A polynomial algorithm for best-subset selection problem

Junxian Zhu[a,1], Canhong Wen[b,1], Jin Zhu[a], Heping Zhang[c,2], and Xueqin Wang[b,2]

[a]School of Mathematics, Sun Yat-sen University, Guangzhou, Guangdong 510275, China; [b]School of Management, University of Science and Technology of China, Hefei, Anhui 230026, China; and [c]Department of Biostatistics, Yale University School of Public Health, New Haven, CT 06525

Edited by Runze Li, Pennsylvania State University, State College, PA, and accepted by Editorial Board Member David A. Weitz November 18, 2020 (received for review July 7, 2020)

Best-subset selection aims to find a small subset of predictors, so that the resulting linear model is expected to have the most desirable prediction accuracy. It is not only important and imperative in regression analysis but also has far-reaching applications in every facet of research, including computer science and medicine. We introduce a polynomial algorithm, which, under mild conditions, solves the problem. This algorithm exploits the idea of sequencing and splicing to reach a stable solution in finite steps when the sparsity level of the model is fixed but unknown. We define an information criterion that helps the algorithm select the true sparsity level with a high probability. We show that when the algorithm produces a stable optimal solution, that solution is the oracle estimator of the true parameters with probability one. We also demonstrate the power of the algorithm in several numerical studies.

best-subset selection | splicing | high dimensional

Subset selection is a classic topic of model selection in statistical learning and is encountered whenever we are interested in understanding the relationship between a response and a set of explanatory variables. Naturally, this problem has been pursued in statistics and mathematics for decades. The classic methods that are commonly described in statistical textbooks include stepwise regression with the Akaike information criterion (1), the Bayesian information criterion (BIC) (2), and Mallows's C_p (3).

Consider n independent observations $(\boldsymbol{x}_i, y_i), i = 1, \ldots, n$, where $\boldsymbol{x}_i \in \mathbb{R}^{1\times p}$, $y_i \in \mathbb{R}$. Let $\boldsymbol{y} = (y_1, \ldots, y_n)$ and $\boldsymbol{X} = (\boldsymbol{x}_1^\top, \ldots, \boldsymbol{x}_n^\top)^\top$. For convenience, we centralize the columns of $\boldsymbol{X}$ to have zero mean. The following is the classic multivariable linear model with regression coefficient vector $\boldsymbol{\beta} \in \mathbb{R}^{p\times 1}$ and error vector $\boldsymbol{\epsilon} \in \mathbb{R}^{n\times 1}$:

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}. \quad [1]$$

Parsimony is desired when we consider a subset of the p explanatory variables in Model 1 with comparable prediction accuracy. When the regression coefficient vector $\boldsymbol{\beta}$ is sparse, we want to identify this subset of nonzero coefficients. This is the commonly known problem of the best-subset selection that minimizes the empirical risk function, e.g., the sum of residual squares, under the cardinality constraint in Model 1,

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \frac{1}{2n}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2, \text{ subject to } \|\boldsymbol{\beta}\|_0 \le s, \quad [2]$$

where $\|\boldsymbol{\beta}\|_0 = \sum_{i=1}^p I(\beta_i \neq 0)$ is the ℓ_0 norm of $\boldsymbol{\beta}$, and the sparsity level s is usually an unknown nonnegative integer.

The Lagrangian of Eq. **2** represents a balance between goodness of fit and parsimony. The latter is characterized by model complexity that is generally defined as an increasing function of the number of nonzero $\boldsymbol{\beta}$ values. Thus, this Lagrangian is not continuous and, of course, not smooth. Greedy methods are usually applied to solve such Lagrangian but suffer from computational difficulties even for a reasonably large p. Alternatively, some relaxation methods, e.g., Least-Absolute Shrinkage and Selection Operator (LASSO) (4), Adaptive LASSO (5), Smoothly Clipped Absolute Deviation Penalty (SCAD) (6), and Minimax Concave Penalty (MCP) (7) have been proposed and investigated to ameliorate the computational issue by replacing the nonsmooth penalty function with a smooth approximation. These recently developed methods are computationally feasible and provide near-optimal solutions even for large p. However, their solutions do not lead to the best subset and are known for lack of important statistical properties (8).

