSLASSO to run SSL; the results of MCP, SCAD and LLA are obtained using the R ncvreg package (Breheny & Huang, 2011), and the results of other methods are based on the R glmnet package (Friedman et al. 2010).

We consider the following linear regression model for the simulation studies

$$y_i = \sum_{j=1}^{p} x_{ij}\beta_j + \epsilon_i,\ i = 1, \dots, n,$$

where x_{ij} are generated from a multivariate normal distribution $N(0, \Sigma)$ and ϵ_i is generated from $N(0, 1)$. Four regression coefficients are set as nonzero, specifically $(\beta_2, \beta_3, \beta_4, \beta_p) = (2, 4, 4, 4)$, and others are set to zero. We consider the case where some irrelevant variable is highly correlated with the relevant variables. Specifically, we set

$$x_{i1} = \frac{7}{8}x_{ip} + \frac{3}{8}x_{i2} + \frac{1}{8}x_{i3} + \frac{1}{8}x_{i4} + \frac{1}{8}x_{i5} + \frac{1}{8}x_{i6} + \frac{1}{8}x_{i7} + \frac{1}{8}e_i,$$

where e_i is generated from $N(0, 1)$. Denote the covariance matrix of the last $(p-1)$ variables as Σ_{-1}. We consider two scenarios: (1) $\Sigma_{-1} = \mathrm{I}$, and (2) $\Sigma_{jj'} = 0.5^{|j-j'|}$, where $j = 2, \dots, p$.

In both scenarios, the RE condition (C.1) holds while the irrepresentable condition fails. Recall the irrepresentable condition states that there exists a positive constant $\eta > 0$ such that

$$\|C_{S^cS}C_{SS}^{-1}\mathrm{sign}(\beta_S)\|_\infty \leqslant 1 - \eta.$$

With $X_S = (X_2, X_3, X_4, X_p)$ and $X_{S^c} = (X_1, X_5, \dots, X_{p-1})$, it is not difficult to check that the irrepresentable condition does not hold. Figure 1 in Section 2 shows the results from one typical simulation repetition when $(n, p) = (200, 400)$ under Scenario 1.

We compare both the estimation and selection performances of the nine methods mentioned above. The l_2-norm ($\|\hat{\beta} - \beta\|_2$) and the l_1-norm errors ($\|\hat{\beta} - \beta\|_1$) are computed. We also report the estimated number of nonzero coefficients (NZ), as well as the false positive rate (FPR) and the true positive rate (TPR), which are, respectively, defined as

$$\mathrm{FPR} = \frac{|j \in \{1, \dots, p\} : \hat{\beta}_j \neq 0 \text{ and } \beta_j = 0|}{|j \in \{1, \dots, p\} : \beta_j = 0|},$$
$$\mathrm{TPR} = \frac{|j \in \{1, \dots, p\} : \hat{\beta}_j \neq 0 \text{ and } \beta_j \neq 0|}{|j \in \{1, \dots, p\} : \beta_j \neq 0|}.$$

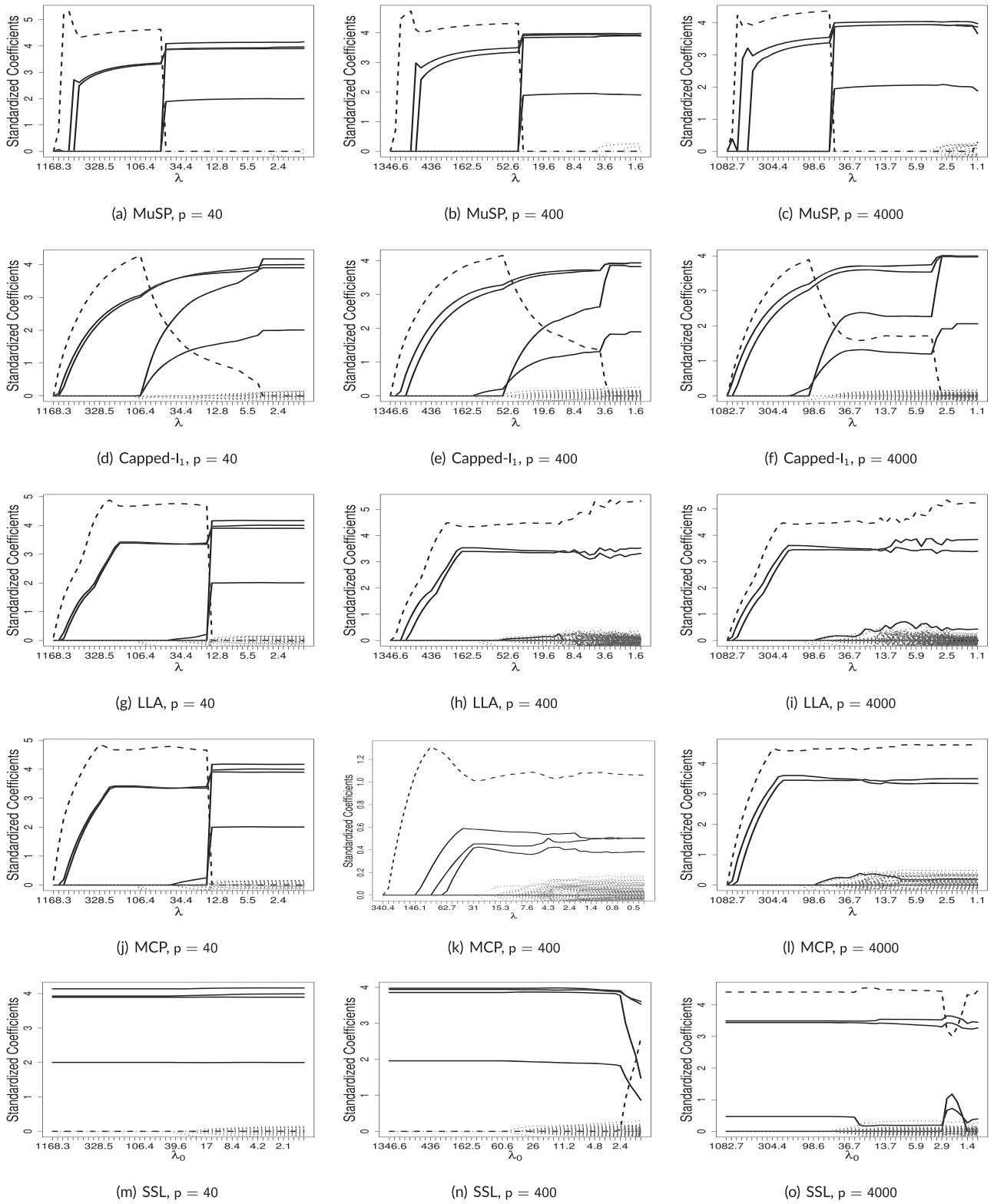
4.1 | Model selection

We consider three different dimensions, $p = 40$, 400, and 4000, and n is fixed to be 200. Due to lack of space, we only show representative results in the main manuscript and delay other results in the Supporting Information. Specifically, Figure 2 shows the comparison between the proposed MuSP, capped-l_1, LLA, MCP and SSL under Scenario 1, whereas Figure 3 shows the comparison between MuSP and Lasso, PLasso, and SCAD under Scenario 2. As we can see in Figure 2, when λ is large, the first four methods select the irrelevant variable X_1 (as it is highly correlated with the relevant variables and the response). When λ decreases, in the case of relatively low dimension (left column), LLA and MCP are able to shrink the estimated coefficient for X_1 to zero but at the same time select many other irrelevant variables, whereas in the case of relatively high dimension (middle and right columns), LLA and MCP are not even able to shrink the estimated coefficient for X_1 to zero. Capped-l_1 performs slightly better as it is able to shrink the estimated coefficient for X_1 to zero in both low- and high-dimensional settings when λ is very small but at the same time also selects many irrelevant variables. SSL chooses the correct model when $p = 40$ and 400; however, when p becomes larger, SSL always selects X_1 as an important variable. As a comparison, MuSP chooses the exact correct model over a wide range of small values of λ in all settings.

The results in Figure 3 are similar as in Scenario 1: Lasso, PLasso, and SCAD are able to shrink the estimated coefficient for X_1 to zero when λ is small and the dimension is relatively low, but at the same time select many other irrelevant variables, and completely fail to shrink the

estimated coefficient for X_1 to zero when the dimension is relatively high. The proposed MuSP is again able to identify the correct model when λ is relatively small in all three considered dimensional settings.

We also want to note that, as one can see in both Figures 2 and 3, the MuSP solution is quite stable over a wide range of small values of λ. This implies that MuSP requires little tuning, which is a convenient and useful property in practice and different from many other regularization methods that require careful selection of the tuning parameter.