There has been little progress on how to find the best-subset selection until recently because such a nonsmooth optimization problem is generally nondeterministic polynomial-time–hard (9). Recently, to make the best-subset selection problem computationally tractable, optimization strategies and algorithms are proposed, including the Iterate Hard Thresholding (IHT) algorithm (10), primal-dual active set (PDAS) methods (11), and the Mixed Integer Optimization (MIO) approach (12). However, their solutions may converge to a local minimizer, and IHT and PDAS may also suffer from the periodic iterative issue. More importantly, these methods do not determine the sparsity-level adaptively, and their statistical properties remain unclear.

In this paper, we directly deal with Eq. **2** and solve the best-subset selection problem with two critical ideas: a splicing algorithm and an information criterion. Our contribution is threefold. Firstly, we propose "splicing," a technique to improve the quality of subset selection, and derive an efficient iterative algorithm based on splicing, Adaptive Best-Subset Selection (ABESS), to tackle problem 2. The ABESS algorithm is applicable to analyze high dimensional datasets with tens of thousands of observations and variables. Secondly, we prove that ABESS algorithm consistently selects important variables and its

Significance

Best-subset selection is a benchmark optimization problem in statistics and machine learning. Although many optimization strategies and algorithms have been proposed to solve this problem, our splicing algorithm, under reasonable conditions, enjoys the following properties simultaneously with high probability: 1) its computational complexity is polynomial; 2) it can recover the true subset; and 3) its solution is globally optimal.

Author contributions: H.Z. and X.W. designed research; Junxian Zhu, C.W., Jin Zhu, H.Z., and X.W. performed research; Junxian Zhu, C.W., Jin Zhu, and X.W. contributed new reagents/analytic tools; Junxian Zhu, C.W., Jin Zhu, and X.W. analyzed data; and Junxian Zhu, C.W., Jin Zhu, H.Z., and X.W. wrote the paper.

The authors declare no competing interest.

This article is a PNAS Direct Submission.

R.L. is a guest editor invited by the Editorial Board.

[1]Junxian Zhu and C.W. contributed equally to this work.

[2]To whom correspondence may be addressed. Email: heping.zhang@yale.edu or wangxq20@ustc.edu.cn.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2014241117/-/DCSupplemental.

First published December 16, 2020.

computational complexity is polynomial. Our algorithm is stringently shown to solve problem 2 within polynomial times. Finally, to determine the most suitable sparsity level, we design an information criterion (special information criterion [SIC]) whose theoretical best-subset selection consistency is rigorously proven.

We define some useful notations for the content below. For $\boldsymbol{\beta}=(\beta_1,\ldots,\beta_p)^\top\in\mathbb{R}^p$, we define the ℓ_q norm of $\boldsymbol{\beta}$ by $\|\boldsymbol{\beta}\|_q=(\sum_{j=1}^p|\beta_j|^q)^{1/q}$, where $q\in[1,\infty)$. Let $\mathcal{S}=\{1,\ldots,p\}$, for any set $\mathcal{A}\subseteq\mathcal{S}$, denote $\mathcal{A}^c=\mathcal{S}\backslash\mathcal{A}$ as the complement of $\mathcal{A}$ and $|\mathcal{A}|$ as its cardinality. We define the support set of vector $\boldsymbol{\beta}$ as $\mathrm{supp}(\boldsymbol{\beta})=\{j:\beta_j\neq0\}$. For an index set $\mathcal{A}\subseteq\{1,\ldots,p\}$, $\boldsymbol{\beta}_{\mathcal{A}}=(\beta_j,j\in\mathcal{A})\in\mathbb{R}^{|\mathcal{A}|}$. For matrix $\boldsymbol{X}\in\mathbb{R}^{n\times p}$, define $\boldsymbol{X}_{\mathcal{A}}=(X_j,j\in\mathcal{A})\in\mathbb{R}^{n\times|\mathcal{A}|}$. For any vector $\boldsymbol{t}$ and any set $\mathcal{A}$, $\boldsymbol{t}^{\mathcal{A}}$ is defined to be the vector whose jth entry $(\boldsymbol{t}^{\mathcal{A}})_j$ is equal to t_j if $j\in\mathcal{A}$ and zero otherwise. For instance, $\hat{\boldsymbol{\beta}}^{\mathcal{A}}$ is the vector whose jth entry is $\hat{\beta}_j$ if $j\in\mathcal{A}$ and zero otherwise. $\hat{\boldsymbol{t}}^{\{j\}}$ is the vector whose jth entry is $\hat{t}_j$ and zero otherwise.