(a) MuSP, p = 40 (b) MuSP, p = 400 (c) MuSP, p = 4000

(d) Capped-l_1, p = 40 (e) Capped-l_1, p = 400 (f) Capped-l_1, p = 4000

(g) LLA, p = 40 (h) LLA, p = 400 (i) LLA, p = 4000

(j) MCP, p = 40 (k) MCP, p = 400 (l) MCP, p = 4000

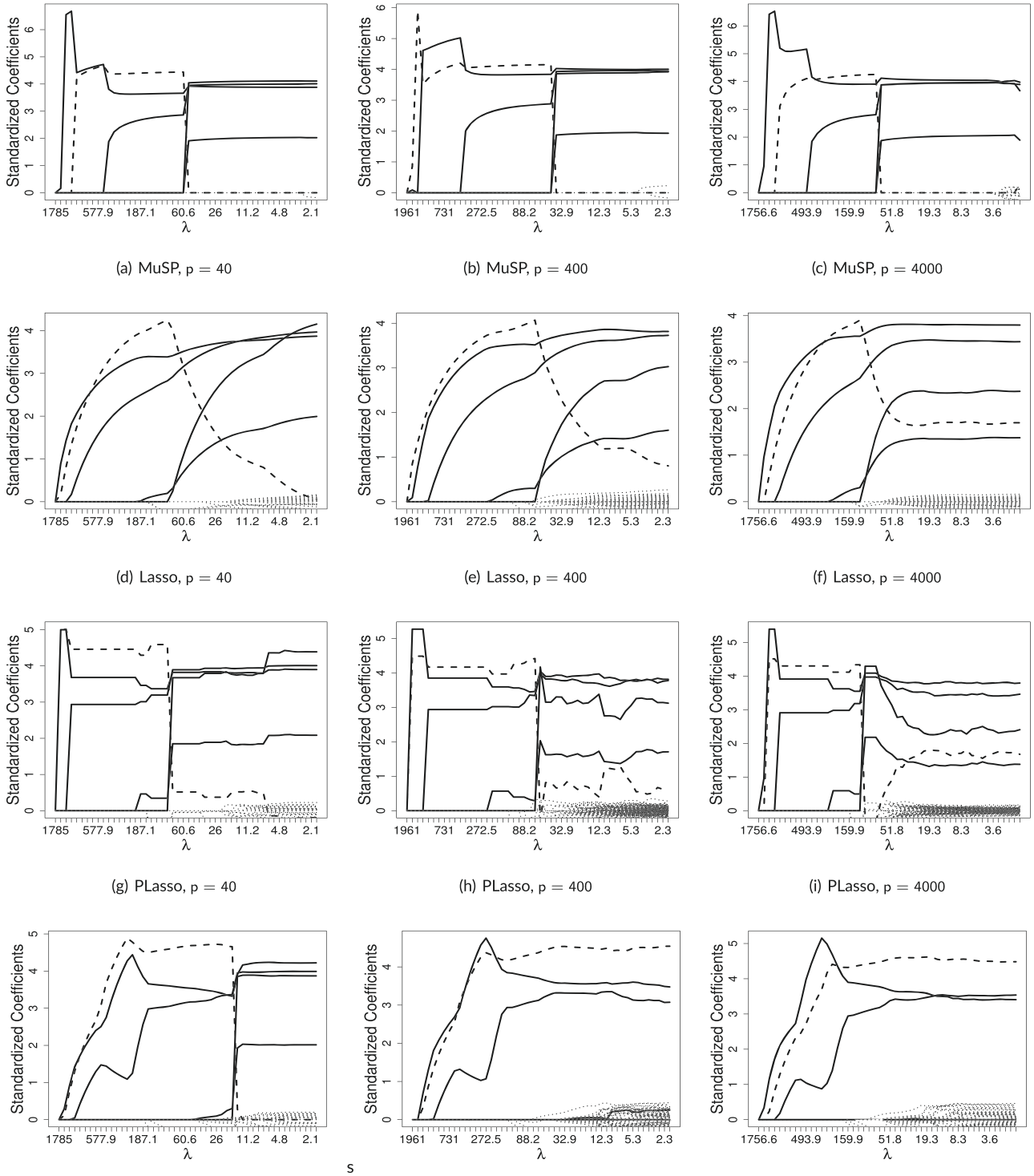
(m) SSL, p = 40 (n) SSL, p = 400 (o) SSL, p = 4000

FIGURE 2 Results for Scenario 1 under three different dimensions. The dashed line corresponds to X_1, which is irrelevant; the dotted lines correspond to other irrelevant variables; the solid lines correspond to the relevant variables

4.2 | Performance comparison

We consider two settings, $(n, p) = (100, 50)$ and $(n, p) = (100, 200)$. While cross-validation is not needed for SSL, the tuning parameter for all the other methods is selected using 10-fold cross-validation. Each simulation is repeated 100 times. The results are summarized in Tables 1 and 2. It can be seen that MuSP uniformly outperforms other methods in both estimation accuracy and model selection.

Figure 4 shows the estimation error of MuSP when λ varies under the same simulation setting of Table 2, that is, Scenario 1 with $(n, p) = (200, 40)$, $(200, 400)$, and $(200, 4000)$. As one can see, the estimation error of MuSP is low and stable over a range of small λ values. The results under Scenario 2 are similar and thus omitted. This implies that cross-validation may not be necessary for MuSP in practice; setting λ at



(a) MuSP, p = 40 (b) MuSP, p = 400 (c) MuSP, p = 4000

(d) Lasso, p = 40 (e) Lasso, p = 400 (f) Lasso, p = 4000

(g) PLasso, p = 40 (h) PLasso, p = 400 (i) PLasso, p = 4000

(j) SCAD, p = 40 (k) SCAD, p = 400 (l) SCAD, p = 4000

FIGURE 3 Results for Scenario 2 under three different dimensions. The dashed line corresponds to X_1, which is irrelevant; the dotted lines correspond to other irrelevant variables; the solid lines correspond to the relevant variables

TABLE 1 Performance comparison under Scenario 1

Method	l_2-error	l_1-error	NZ	FPR	TPR
		(n, p)	=	(100, 50)	
MuSP	0.19 (0.07)	0.33 (0.12)	4.00 (0.00)	0.00 (0.00)	1.00 (0.00)
Lasso	2.05 (0.81)	2.16 (0.40)	19.47 (2.98)	0.34 (0.06)	1.00 (0.00)
MCP	6.44 (0.12)	3.61 (0.06)	7.74 (1.22)	0.11 (0.02)	0.63 (0.13)
SCAD	6.40 (0.11)	3.56 (0.06)	8.14 (1.29)	0.12 (0.03)	0.65 (0.12)
SSL	0.31 (0.61)	0.73 (1.26)	9.94 (3.53)	0.13 (0.08)	1.00 (0.05)
ALasso	0.38 (0.35)	0.79 (0.33)	5.56 (1.06)	0.03 (0.02)	1.00 (0.00)
PLasso	1.13 (1.29)	1.46 (0.63)	8.46 (1.45)	0.10 (0.03)	0.99 (0.05)
Capped-l_1	2.05 (0.81)	2.16 (0.40)	19.47 (2.98)	0.34 (0.06)	1.00 (0.00)
LLA	6.40 (0.11)	3.56 (0.06)	8.14 (1.29)	0.12 (0.03)	0.65 (0.12)
		(n, p)	=	(100, 200)	
MuSP	0.25 (0.64)	0.44 (1.32)	4.02 (0.20)	0.00 (0.00)	1.00 (0.05)
Lasso	3.50 (0.88)	7.66 (1.83)	21.53 (3.48)	0.09 (0.02)	1.00 (0.00)
MCP	6.51 (0.70)	14.33 (1.65)	18.00 (2.66)	0.08 (0.01)	0.64 (0.13)
SCAD	6.50 (0.22)	13.88 (0.74)	21.55 (3.73)	0.10 (0.02)	0.66 (0.12)
SSL	2.77 (3.01)	6.41 (6.51)	24.88 (8.08)	0.11 (0.04)	0.88 (0.17)
ALasso	1.19 (1.36)	2.23 (2.58)	5.11 (0.96)	0.01 (0.01)	1.00 (0.03)
PLasso	4.89 (2.27)	9.66 (4.49)	6.01 (1.23)	0.02 (0.01)	0.76 (0.15)
Capped-l_1	3.50 (0.88)	7.66 (1.83)	21.53 (3.48)	0.09 (0.02)	1.00 (0.00)
LLA	6.50 (0.22)	13.88 (0.74)	21.55 (3.73)	0.10 (0.02)	0.66 (0.12)