Method

Splicing. In this section, we describe the splicing method. Consider the ℓ_0 constraint minimization problem,

$$\min_{\boldsymbol{\beta}}\mathcal{L}_n(\boldsymbol{\beta}),\ \ s.t\ \|\boldsymbol{\beta}\|_0\le s,$$

where $\mathcal{L}_n(\boldsymbol{\beta})=\frac{1}{2n}\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}\|_2^2$. Without loss of generality, we consider $\|\boldsymbol{\beta}\|_0=s$. Given any initial set $\mathcal{A}\subset\mathcal{S}=\{1,2,\ldots,p\}$ with cardinality $|\mathcal{A}|=s$, denote $\mathcal{I}=\mathcal{A}^c$ and compute

$$\hat{\boldsymbol{\beta}}=\arg\min_{\boldsymbol{\beta}_{\mathcal{I}}=0}\mathcal{L}_n(\boldsymbol{\beta}).$$

We call $\mathcal{A}$ and $\mathcal{I}$ as the active set and the inactive set, respectively.

Given the active set $\mathcal{A}$ and $\hat{\boldsymbol{\beta}}$, we can define the following two types of sacrifices:

1) Backward sacrifice: For any $j\in\mathcal{A}$, the magnitude of discarding variable j is,

$$\xi_j=\mathcal{L}_n\left(\hat{\boldsymbol{\beta}}^{\mathcal{A}\backslash\{j\}}\right)-\mathcal{L}_n(\hat{\boldsymbol{\beta}}^{\mathcal{A}})=\frac{\boldsymbol{X}_j^\top\boldsymbol{X}_j}{2n}(\hat{\beta}_j)^2. \quad [3]$$

2) Forward sacrifice: For any $j\in\mathcal{I}$, the magnitude of adding variable j is,

$$\zeta_j=\mathcal{L}_n(\hat{\boldsymbol{\beta}^{\mathcal{A}}})-\mathcal{L}_n\left(\hat{\boldsymbol{\beta}}^{\mathcal{A}}+\hat{\boldsymbol{t}}^{\{j\}}\right)=\frac{\boldsymbol{X}_j^\top\boldsymbol{X}_j}{2n}\left(\frac{\hat{d}_j}{\boldsymbol{X}_j^\top\boldsymbol{X}_j/n}\right)^2, \quad [4]$$

where $\hat{\boldsymbol{t}}=\arg\min_{\boldsymbol{t}}\mathcal{L}_n\left(\hat{\boldsymbol{\beta}}^{\mathcal{A}}+\boldsymbol{t}^{\{j\}}\right)$, $\hat{d}_j=\boldsymbol{X}_j^\top(\boldsymbol{y}-\boldsymbol{X}\hat{\boldsymbol{\beta}})/n$.

Intuitively, for $j\in\mathcal{A}$ (or $j\in\mathcal{I}$), a large ξ_j (or ζ_j) implies the jth variable is potentially important. Unfortunately, it is noteworthy that these two sacrifices are incomparable because they have different sizes of support set. However, if we exchange some "irrelevant" variables in $\mathcal{A}$ and some "important" variables in $\mathcal{I}$, it may result in a higher-quality solution. This intuition motivates our splicing method.

Specifically, given any splicing size $k\le s$, define

$$\mathcal{A}_k=\left\{j\in\mathcal{A}:\sum_{i\in\mathcal{A}}\mathsf{I}(\xi_j\ge\xi_i)\le k\right\}$$

to represent k least relevant variables in $\mathcal{A}$ and

$$\mathcal{I}_k=\left\{j\in\mathcal{I}:\sum_{i\in\mathcal{I}}\mathsf{I}(\zeta_j\le\zeta_i)\le k\right\}$$

to represent k most relevant variables in $\mathcal{I}$. Then, we splice $\mathcal{A}$ and $\mathcal{I}$ by exchanging $\mathcal{A}_k$ and $\mathcal{I}_k$ and obtain a new active set

$$\tilde{\mathcal{A}}=(\mathcal{A}\setminus\mathcal{A}_k)\cup\mathcal{I}_k.$$

Let $\tilde{\mathcal{I}}=\tilde{\mathcal{A}}^c$, $\tilde{\boldsymbol{\beta}}=\arg\min_{\boldsymbol{\beta}_{\tilde{\mathcal{I}}}=0}\mathcal{L}_n(\boldsymbol{\beta})$, and $\tau_s>0$ be a threshold. If $\tau_s<\mathcal{L}_n(\tilde{\boldsymbol{\beta}})-\mathcal{L}_n(\hat{\boldsymbol{\beta}})$, then $\tilde{\mathcal{A}}$ is preferable to $\mathcal{A}$. The active set can be updated iteratively until the loss function cannot be improved by splicing. Once the algorithm recovers the true active set, we may splice some irrelevant variables, and then the loss function may decrease slightly. The threshold τ_s can reduce this unnecessary calculation. Typically, τ_s is relatively small, e.g., $\tau_s=0.01s\log(p)\log(\log n)/n$.