TABLE 2 Performance comparison under Scenario 2

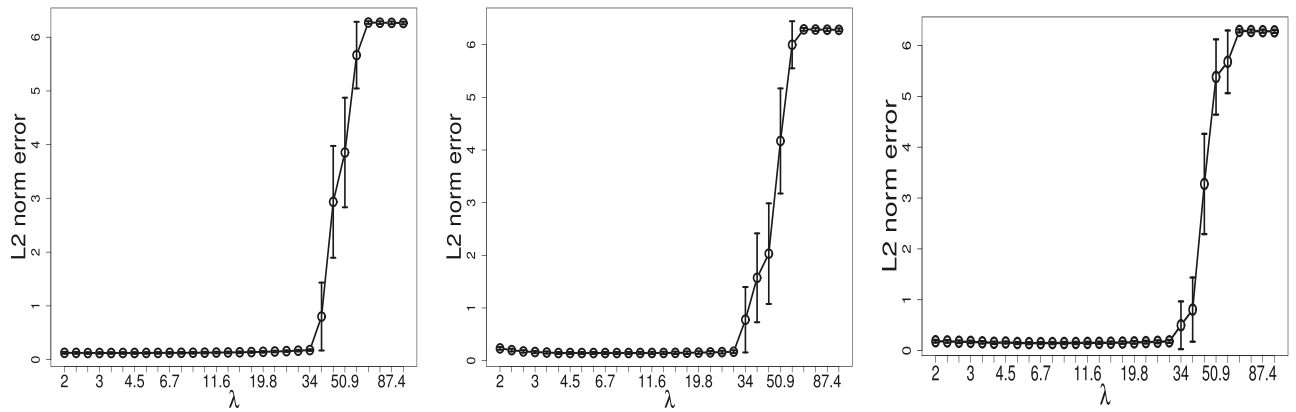
Method	l_2-error	l_1-error	NZ	FPR	TPR
		(n, p)	=	(100, 50)	
MuSP	0.22 (0.09)	0.37 (0.17)	4.00 (0.00)	0.00 (0.00)	1.00 (0.00)
Lasso	1.82 (0.78)	4.91 (1.74)	25.87 (3.14)	0.48 (0.07)	1.00 (0.00)
MCP	6.46 (0.19)	13.14 (0.50)	7.89 (1.52)	0.12 (0.03)	0.63 (0.13)
SCAD	6.43 (0.17)	13.44 (0.50)	13.60 (2.35)	0.24 (0.05)	0.69 (0.11)
SSL	0.23 (0.09)	0.43 (0.19)	5.67 (1.84)	0.04 (0.04)	1.00 (0.00)
ALasso	0.61 (0.55)	1.15 (1.06)	4.88 (0.90)	0.02 (0.02)	1.00 (0.00)
PLasso	1.14 (1.08)	2.63 (2.24)	9.18 (1.79)	0.11 (0.04)	0.99 (0.04)
Capped-l_1	1.82 (0.78)	4.91 (1.74)	25.87 (3.14)	0.48 (0.07)	1.00 (0.00)
LLA	6.43 (0.17)	13.44 (0.50)	13.60 (2.35)	0.24 (0.05)	0.69 (0.11)
		(n, p)	=	(100, 200)	
MuSP	0.28 (0.63)	0.49 (1.28)	4.01 (0.10)	0.00 (0.00)	1.00 (0.05)
Lasso	2.96 (1.11)	2.67 (0.43)	33.50 (4.38)	0.15 (0.02)	1.00 (0.03)
MCP	6.48 (0.12)	3.62 (0.05)	7.67 (1.74)	0.03 (0.01)	0.55 (0.10)
SCAD	6.45 (0.12)	3.59 (0.05)	8.66 (2.24)	0.03 (0.01)	0.56 (0.10)
SSL	1.55 (2.53)	3.24 (5.25)	9.73 (4.32)	0.03 (0.02)	0.93 (0.16)
ALasso	0.89 (1.28)	1.14 (0.71)	6.57 (1.77)	0.01 (0.01)	1.00 (0.04)
PLasso	2.01 (2.15)	1.95 (0.93)	10.43 (2.29)	0.03 (0.01)	0.95 (0.11)
Capped-l_1	2.96 (1.11)	2.67 (0.43)	33.50 (4.38)	0.15 (0.02)	1.00 (0.03)
LLA	6.45 (0.12)	3.59 (0.05)	8.66 (2.24)	0.03 (0.01)	0.56 (0.10)

an appropriately small value often works well; for example, we found $\lambda = (1/5)(n \log n)^{1/2}$ is a reasonable choice after standardizing the design matrix and the response variable.

4.3 | Computational cost

To compare the computational cost of different methods, we considering the following settings: $n = (100, 1000)$ and $p = (100, 1000, 10000, 100000)$ under Scenario 1. Each running time involves 100 different λ values covering a wide range. For MuSP, Lasso, ALasso, LLA, cappled-l_1, and MSA, we used the R glmnet package (Friedman et al. 2010); for SSL, we used the SSLASSO package (Ročková & George, 2018), and for SCAD and MCP, we used the R ncvreg package (Breheny & Huang, 2011).

Table 3 summarizes the results. As one can see, the computational cost of MuSP is in general larger than that of Lasso but becomes more comparable as both n and p increase. In comparison with nonconvex one-step methods, including MCP and SCAD, MuSP is slower when n and p are small but faster when n and p are large. Further, the computational cost of MuSP is much lower than that of SSL and those of other multi(two)-step methods, including the adaptive Lasso, LLA, capped-l_1 and MSA; this is because the MuSP only deals with the high-dimensional data in the first step, whereas other methods deal with the entire data set in every step.



(a) $(n, p) = (200, 40)$ (b) $(n, p) = (200, 400)$ (c) $(n, p) = (200, 4000)$

FIGURE 4 Illustration of the stability of MuSP with respect to λ under Scenario 1

TABLE 3 Comparison of average running time in seconds

(n, p)		Lasso	MCP	SCAD	MuSP	ALasso	SSL	LLA	Capped-l_1	MSA
$(10^2, 10^2)$	Mean	0.01	0.05	0.04	0.71	0.28	0.78	1.02	1.46	1.48
	SD	0.01	0.01	0.01	0.02	0.03	0.09	0.41	0.26	0.24
$(10^2, 10^3)$	Mean	0.02	0.04	0.07	0.58	0.89	0.77	2.01	4.64	4.04
	SD	0.01	0.01	0.01	0.02	0.07	0.42	0.14	0.75	0.87
$(10^2, 10^4)$	Mean	0.25	0.52	0.38	0.59	7.67	4.10	16.47	21.04	18.02
	SD	0.02	0.04	0.03	0.04	0.42	1.58	1.44	2.65	3.76
$(10^3, 10^3)$	Mean	0.33	4.08	4.83	2.03	7.31	76.15	95.76	216.24	24.16
	SD	0.02	0.60	0.58	0.09	0.51	3.57	6.13	53.04	4.84
$(10^3, 10^4)$	Mean	1.21	7.79	4.28	3.77	43.87	81.75	282.58	388.00	148.18
	SD	0.17	0.70	0.29	0.17	1.67	17.72	36.03	46.56	6.15
$(10^3, 10^5)$	Mean	10.96	33.22	32.30	14.48	644.87	302.12	1387	1508	1535
	SD	0.08	3.30	0.22	0.23	29.73	86.47	126	196	224

5 | EMPIRICAL ANALYSIS: INDEX TRACKING

In this section, we apply the proposed method to the important and useful index tracking problem in financial modelling. Roughly speaking, index tracking aims to replicate the movement of a financial index using a small set of financial assets, for example, stocks, and is the core of the index fund. This is a high-dimensional data modelling problem as the number of stocks that one can choose from is often on the order of hundreds or thousands, whereas the number of observations (days) is on the order of tens or hundreds. Further, due to transactional cost, one only wishes to select a few rather than many stocks (i.e., a sparse model) to mimic the behaviour of the index.

We consider the S&P500 index and the following model: $y_t = \sum_{j=1}^{p} \beta_j x_{jt} + \epsilon_t$, where y_t denotes the return of the S&P500 index on day t, x_{jt} denotes the return of stock j on day t and β_j is the weight of stock j. We consider 19 rolling periods from January 2016 till December 2017 and divide each period into training (=100 days) and testing (=20 days) parts. The training period is used to select stocks and estimate the corresponding β_j's and then the testing part is used to evaluate the performance.

We compare four methods, including MuSP, LLA, capped-l_1, and ALasso, as these four methods had better performances in simulation studies. To measure the performance of different methods, we use the tracking error (Meade & Salkin, 1989), which is a standard measure used in the financial industry to assess the performance of tracking. It is defined as

$$\text{TrackingError}_{\text{year}} = \sqrt{252} \times \sqrt{\sum (\text{err}_t - \text{mean}(\text{err}))^2/(T-1)},$$

where $\text{err}_t = y_t - \hat{y}_t$, with y_t and $\hat{y}_t$ being the daily return of the index and the daily return of the constructed index on day t, respectively.

We did not use the validation or cross-validation approach to select the tuning parameter; instead, we chose the tuning parameter for each method such that the number of selected stocks is 20, which is often the way done in practice. Figure 5 shows the 19 tracking errors for both training and testing sets over time. As can be seen, MuSP always produces lower and more stable errors than other methods, except for one rolling period.

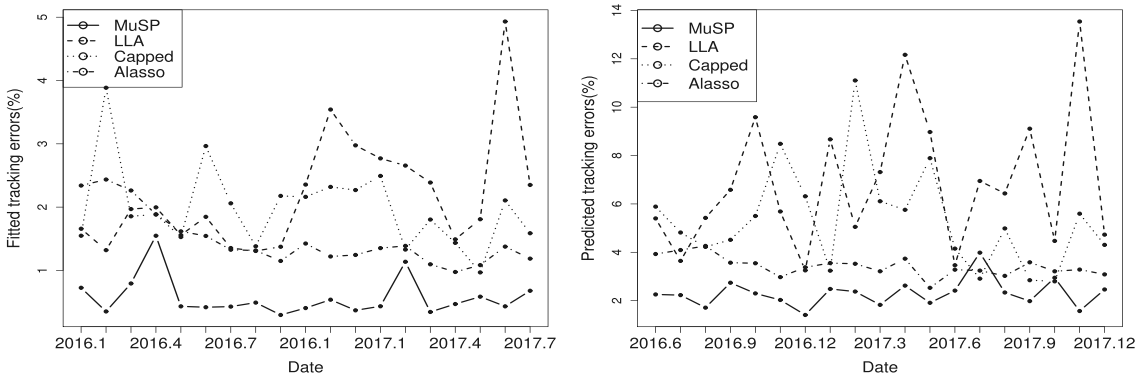


FIGURE 5 Tracking errors for both the training (left) and testing (right) sets by different methods

ACKNOWLEDGEMENTS

Yang's research was partially supported by the National Natural Science Foundation of China (Grant No. 12001557) and the Program for Innovation Research in Central University of Finance and Economics. George's research was partially supported by NSF Grant DMS-1916245.

CONFLICT OF INTEREST

The authors declare no potential conflict of interests.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Yuehan Yang https://orcid.org/0000-0001-8321-6859
Ji Zhu https://orcid.org/0000-0002-7812-5378

REFERENCES

Belloni, A., & Chernozhukov, V. (2013). Least squares after model selection in high-dimensional sparse models. *Bernoulli*, *19*, 521–547.

Breheny, P., & Huang, J. (2011). Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *Annals of Applied Statistics*, *5*, 232–253.

Bühlmann, P., & Meier, L. (2008). Discussion: One-step sparse estimates in nonconcave penalized likelihood models, by Zou, H. and Li, R. *Annals of Statistics*, *36*, 1534–1541.

Bühlmann, P., & Van De Geer, S. (2011). *Statistics for high-dimensional data: Methods, theory and applications*. New York: Springer Science & Business Media.

Candes, E., & Tao, T. (2007). The Dantzig selector: Statistical estimation when p is much larger than n. *Annals of Statistics*, *35*(6), 2313–2351.

Candès, E., & Plan, Y. (2009). Near-ideal model selection by ℓ_1 minimization. *Annals of Statistics*, *37*(5A), 2145–2177.

Efron, B., Hastie, T., Johnstone, L., & Tibshirani, R. (2004). Least angle regression. *Annals of Statistics*, *32*(2), 407–451.

Fan, J., & Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, *96*(456), 1348–1360.

Fan, J., & Lv, J. C. (2010). A selective overview of variable selection in high dimensional feature space. *Statistica Sinica*, *20*(1), 101.

Fan, J., & Peng, H. (2004). Nonconcave penalized likelihood with a diverging number of parameters. *Annals of Statistics*, *32*(3), 928–961.

Fan, J., Xue, L., & Zou, H. (2014). Strong oracle optimality of folded concave penalized estimation. *Annals of Statistics*, *42*(3), 819–849.

Friedman, J., Hastie, T., & Tibshirani, R. J. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, *33*(1), 1–22.

Hastie, T., Tibshirani, R., & Wainwright, M. (2015). *Statistical learning with sparsity: The lasso and generalizations*. London: Chapman and Hall/CRC.

Huang, J., Ma, S., & Zhang, C.-H. (2008). Adaptive lasso for sparse high-dimensional regression models. *Statistica Sinica*, *18*(4), 1603–1618.

Huang, J., & Zhang, C.-H. (2012). Estimation and selection via absolute penalized convex minimization and its multistage adaptive applications. *Journal of Machine Learning Research*, *13*(1), 1839–1864.

Liu, H., Yao, T., & Li, R. (2016). Global solutions to folded concave penalized nonconvex learning. *Annals of statistics*, *44*(2), 629.

Meade, N., & Salkin, G. R. (1989). Index funds-construction and performance measurement. *Journal of the Operational Research Society*, *40*(10), 871–879.

Meinshausen, N., & Bühlmann, P. (2006). High-dimensional graphs and variable selection with the lasso. *Annals of Statistics*, *34*(3), 1436–1462.

Meinshausen, N., & Yu, B. (2009). Lasso-type recovery of sparse representations for high-dimensional data. *Annals of Statistics*, *37*(1), 246–270.

Negahban, S., Ravikumar, P., Wainwright, M. J., & Yu, B. (2012). A unified framework for high-dimensional analysis of *m*-estimators with decomposable regularizers. *Statistical Science*, *27*(4), 1348–1356.

Raskutti, G., Wainwright, M. J., & Yu, B. (2010). Restricted eigenvalue properties for correlated Gaussian designs. *Journal of Machine Learning Research*, *11*, 2241–2259.

Ročková, V., & George, E. I. (2018). The spike-and-slab lasso. *Journal of the American Statistical Association*, *113*(521), 431–444.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B*, *58*(1), 267–288.

Tibshirani, R. J. (2013). The lasso problem and uniqueness. *Electronic Journal of Statistics*, *7*, 1456–1490.

Wainwright, M. J. (2009). Sharp thresholds for noisy and high-dimensional recovery of sparsity using l_1-constrained quadratic programming (lasso). *IEEE Transactions on Information Theory*, *55*(5), 2183–2202.

Wang, L., Kim, Y., & Li, R. (2013). Calibrating non-convex penalized regression in ultra-high dimension. *Annals of statistics*, *41*(5), 2505.

Zhang, C.-H. (2010a). Nearly unbiased variable selection under minimax concave penalty. *Annals of Statistics*, *38*(2), 894–942.

Zhang, C.-H., & Huang, J. (2008). The sparsity and bias of the lasso selection in high-dimensional linear regression. *Annals of Statistics*, *36*(4), 1567–1594.

Zhang, C.-H., & Zhang, T. (2012). A general theory of concave regularization for high-dimensional sparse estimation problems. *Statistical Science*, *27*(4), 576–593.

Zhang, T. (2010b). Analysis of multi-stage convex relaxation for sparse regularization. *Journal of Machine Learning Research*, *11*, 1081–1107.

Zhao, P., & Yu, B. (2006). On model selection consistency of lasso. *Journal of Machine Learning Research*, *7*, 2541–2563.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, *101*(476), 1418–1429.

Zou, H., & Li, R. (2008). One-step sparse estimates in nonconcave penalized likelihood models. *Annals of Statistics*, *36*(4), 1509–1533.

How to cite this article: Yang Y, Zhu J, George EI. MuSP: A multistep screening procedure for sparse recovery. *Stat*. 2021;10:e352. https://doi.org/10.1002/sta4.352

APPENDIX A: PROOF OF MAIN THEOREMS

Let the vector $\hat{\beta}^{[1]}$ be the solution to

$$\hat{\beta}^{[1]} := \arg\min \left\{ \frac{1}{2}\|y - X\beta\|_2^2 + \lambda_0\|\beta\|_1 \right\}.$$

We set $\hat{u}^{[1]} = n^{1/2}(\hat{\beta}^{[1]} - \beta)$ and $W = X^{\mathsf{T}}\epsilon/n^{1/2}$. Let $\hat{u}_S^{[1]}$, $\hat{\beta}_S^{[1]}$, β_S, W_S and $\hat{u}_{S^c}^{[1]}$, $\hat{\beta}_{S^c}^{[1]}$, β_{S^c}, W_{S^c} denote the S and S^c entries of $\hat{u}^{[1]}$, $\hat{\beta}^{[1]}$, β and W, respectively. We first provide the following lemma that shows $\hat{u}^{[1]} \in G(S)$ needed for (C.1).