The remaining problem is to determine the initial set. Typically, we select the first s variables that are most correlated with $\boldsymbol{y}$ variables as the initial set $\mathcal{A}$. Let k_{max} be the maximum splicing size, $k_{max}\le s$. In the following, we summarize our arguments in the above:

Algorithm 1: BESS.Fix(s): Best-Subset Selection with a given support size s.

1) Input: $\boldsymbol{X}$, $\boldsymbol{y}$, a positive integer k_{max}, and a threshold τ_s.
2) Initialize $\mathcal{A}^0=\{j:\sum_{i=1}^p\mathsf{I}(|\frac{\boldsymbol{x}_j^\top\boldsymbol{y}}{\sqrt{\boldsymbol{x}_j^\top\boldsymbol{x}_j}}|\le|\frac{\boldsymbol{x}_i^\top\boldsymbol{y}}{\sqrt{\boldsymbol{x}_i^\top\boldsymbol{x}_i}}\le s\}$, $\mathcal{I}^0=(\mathcal{A}^0)^c$,
and $(\boldsymbol{\beta}^0,\boldsymbol{d}^0)$:
$\boldsymbol{\beta}^0_{\mathcal{I}^0}=0$,
$\boldsymbol{d}^0_{\mathcal{A}^0}=0$,
$\boldsymbol{\beta}^0_{\mathcal{A}^0}=(\boldsymbol{X}_{\mathcal{A}^0}^\top\boldsymbol{X}_{\mathcal{A}^0})^{-1}\boldsymbol{X}_{\mathcal{A}^0}^\top\boldsymbol{y}$,
$\boldsymbol{d}^0_{\mathcal{I}^0}=\boldsymbol{X}_{\mathcal{I}^0}^\top\left(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}^0\right)/n$.
3) **For** $m=0,1,\ldots,$ **do**
$(\boldsymbol{\beta}^{m+1},\boldsymbol{d}^{m+1},\mathcal{A}^{m+1},\mathcal{I}^{m+1})$ = Splicing($\boldsymbol{\beta}^m,\boldsymbol{d}^m,\mathcal{A}^m,\mathcal{I}^m,k_{max},\tau_s$).
If $(\mathcal{A}^{m+1},\mathcal{I}^{m+1})=(\mathcal{A}^m,\mathcal{I}^m)$, **then** stop
end for
4) Output $(\hat{\boldsymbol{\beta}},\hat{\boldsymbol{d}},\hat{\mathcal{A}},\hat{\mathcal{I}})=(\boldsymbol{\beta}^{m+1},\boldsymbol{d}^{m+1}\mathcal{A}^{m+1},\mathcal{I}^{m+1})$.

Note that splicing size k is an important parameter in splicing. Typically, we can try all possible values of $k\le s$.

Algorithm 2: Splicing ($\boldsymbol{\beta}$, $\boldsymbol{d}$, $\mathcal{A}$, $\mathcal{I}$, k_{max}, τ_s).