Lemma 1. *Assume ϵ_i are i.i.d. Gaussian random variables with mean 0 and variance σ^2, $i = 1, \ldots, n$. Conditional on*

$$\left\{ 2\|W\|_\infty \leqslant \frac{\lambda_0}{n^{1/2}} \right\},$$

we have

$$\|\hat{u}_{S^c}^{[1]}\|_1 \leqslant 3\|\hat{u}_S^{[1]}\|_1.$$

Proof of Lemma 1. Based on the definition of $\hat{\beta}^{[1]}$, we have the following inequality:

$$\frac{1}{2}\|y - X\hat{\beta}^{[1]}\|_2^2 + \lambda_0\|\hat{\beta}^{[1]}\|_1 \leqslant \frac{1}{2}\|y - X\beta\|_2^2 + \lambda_0\|\beta\|_1.$$

We also have

$$\|\hat{\beta}^{[1]}\|_1 = \|\hat{\beta}_S^{[1]}\|_1 + \|\hat{\beta}_{S^c}^{[1]}\|_1$$

and

$$\|\hat{\beta}_S^{[1]}\|_1 + \|\hat{\beta}_S^{[1]} - \beta_S\|_1 \geqslant \|\beta_S\|_1.$$

By the three inequalities above, the following inequality holds:

$$\|X(\hat{\beta}^{[1]}-\beta)\|_2^2+2\lambda_0\|\hat{\beta}^{[1]}_{S^c}\|_1\leqslant 2\epsilon^{\mathsf{T}}X(\hat{\beta}^{[1]}-\beta)+2\lambda_0\|\hat{\beta}^{[1]}_S-\beta_S\|_1. \tag{A1}$$

Further, conditional on $\{2\|W\|_\infty\leqslant\lambda_0/n^{1/2}\}$, we have

$$2\epsilon^{\mathsf{T}}X(\hat{\beta}^{[1]}-\beta)\leqslant\lambda_0\|\hat{\beta}^{[1]}-\beta\|_1=\lambda_0\|\hat{\beta}^{[1]}_S-\beta_S\|_1+\lambda_0\|\hat{\beta}^{[1]}_{S^c}\|_1. \tag{A2}$$

Combining (A1) and (A2), we have

$$\|X(\hat{\beta}^{[1]}-\beta)\|_2^2+2\lambda_0\|\hat{\beta}^{[1]}_{S^c}\|_1\leqslant 3\lambda_0\|\hat{\beta}^{[1]}_S-\beta_S\|_1+\lambda_0\|\hat{\beta}^{[1]}_{S^c}\|_1,$$

and hence,

$$\|\hat{\beta}^{[1]}_{S^c}\|_1\leqslant 3\|\hat{\beta}^{[1]}_S-\beta_S\|_1.$$

Next we provide control on $\hat{u}^{[1]}$, measured in l_2-norm.

Lemma 2. *Assume ϵ_i are i.i.d. Gaussian random variables with mean 0 and variance σ^2. Suppose (C.1) holds and set $\lambda_0/n^{1/2}=4\sigma(\log p)^{1/2}$. Conditional on $\{2\|W\|_\infty\leqslant\lambda_0/n^{1/2}\}$, we have with probability exceeding $1-1/p$:*

$$\|\hat{u}^{[1]}\|_2\leqslant\frac{12\sigma}{K_2}(q\log p)^{1/2}.$$

Proof of Lemma. According to the Gaussian tail bound, we have for $t\geqslant\sigma$,

$$P(|\epsilon|>t)<\exp\left(-\frac{t^2}{2\sigma^2}\right),$$

and then we have the tail probability bound of W:

$$P\left(\|W\|_\infty>2\sigma(\log p)^{1/2}\right)<p\cdot\exp\left(-\frac{4\sigma^2\log p}{2\sigma^2}\right)=\frac{1}{p}.$$

Set

$$F(\beta)=\frac{1}{2}\|y-X\beta\|_2^2+\lambda_0\|\beta\|_1.$$

Define $V(u)=F(\hat{\beta}^{[1]})-F(\beta)$. We have

$$V(u)=\frac{1}{2}u^{\mathsf{T}}Cu-u^{\mathsf{T}}W+\lambda_0\sum_{j\in S}\left(\left|\beta_j+\frac{u_j}{n^{1/2}}\right|-|\beta_j|\right)+\lambda_0\sum_{j\in S^c}\left|\beta_j+\frac{u_j}{n^{1/2}}\right|. \tag{A3}$$

The solution $\hat{\beta}^{[1]}$ can, for each value of λ_0, be written as $\hat{\beta}^{[1]}=\beta+\hat{u}^{[1]}/n^{1/2}$, where

$$\hat{u}^{[1]}=\arg\min V(u).$$

Because the value of the function $V(u)$ is 0 when $u=0$, it follows that $V(\hat{u}^{[1]})\leqslant 0$. For the first term on the right hand of (A3), according to (C.1) and Lemma 1, the following inequality holds:

$$\frac{1}{2}(\hat{u}^{[1]})^{\mathsf{T}}C\hat{u}^{[1]}\geqslant\frac{K_2}{2}\|\hat{u}^{[1]}\|_2^2.$$

Then we have for u satisfying (C.1) that

$$V(u)\geqslant\frac{K_2}{2}\|u\|_2^2-u_S^{\mathsf{T}}W_S-\frac{\lambda_0}{n^{1/2}}\|u_S\|_1+\left[\frac{\lambda_0}{n^{1/2}}-\|W_{S^c}\|_\infty\right]\|u_{S^c}\|_1.$$

By $\lambda_0/n^{1/2}=4\sigma(\log p)^{1/2}$ and conditional on $\{2\|W\|_\infty\leqslant\lambda_0/n^{1/2}\}$, we have

$$\left\{2\|W_S\|_2\leqslant q^{1/2}\frac{\lambda_0}{n^{1/2}}\right\}.$$

Then it is straightforward to see that

$$\frac{K_2}{2}\|u\|_2^2 - u_S^{\mathsf{T}} W_S - \frac{\lambda_0}{n^{1/2}}\|u_S\|_1 \geqslant \|u_S\|_2 \left\{ \frac{K_2}{2}\|u\|_2 - \frac{3}{2}q^{1/2} \cdot \frac{\lambda_0}{n^{1/2}} \right\}$$

and

$$\left[\frac{\lambda_0}{n^{1/2}} - \|W_{S^c}\|_\infty\right] \sum_{j \in S^c} |u_j| > 0.$$

The two inequalities above imply that when

$$\|\hat{u}^{[1]}\|_2 > \frac{3\lambda_0 q^{1/2}}{K_2 n^{1/2}} = \frac{12\sigma}{K_2}(q \log p)^{1/2},$$

we have $V(\hat{u}) > 0$, which implies that the minimum of $V(u)$ is not attained. Hence,

$$\|\hat{u}^{[1]}\|_2 \leqslant \frac{12\sigma}{K_2}(q \log p)^{1/2}.$$

The following result shows that the initial step of the algorithm (i.e., Lasso) would not shrink the estimate of the nonzero coefficients to zero with high probability.

Proof of Proposition 1. According to Lemma 1 and Lemma 2, we have with probability $1 - 1/p$:

$$\|\hat{\beta}^{[1]} - \beta\|_2 \leqslant \frac{12\sigma}{K_2}(q \log p/n)^{1/2}.$$

The bound on the l_2-norm implies trivially the same bound on the l_∞-norm between $\hat{\beta}^{[1]}$ and β. When (2) holds, because $p = O(\exp(n^{c_1}))$ and $q = O(n^{c_2})$ where $c_1 + c_2 < 1$, it implies for $j \in S$, $\text{sign}(\hat{\beta}_j^{[1]}) = \text{sign}(\beta_j)$. Then we have

$$P(S \subseteq \mathcal{A}^{[1]}) \geqslant 1 - 1/p.$$

□

Proof of Theorem. We begin by stating the uniqueness of the Lasso solution. The sufficient condition for uniqueness has appeared many times in the literature, for example, Candès and Plan (2009), Tibshirani (2013), and Wainwright (2009). We summarize the result as the following: For any y, X, and $\lambda > 0$, if the predictor matrix X is drawn from a continuous probability distribution, then the Lasso solution is unique and the solution has at most $\min\{n, p\}$ nonzero components. Hence, after the first step of the algorithm, we have $|\mathcal{A}^{[1]}| \leqslant n$ and with probability at least $1 - 1/p$ that

$$(\hat{\beta}^{[2]} - \beta)X^{\mathsf{T}}\epsilon = (\hat{\beta}^{[2]}_{\mathcal{A}^{[1]}} - \beta_{\mathcal{A}^{[1]}})X^{\mathsf{T}}_{\mathcal{A}^{[1]}}\epsilon.$$

By $W_{\mathcal{A}^{[1]}} = X^{\mathsf{T}}_{\mathcal{A}^{[1]}}\epsilon/n^{1/2}$, we have

$$P(\|W_{\mathcal{A}^{[1]}}\|_\infty > 2\sigma(\log n)^{1/2}) < n \exp\left(-\frac{4\sigma^2 \log n}{2\sigma^2}\right) = 1/n.$$

By $\lambda/n^{1/2} = 4\sigma(\log n)^{1/2}$, with probability at least $1 - 1/n$, we have

$$2\|W_{\mathcal{A}^{[1]}}\|_\infty \leqslant \lambda/n^{1/2}.$$

We now consider the estimate of the second step of the algorithm:

$$\hat{\beta}^{[2]} := \arg\min_{\beta_{(\mathcal{A}^{[1]})^c}=0} \left\{ \frac{1}{2}\|y - X\beta\|_2^2 + \lambda \sum_{j \in \mathcal{A}^{[1]}} |\beta_j/\hat{\beta}_j^{[1]}| \right\}.$$

Set $\hat{u}^{[2]} = n^{1/2}(\hat{\beta}^{[2]} - \beta)$. We first prove $\hat{u}^{[2]} \in G(S)$ of (C.1). Following the same arguments as the proof of Lemma 1, the following inequalities hold:

$$\frac{1}{2}\|y - X\hat{\beta}^{[2]}\|_2^2 + \lambda \sum_{j \in \mathcal{A}^{[1]}} \left|\frac{\hat{\beta}_j^{[2]}}{\hat{\beta}_j^{[1]}}\right| \leqslant \frac{1}{2}\|y - X\beta\|_2^2 + \lambda \sum_{j \in \mathcal{A}^{[1]}} \left|\frac{\beta_j}{\hat{\beta}_j^{[1]}}\right|$$

and

$$\|X(\hat{\beta}^{[2]}-\beta)\|_2^2+2\lambda\sum_{j\in\mathcal{A}^{[1]}/S}\left|\frac{\hat{\beta}_j^{[2]}}{\hat{\beta}_j^{[1]}}\right|\leqslant 2\epsilon^{\mathrm{T}}X(\hat{\beta}^{[2]}-\beta)+2\lambda\sum_{j\in S}\left|\frac{\hat{\beta}_j^{[2]}-\beta_j}{\hat{\beta}_j^{[1]}}\right|. \tag{A4}$$

By Lemma 2, the following inequality holds with probability at least $1-1/p$:

$$\|\hat{\beta}^{[1]}-\beta\|_2\leqslant\frac{12\sigma}{K_2}\left(\frac{q\log p}{n}\right)^{1/2}$$

and when n is large, we have

$$\max_{j\notin S}|\hat{\beta}_j^{[1]}|<1.$$

Further, conditional on $\{\hat{\beta}^{[2]}_{(\mathcal{A}^{[1]})^c}=\beta_{(\mathcal{A}^{[1]})^c}=0\}$ and $\{2\|W_{\mathcal{A}^{[1]}}\|_\infty\leqslant\lambda/n^{1/2}\}$, we have

$$\begin{aligned}2\epsilon^{\mathrm{T}}X(\hat{\beta}^{[2]}-\beta)&\leqslant\lambda\|\hat{\beta}^{[2]}-\beta\|_1\\&=\lambda\|\hat{\beta}_S^{[2]}-\beta_S\|_1+\lambda\|\hat{\beta}^{[2]}_{\mathcal{A}^{[1]}/S}\|_1\\&<\frac{\lambda}{\max_{j\notin S}|\hat{\beta}_j^{[1]}|}\left(\|\hat{\beta}_S^{[2]}-\beta_S\|_1+\|\hat{\beta}^{[2]}_{\mathcal{A}^{[1]}/S}\|_1\right).\end{aligned} \tag{A5}$$

Combining (A4) and (A5) yields

$$\begin{aligned}\frac{\lambda}{\max_{j\notin S}|\hat{\beta}_j^{[1]}|}\|\hat{\beta}^{[2]}_{j\in\mathcal{A}^{[1]}/S}\|_1&\leqslant 2\lambda\sum_{j\in\mathcal{A}^{[1]}/S}\left|\frac{\hat{\beta}_j^{[2]}}{\hat{\beta}_j^{[1]}}\right|-\frac{\lambda}{\max_{j\notin S}|\hat{\beta}_j^{[1]}|}\|\hat{\beta}^{[2]}_{\mathcal{A}^{[1]}/S}\|_1\\&\leqslant\frac{\lambda}{\max_{j\notin S}|\hat{\beta}_j^{[1]}|}\|\hat{\beta}_S^{[2]}-\beta_S\|_1+2\frac{\lambda}{\min_{j\in S}|\hat{\beta}_j^{[1]}|}\|\hat{\beta}_S^{[2]}-\beta_S\|_2.\end{aligned}$$

According to (2) and the l_∞-norm bound on the difference between $\hat{\beta}^{[1]}$ and β, when n is large, we have

$$\frac{\lambda}{\min_{j\in S}|\hat{\beta}_j^{[1]}|}<\frac{\lambda}{\max_{j\notin S}|\hat{\beta}_j^{[1]}|}$$

and the following inequality holds:

$$\|\hat{\beta}^{[2]}_{S^c}\|_1\leqslant 3\|\hat{\beta}_S^{[2]}-\beta_S\|_1. \tag{A6}$$

By $\hat{u}^{[2]}=n^{1/2}(\hat{\beta}^{[2]}-\beta)$, (A6) implies $\|\hat{u}^{[2]}_{S^c}\|_1\leqslant 3\|\hat{u}^{[2]}_S\|_1$. According to (C.1), we have

$$(\hat{u}^{[2]})^{\mathrm{T}}C\hat{u}^{[2]}\geqslant K_2\|\hat{u}^{[2]}\|_2^2.$$

Using the similar notation of Lemma 2, the solution $\hat{\beta}^{[2]}$ can be written as $\hat{\beta}^{[2]}=\beta+\hat{u}^{[2]}/n^{1/2}$, where

$$\hat{u}^{[2]}=\arg\min V(u_{\mathcal{A}^{[1]}},0)$$

and with $u_{\mathcal{A}^{[1]}}$ satisfying (C.1), we have

$$\begin{aligned}V(u_{\mathcal{A}^{[1]}},0)&=\frac{1}{2}u_{\mathcal{A}^{[1]}}^{\mathrm{T}}Cu_{\mathcal{A}^{[1]}}-u_{\mathcal{A}^{[1]}}^{\mathrm{T}}W_{\mathcal{A}^{[1]}}+\lambda\sum_{j\in\mathcal{A}^{[1]}}\left(\left|\beta_j+\frac{\hat{u}_j}{n^{1/2}}\right|-|\beta_j|\right)/|\hat{\beta}_j^{[1]}|\\&\geqslant L_1+L_2,\end{aligned}$$

where

$$\begin{aligned}L_1&=\frac{K_2}{2}\|u_{\mathcal{A}^{[1]}}\|_2^2-u_S^{\mathrm{T}}W_S-\frac{\lambda}{n^{1/2}}\frac{\|u_S\|_1}{\min_{j\in S}|\hat{\beta}_j^{[1]}|},\\L_2&=\frac{\lambda}{n^{1/2}}\frac{\|u_{\mathcal{A}^{[1]}/S}\|_1}{\max_{j\notin S}|\hat{\beta}_j^{[1]}|}-u_{\mathcal{A}^{[1]}/S}^{\mathrm{T}}W_{\mathcal{A}^{[1]}/S}.\end{aligned}$$

Again, with probability $1 - 1/p$, we have

$$n^{1/2}\|\hat{\beta}^{[1]} - \beta\|_2 \leqslant \frac{12\sigma}{K_2}(q \log p)^{1/2}.$$

According to (2), there exists a constant $0 < K_3 < K_1$ such that, for $c_1 + c_2 \leqslant c_3$ and $j \in S$,

$$|\hat{\beta}_j^{[1]}| \geqslant K_3 n^{(c_3-1)/2}$$

and for $i \notin S$,

$$|\hat{\beta}_j^{[1]}| \leqslant \frac{12\sigma}{K_2}\left(\frac{q \log p}{n}\right)^{1/2}.$$

Hence, with $\lambda/n^{1/2} = 4\sigma(\log n)^{1/2}$ and $\{\|W_{\mathcal{A}^{[1]}}\|_\infty \leqslant 2\sigma(\log n)^{1/2}\}$, we have

$$\begin{aligned} L_1 &\geqslant \frac{K_2}{2}\|u_{\mathcal{A}^{[1]}}\|_2^2 - 2\sigma(q \log n)^{1/2}\|u_S\|_2 - \frac{4\sigma}{K_3}(q \cdot n^{1-c_3} \log n)^{1/2}\|u_S\|_2^2 \\ &\geqslant \|u_S\|_2^2 \left\{ \frac{K_2}{2}\|u_{\mathcal{A}^{[1]}}\|_2 - 2\sigma(q \log n)^{1/2} - \frac{4\sigma}{K_3}(q \cdot n^{1-c_3} \log n)^{1/2} \right\} \end{aligned}$$

and

$$L_2 \geqslant \left(\frac{K_2}{3} \cdot \left(\frac{n \log n}{q \log p} \right)^{1/2} - 2\sigma(\log n)^{1/2} \right) \cdot \|\hat{u}_{\mathcal{A}^{[1]}/S}\|_1 > 0.$$