1) Input: $\boldsymbol{\beta}$, $\boldsymbol{d}$, $\mathcal{A}$, $\mathcal{I}$, k_{max}, and τ_s.
2) Initialize $L_0=L=\frac{1}{2n}\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}\|_2^2$, and set
$\xi_j=\frac{\boldsymbol{x}_j^\top\boldsymbol{x}_j}{2n}(\beta_j)^2$, $\zeta_j=\frac{\boldsymbol{x}_j^\top\boldsymbol{x}_j}{2n}(\frac{d_j}{\boldsymbol{x}_j^\top\boldsymbol{x}_j/n})^2$, $j=1,\ldots,p$.
3) **For** $k=1,2,\ldots,k_{max}$, **do**
$\mathcal{A}_k=\{j\in\mathcal{A}:\sum_{i\in\mathcal{A}}\mathsf{I}(\xi_j\ge\xi_i)\le k\}$,
$\mathcal{I}_k=\{j\in\mathcal{I}:\sum_{i\in\mathcal{I}}\mathsf{I}(\zeta_j\le\zeta_i)\le k\}$.
Let $\tilde{\mathcal{A}}_k=(\mathcal{A}\backslash\mathcal{A}_k)\cup\mathcal{I}_k$, $\tilde{\mathcal{I}}_k=(\mathcal{I}\backslash\mathcal{I}_k)\cup\mathcal{A}_k$ and solve
$\tilde{\boldsymbol{\beta}}_{\tilde{\mathcal{A}}_k}=\left(\boldsymbol{X}_{\tilde{\mathcal{A}}_k}^\top\boldsymbol{X}_{\tilde{\mathcal{A}}_k}\right)^{-1}\boldsymbol{X}_{\tilde{\mathcal{A}}_k}^\top\boldsymbol{y}$, $\tilde{\boldsymbol{\beta}}_{\tilde{\mathcal{I}}_k}=0$,
$\tilde{\boldsymbol{d}}=\boldsymbol{X}^\top(\boldsymbol{y}-\boldsymbol{X}\tilde{\boldsymbol{\beta}})/n$, $\mathcal{L}_n(\tilde{\boldsymbol{\beta}})=\frac{1}{2n}\|\boldsymbol{y}-\boldsymbol{X}\tilde{\boldsymbol{\beta}}\|_2^2$.
If $L>\mathcal{L}_n(\tilde{\boldsymbol{\beta}})$, **then**
$(\hat{\boldsymbol{\beta}},\hat{\boldsymbol{d}},\hat{\mathcal{A}},\hat{\mathcal{I}})=(\tilde{\boldsymbol{\beta}},\tilde{\boldsymbol{d}},\tilde{\mathcal{A}}_k,\tilde{\mathcal{I}}_k)$,
$L=\mathcal{L}_n(\tilde{\boldsymbol{\beta}})$.
End for
4) **If** $L_0-L<\tau_s$, then $(\hat{\boldsymbol{\beta}},\hat{\boldsymbol{d}},\hat{\mathcal{A}},\hat{\mathcal{I}})=(\boldsymbol{\beta},\boldsymbol{d},\mathcal{A},\mathcal{I})$.
5) Output $(\hat{\boldsymbol{\beta}},\hat{\boldsymbol{d}},\hat{\mathcal{A}},\hat{\mathcal{I}})$.

ABESS. In practice, the support size is usually unknown. We use a data-driven procedure to determine s. Information criteria such as high-dimensional BIC (HBIC) (13) and extended BIC (EBIC) (14) are commonly used for this purpose. Specifically, HBIC (13) can be applied to select the tuning parameter in penalized likelihood estimation. To recover the support size s for the best-subset selection, we introduce a criterion that is a special case of HBIC (13). While HBIC aims to tune the parameter for a nonconvex penalized regression, our proposal is used to determine the size of best subset. For any active set $\mathcal{A}$, define an SIC as follows:

$$\mathrm{SIC}(\mathcal{A})=n\log\mathcal{L}_{\mathcal{A}}+|\mathcal{A}|\log(p)\log\log n,$$

where $\mathcal{L}_{\mathcal{A}}=\min_{\boldsymbol{\beta}_{\mathcal{I}}=0}\mathcal{L}_n(\boldsymbol{\beta})$, $\mathcal{I}=(\mathcal{A})^c$. To identify the true model, the model complexity penalty is $\log p$ and the slow diverging rate $\log\log n$ is set to prevent underfitting. *Theorem 4* states that the following **ABESS** algorithm selects the true support size via SIC.

Let s_{max} be the maximum support size. *Theorem 4* suggests $s_{max}=o(\frac{n}{\log p})$ as the maximum possible recovery size. Typically, we set $s_{max}=[\frac{n}{\log p\log\log n}]$, where $[x]$ denotes the integer part of x.

Algorithm 3: ABESS.

1) Input: $\boldsymbol{X}$, $\boldsymbol{y}$, and the maximum support size $s_{\max}$.
2) **For** $s = 1, 2, \ldots, s_{max}$, **do**
   $(\hat{\boldsymbol{\beta}}_s, \hat{\boldsymbol{d}}_s, \hat{\mathcal{A}}_s, \hat{\mathcal{I}}_s) =$ **BESS.Fixed**(s).
   End for
3) Compute the minimum of SIC:
   $s_{min} = \arg\min_s \text{SIC}(\hat{\mathcal{A}}_s)$.
4) Output $(\hat{\boldsymbol{\beta}}_{s_{min}}, \hat{\boldsymbol{d}}_{s_{min}}, \hat{\mathcal{A}}_{s_{min}}, \hat{\mathcal{I}}_{s_{min}})$.