Following the arguments as the proof of Lemma 2, when

$$\|\hat{u}^{[2]}_{\mathcal{A}^{[1]}}\|_2 > \frac{8\sigma}{K_2 \cdot K_3}(q \cdot n^{1-c_3} \log n)^{1/2},$$

it implies $V(u_{\mathcal{A}^{[1]}}, 0) > 0$, and we have $V(\hat{u}^{[2]}_{\mathcal{A}^{[1]}}, 0) \leqslant 0$. Hence, with probability at least $1 - 1/n$, we have

$$\|\hat{u}^{[2]}\|_2 \leqslant \frac{8\sigma}{K_2 \cdot K_3}(q \cdot n^{1-c_3} \log n)^{1/2}. \tag{A7}$$

According to (2), following the arguments as the proof of Proposition 1, we have

$$S \subseteq \mathcal{A}^{[2]}. \tag{A8}$$

Similarly, one can prove that (A7) and (A8) hold at the $(k+1)$th step when they hold at the kth step. Thus, by induction, they hold for $k = 2, 3, \ldots$. When the iteration converges, denote the set that $\mathcal{A}^{[k]}$ converges to as $\mathcal{A}$, and $\hat{\beta}$ is written as $\hat{\beta} = \beta + \hat{u}/n^{1/2}$ where

$$\hat{u} = \arg\min V(u_{\mathcal{A}}, 0)$$

and

$$\|\hat{u}\|_2 \leqslant M,$$

where $M = (8\sigma/K_2K_3)(q \cdot n^{1-c_3} \log n)^{1/2}$. Then according to $c_3 > (1 + 3c_2)/2$ and $c_2 < 1/3$, the following inequalities hold uniformly over $\{u \in \mathcal{R}^p : \|u\|_2 \leqslant M, u_{S^c} \neq 0\}$,

$$\begin{aligned} V(u) - V(u_S, 0) &\geqslant u_S^{\mathrm{T}} C_{SS^c} u_{S^c} + u_{S^c}^{\mathrm{T}} C_{S^cS^c} u_{S^c} - u_{S^c}^{\mathrm{T}} W_{S^c} + \lambda \sum_{j \in S^c} \frac{u_j}{M} \\ &\geqslant \sum_{j \in \mathcal{A}/S} |u_j| \left(\lambda/M - \|W_{\mathcal{A}}\|_\infty - q^{1/2}\|u_S\|_2 \right) \\ &\geqslant \sum_{j \in \mathcal{A}/S} |u_j| \left(\lambda/M - 2\sigma(\log n)^{1/2} - q^{1/2}M \right) \\ &> 0. \end{aligned} \tag{A9}$$

Because $V(0) = 0$, (A9) implies that the minimum of $V(u)$ cannot be attained at any u satisfying $u_{S^c} \neq 0$. Thus, we have

$$P(\mathrm{sign}(\hat{\beta}) = \mathrm{sign}(\beta)) \geqslant 1 - 1/n \to 1 \,\mathrm{as}\, n \to \infty$$

and with probability at least $1 - 1/n$,

$$\|\hat{\beta} - \beta\|_2 \leqslant \frac{8\sigma}{K_2 \cdot K_3}\left(\frac{q \log n}{n^{c_3}}\right)^{1/2}.$$

Note that $\|\hat{\beta} - \beta\|_1 \leqslant 4\|\hat{\beta}_S - \beta_S\|_1 \leqslant 4q^{1/2} \cdot \|\hat{\beta}_S - \beta_S\|_2$, then with probability $1 - 1/n$, we have

$$\|\hat{\beta} - \beta\|_1 \leqslant \frac{32\sigma \cdot q}{K_2 \cdot K_3}\left(\frac{\log n}{n^{c_3}}\right)^{1/2}.$$

□

Proof of Corollary 1. We omit the discussion about the first step estimation and directly consider the estimate of the second step. Following the arguments as the proof of Theorem 1, the solution $\hat{\beta}^{[2]}$ can be written as $\hat{\beta}^{[2]} = \beta + \hat{u}^{[2]}/n^{1/2}$, where

$$\hat{u}^{[2]} = \arg\min V(u_{\mathcal{A}^{[1]}}, 0)$$

and

$$V(u_{\mathcal{A}^{[1]}}, 0) \geqslant L_1 + L_2, \tag{A10}$$

when n is large enough and where

$$L_1 = \frac{K_2}{2}\|u_{\mathcal{A}^{[1]}}\|_2^2 - u_S^{\mathsf{T}} W_S - \frac{\lambda}{n^{1/2}}\frac{\|u_S\|_1}{c},$$
$$L_2 = \frac{\lambda}{n^{1/2}} \cdot \frac{\|u_{\mathcal{A}^{[1]}/S}\|_1}{\max_{i \notin S} |\hat{\beta}_i^{[1]}|} - u_{\mathcal{A}^{[1]}/S}^{\mathsf{T}} W_{\mathcal{A}^{[1]}/S}.$$

This is because c is a positive constant and we have

$$\|\hat{\beta}^{[1]} - \beta\|_2 \leqslant \frac{12\sigma}{K_2}\left(\frac{q \log p}{n}\right)^{1/2} \to 0 \text{ as } n \to \infty.$$

Set $\lambda = 4\sigma(n \log n)^{1/2}$. We have $L_2 > 0$ and

$$L_1 \geqslant \|u_S\|_2 \left\{\frac{K_2}{2}\|u_{\mathcal{A}^{[1]}}\|_2 - 2\sigma(q \log n)^{1/2} - \frac{4\sigma}{c}(q \log n)^{1/2}\right\}.$$

For $V(u_{\mathcal{A}^{[1]}}^{[2]}, 0) \leqslant 0$, we have with probability $1 - 1/n$,

$$\|\hat{u}^{[2]}\|_2 \leqslant \frac{8\sigma}{cK_2}(q \log n)^{1/2}.$$

Hence

$$\|\hat{\beta}^{[2]} - \beta\|_2 \leqslant \frac{8\sigma}{cK_2}\left(\frac{q \log n}{n}\right)^{1/2} \tag{A11}$$

and

$$S \subseteq \mathcal{A}^{[2]}. \tag{A12}$$

By induction, (A11) and (A12) hold for $k = 2, 3, \ldots$. When the iteration converges, $\hat{\beta}$ satisfies

$$\|\hat{\beta} - \beta\|_2 \leqslant \frac{8\sigma}{cK_2}\left(\frac{q \log n}{n}\right)^{1/2}$$

and

$$\|\hat{\beta} - \beta\|_1 \leqslant \frac{32\sigma \cdot q}{cK_2}\left(\frac{\log n}{n}\right)^{1/2}.$$

Write $\hat{\beta}$ as $\hat{\beta} = \beta + \hat{u}/n^{1/2}$ where

$$\hat{u} = \arg\min V(u_{\mathcal{A}}, 0).$$

Set $M = (8\sigma/cK_2)(q \log n)^{1/2}$. According to $0 < c_2 < 1/3$, the following inequalities hold uniformly over $\{u \in \mathcal{R}^p : \|u\|_2 \leqslant M, u_{S^c} \neq 0\}$,

$$\begin{aligned} V(u) - V(u_S, 0) &\geqslant u_S^{\mathsf{T}} C_{SS^c} u_{S^c} + u_{S^c}^{\mathsf{T}} C_{S^cS^c} u_{S^c} - u_{S^c}^{\mathsf{T}} W_{S^c} + \lambda \sum_{j \in S^c} \frac{u_j}{M} \\ &\geqslant \sum_{j \in \mathcal{A}/S} |u_j| \left[\lambda/M - 2\sigma(\log n)^{1/2} - q^{1/2}M\right] \\ &> 0. \end{aligned}$$