Theoretical Results

We establish the computational complexity and the consistency of the best subset recovery from the ABESS algorithm.

Conditions. Let $\boldsymbol{\beta}^*$ be the true regression coefficient with the sparsity level s^* in Model 1. Denote the true active set by $\mathcal{A}^* = \text{supp}(\boldsymbol{\beta}^*)$ and the minimal signal strength by $b^* = \min_{j\in\mathcal{A}^*}(\beta_j^*)^2$. Without loss of generality, assume the design matrix $\boldsymbol{X}$ has $\sqrt{n}$-normalized columns, i.e., $\boldsymbol{X}_j^\top \boldsymbol{X}_j = n, j = 1, 2, \ldots, p$. We say that X satisfies the Sparse Restricted Condition (SRC) (15) with order s and spectrum bound $0 < c_-(s) < c_+(s) < \infty$ if $\forall \mathcal{A} \subset \mathcal{S}$ with $|\mathcal{A}| \le s$ and $\forall \boldsymbol{u} \neq 0, \boldsymbol{u} \in \mathbb{R}^{|\mathcal{A}|}$,

$$c_-(s) \le \frac{\|\boldsymbol{X}_{\mathcal{A}}\boldsymbol{u}\|_2^2}{n\|\boldsymbol{u}\|_2^2} \le c_+(s).$$

We denote this condition by $\boldsymbol{X} \sim \text{SRC}\{s, c_-(s), c_+(s)\}$. The SRC gives the range of the spectrum of the diagonal submatrices of the Gram matrix $G = \boldsymbol{X}^\top \boldsymbol{X}/n$. The spectrum of the off-diagonal submatrices of G can be bounded by the sparse orthogonality constant $\theta_{a,b}$, defined as the smallest number such that

$$\theta_{a,b} \ge \frac{\|\boldsymbol{X}_{\mathcal{A}}^\top \boldsymbol{X}_{\mathcal{B}}\boldsymbol{u}\|_2}{n\|\boldsymbol{u}\|_2},$$

for $\forall \mathcal{A}, \mathcal{B} \subset \mathcal{S}, |\mathcal{A}| \le a, |\mathcal{B}| \le b$ and $\mathcal{A} \cap \mathcal{B} = \varnothing$ and $\forall \boldsymbol{u} \neq 0, \boldsymbol{u} \in \mathbb{R}^{|\mathcal{B}|}$. For any $0 < \Delta < \frac{1}{2}$, denote

$$\delta_s \doteq \frac{8c_+(s)\left((1+\Delta)\frac{\theta_{s,s}}{c_-(s)}\left(1+\frac{\theta_{s,s}}{c_-(s)}\right)\right)^2}{(1-\Delta)\left(c_-(s) - \frac{\theta_{s,s}^2}{c_-(s)}\right)}. \quad \textbf{[5]}$$

To prove the theoretical properties of the ℓ_0 estimator, we assume the following conditions:

1) The random errors $\epsilon_1, \ldots, \epsilon_n$ are *i.i.d* with mean zero and sub-Gaussian tails; that is, there exists a $\sigma > 0$ such that $P\{|\epsilon_i| \ge t\} \le 2\exp(-t^2/\sigma^2)$, for all $t \ge 0$.
2) $\boldsymbol{X} \sim \text{SRC}\{2s, c_-(2s), c_+(2s)\}$.
3) $\delta_s < 1$, where δ_s is defined in Eq. **5**.
4) $\tau_s = O(\frac{s\log p \log\log n}{n})$.
5) $\frac{s^*\log p}{n} = o(1)$.
6) $\frac{1}{b^*} = o(\frac{n}{s\log p\log\log n})$.
7) $\frac{s^*\log p\log(\log n)}{n} = o(1)$ and $\frac{s_{max}\log p}{n} = o(1)$.

Remark 1: The sub-Gaussian condition is often assumed in the related literature and slightly weaker than the standard normality assumption. Condition 2 imposes bounds on the $2s$-sparse eigenvalues of the design matrix. As a typical condition in modeling involving high-dimensional data, it restricts the correlation among a small number of variables and thus guarantees the identifiability of the true active set. For example, the SRC has been assumed in existing methods (15–17). Sufficient conditions are provided for a design matrix to satisfy the SRC in propositions 4.1 and 4.2 in ref. 15.