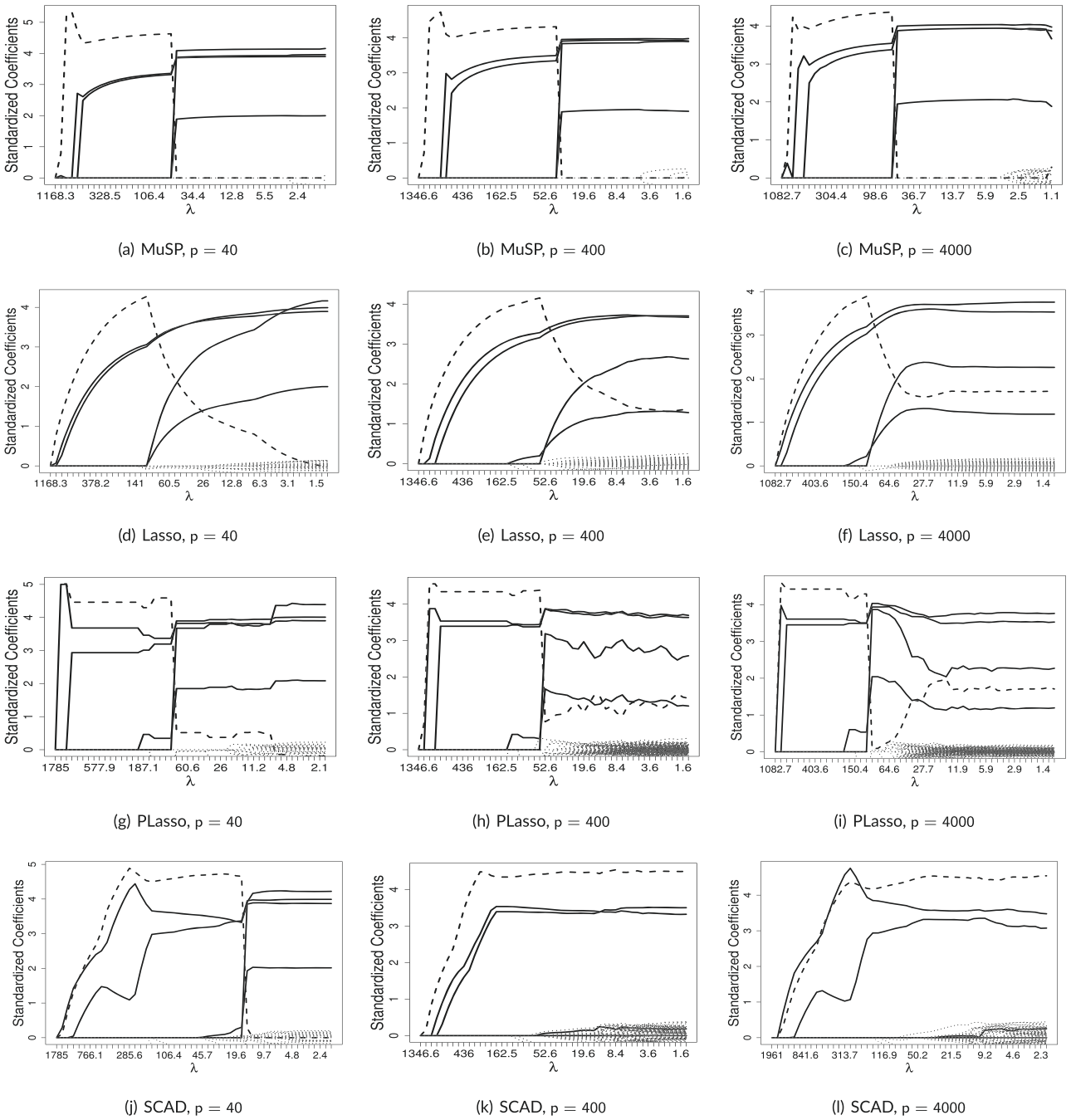
It follows that the minimum of $V(u)$ cannot be attained at any u satisfying $u_{S^c} \neq 0$. Then we have

$$P(\text{sign}(\hat{\beta}) = \text{sign}(\beta)) \geqslant 1 - 1/n \to 1 \text{as} n \to \infty.$$

□

APPENDIX B: ADDITIONAL SIMULATION RESULTS

Figure B1 shows the comparison of the proposed MuSP with Lasso, PLasso, and SCAD under Scenario 1, and Figure B2 shows the comparison of the proposed MuSP with Capped-l_1, LLA, MCP, and SSL under Scenario 2.



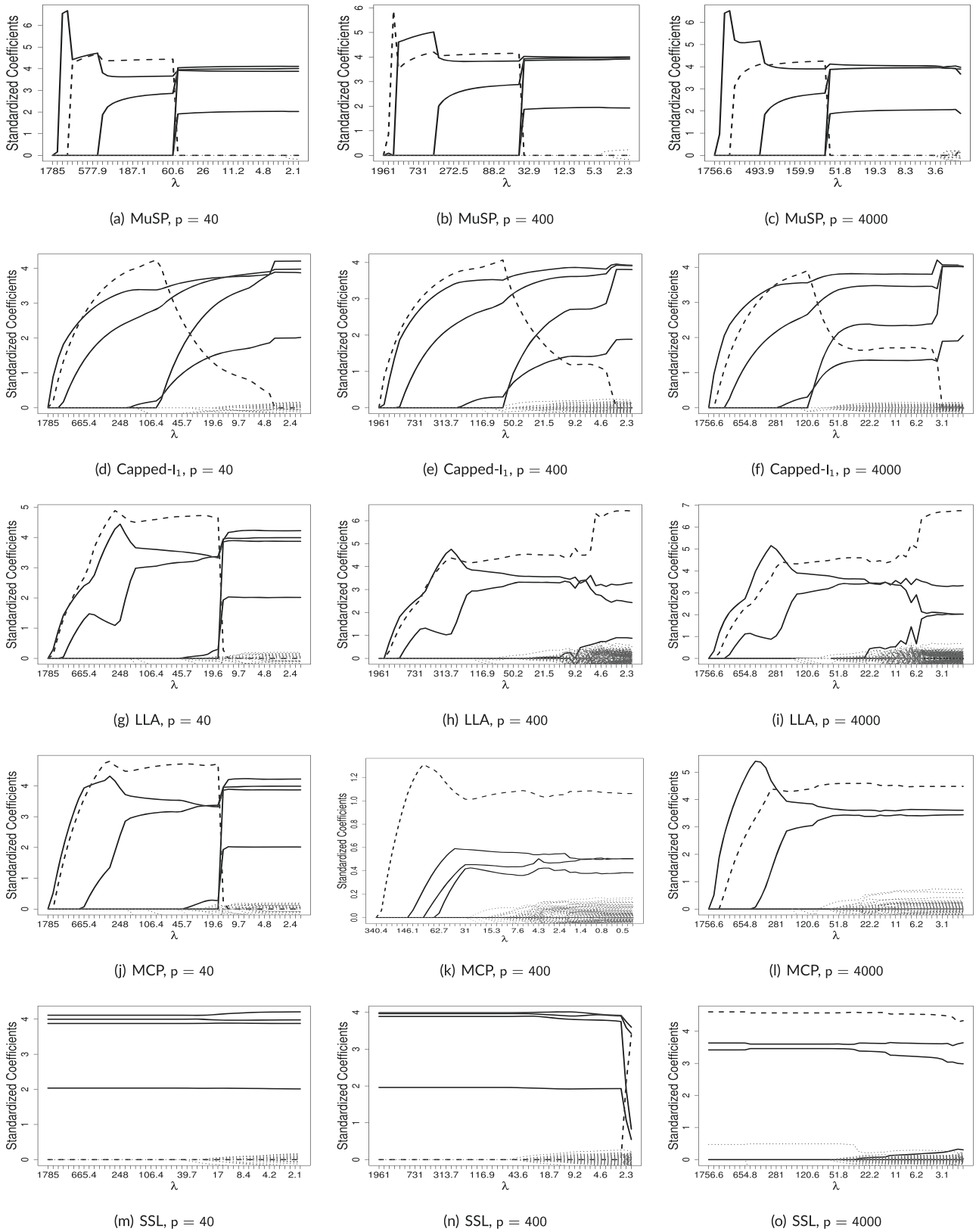
(a) MuSP, p = 40 (b) MuSP, p = 400 (c) MuSP, p = 4000

(d) Lasso, p = 40 (e) Lasso, p = 400 (f) Lasso, p = 4000

(g) PLasso, p = 40 (h) PLasso, p = 400 (i) PLasso, p = 4000

(j) SCAD, p = 40 (k) SCAD, p = 400 (l) SCAD, p = 4000

FIGURE B1 Results under Scenario 1 for three different dimensions. The dashed line corresponds to X_1, which is irrelevant; the dotted lines correspond to other irrelevant variables; the solid lines correspond to the relevant variables



(a) MuSP, p = 40 (b) MuSP, p = 400 (c) MuSP, p = 4000

(d) Capped-l_1, p = 40 (e) Capped-l_1, p = 400 (f) Capped-l_1, p = 4000

(g) LLA, p = 40 (h) LLA, p = 400 (i) LLA, p = 4000

(j) MCP, p = 40 (k) MCP, p = 400 (l) MCP, p = 4000

(m) SSL, p = 40 (n) SSL, p = 400 (o) SSL, p = 4000

FIGURE B2 Results under Scenario 2 for three different dimensions. The dashed line corresponds to X_1, which is irrelevant; the dotted lines correspond to other irrelevant variables; the solid lines correspond to the relevant variables