Remark 2: To verify condition 3, let $c(s) = (1 - c_-(2s)) \vee (c_+(2s) - 1)$, which is closely related to the restricted isometry property (RIP) (18) constant δ_{2s} for $\boldsymbol{X}$. By lemma 20 in ref. 19, a sufficient condition for condition 3 is $c(s) \le 0.1877$, i.e., $c_-(2s) \ge 0.8123$, $c_+(2s) \le 1.1877$, which is weaker than the condition $c(s) \le 0.1599$ in ref. 19.
Remark 3: Condition 4 ensures that the threshold τ_s can control random errors. Condition 6 is the minimal magnitude of the signal for the best subset recovery. To discriminate between the signal and threshold, the signal needs to be stronger than the threshold. The condition is slightly stronger than the condition in ref. 20.
Remark 4: For the recovery of the true active set, the true sparsity level s^* and the maximum model size $s_{\max}$ cannot be too large. Condition 7 is weaker than the condition in ref. 13 as we consider the least-squares loss function without concave penalty. As shown in the *SI Appendix*, condition 5 can be removed.

Computational Theory. Firstly, we show that the splicing method converges in finite steps.
Theorem 1. *Algorithm* 1 *terminates in a finite number of iterations.*
This follows immediately from the fact that $\mathcal{L}_n(\boldsymbol{\beta}^{m+1}) < \mathcal{L}_n(\boldsymbol{\beta}^m)$. Furthermore, the next theorem delineates the polynomial complexity of the ABESS algorithm.
Theorem 2. *Suppose conditions* 1 *and* 4 *hold. Assume conditions* 2, 3, and 6 *hold with* s_{max}. *The computational complexity of ABESS for a given* s_{max} *is*

$$O\left(\left(s_{max}\log\frac{\|\boldsymbol{y}\|_2^2}{\log p\log\log n} + \frac{n\|\boldsymbol{y}\|_2^2}{\log p\log\log n}\right)(np s_{max} + ns_{max}^2 + k_{max} p s_{max})\right).$$

If $s \ge s^*$, Algorithm 1 will find the true active set in high probability under conditions 1–4 (Lemma 1). Furthermore, by splicing, the loss function decreases drastically at the first several iterations and the convergence rate $O(\log\frac{\|\boldsymbol{y}\|_2^2}{s\log p\log\log n})$ of Algorithm 1 is presented in Theorem 3. However, if $s < s^*$, we can determine the iterations $O(\frac{n\|\boldsymbol{y}\|_2^2}{s\log p\log\log n})$ of Algorithm 1 by using thresholding τ_s to exclude useless splicing. Thus, we can show that the number of iterations of Algorithm 1 is polynomial.

Statistical Theory. Let $\gamma_s(n,p) = O(\exp\{\log p - \frac{K_s n b^*}{s}\}) + O(\exp\{\log p - \frac{K_s n}{s^*}\})$, where K_s is some constant depending on s. The following lemma gives an interesting property of the active set output by Algorithm 1.
Lemma 1. *Suppose* $(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{d}}, \hat{\mathcal{A}}, \hat{\mathcal{I}})$ *is the solution of Algorithm* 1 *for a given support size* $s \ge s^*$ *and conditions* 1–4 *hold. Then, we have*

$$\text{P}(\hat{\mathcal{A}} \supseteq \mathcal{A}^*) \ge 1 - \gamma_s(n,p).$$

Furthermore, if conditions 5 *and* 6 *hold,*

$$\lim_{n\to\infty} \text{P}(\hat{\mathcal{A}} \supseteq \mathcal{A}^*) = 1.$$

Especially, if $s = s^*$, *we have*

$$\lim_{n\to\infty} \text{P}(\hat{\mathcal{A}} = \mathcal{A}^*) = 1.$$

Lemma 1 indicates that our estimator of the active set will eventually include the true active set. The next theorem characterizes the number of iterations and the ℓ_2 bound error of the splicing method.
Theorem 3. *Suppose* $(\boldsymbol{\beta}^m, \boldsymbol{d}^m, \mathcal{A}^m, \mathcal{I}^m)$ *is the* m*th iteration of Algorithm* 1 *for a given support size* $s \ge s^*$. *Suppose conditions* 1–4 *hold. Then, with probability* $1 - \gamma_s(n,p)$, *we have*

1)

$$\mathcal{A}^m \supseteq \mathcal{A}^*, \text{ if } m \geq \log_{\frac{1}{\delta_s}} \left(\frac{\|\boldsymbol{y}\|_2^2}{C_{s,1}} \right),$$

where $C_{s,1} = n(1-\Delta)\left(c_-(s) - \frac{\theta_{s,s}^2}{c_-(s)^2}\right)b^*$, b^* *is the minimal signal strength, and* δ_s *is defined in condition* 3;
2)

$$\|\boldsymbol{\beta}^m - \boldsymbol{\beta}^*\|_2^2 \leq C_{s,2}\delta^m \|\boldsymbol{y}\|_2^2.$$

where $C_{s,2} = (1 + \Delta + \frac{\theta_{s,s}}{c_-(s)})/\left((1-\Delta)n(c_-(s) - \frac{\theta_{s,s}^2}{c_-(s)})\right)$.

With the threshold τ_s, Theorem 3 suggests that our splicing method terminates at a logarithm number of iterations. The estimation error decays geometrically.

The next theorem guarantees that the splicing method can recover the true active set with a high probability.

Theorem 4 (Consistency of Best-Subset Recovery). *Suppose conditions* 1, 4, *and* 7 *hold. Assume conditions* 2, 3, *and* 6 *hold with* s_{max}. *Then, under the information criterion SIC, with probability* $1 - O(p^{-\alpha})$, *for some positive constant* $\alpha > 0$ *and a sufficiently large* n, *the ABESS algorithm selects the true active set, that is,* $\hat{\mathcal{A}}_{s_{min}} = \mathcal{A}^*$.

Theorem 4 implies that the solution of the splicing method is the same as the oracle least-squares estimator with an unknown sparsity level. Since our approach can recover the true active set, we can directly deduce the asymptotic distribution of $\boldsymbol{\beta}$.

Corollary 1 (Asymptotic Properties). *Suppose the assumptions and conditions in Theorem* 4 *hold. Then, with a high probability, the solution* $\hat{\boldsymbol{\beta}}_{s_{min}}$ *of ABESS is the oracle estimator, i.e.,*

$$P\{\hat{\boldsymbol{\beta}}_{s_{min}} = \hat{\boldsymbol{\beta}}^o\} = 1 - O(p^{-\alpha}),$$

where $\alpha > 0$ *and* $\hat{\boldsymbol{\beta}}^o$ *is the least-squares estimator given the true active set* $\mathcal{A}^*$. *Furthermore,*

$$\hat{\boldsymbol{\beta}}_{\mathcal{A}^*} \sim N(\boldsymbol{\beta}^*_{\mathcal{A}^*}, \boldsymbol{\Sigma}),$$

where $\boldsymbol{\Sigma} = (\boldsymbol{X}_{\mathcal{A}^*}^\top \boldsymbol{X}_{\mathcal{A}^*})^{-1}$.

Simulation

In this part, we compare the proposed ABESS algorithm with other variable selection algorithms under Model 1, where the rows of the design matrix are *i.i.d*-sampled from the multivariate normal distribution with mean $\mathbf{0}$ and covariance matrix Σ. The n error terms are *i.i.d*-drawn from the normal distribution $N(0, \sigma^2)$.

We consider four criteria to assess the methods. The first two criteria, true-positive rate (TPR) and true-negative rate (TNR), are used to evaluate the performance of variable selection. The estimation accuracy for $\boldsymbol{\beta}$ is measured by the relative error (ReErr): $\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|_2 / \|\boldsymbol{\beta}\|_2$. We also examine the dispersion between the sparsity level estimation $\hat{s}$ and the ground truth, which is measured by the sparsity-level error (SLE): $\hat{s} - s^*$. All simulation results are based on 100 synthetic datasets.

Low-Dimensional Case. We begin with a low-dimensional setting and compare ABESS and all-subsets regression (ASR), which exhaustively searches for the best subsets of the explanatory variables to predict the response via an efficient branch-and-bound algorithm (21). We use SIC (ASR-SIC) to select a model size for ASR.

We adopt a simulation model from ref. 6. Specifically, the coefficient is fixed at $\boldsymbol{\beta} = (3, 1.5, 0, 0, 2, 0, 0, 0)^\top$, the covariance matrix Σ has a decayed structure, i.e., $\Sigma_{ij} = 0.5^{|i-j|}$ for all $i, j \in \{1, \ldots, p\}$. The pair of sample size and noise level (n, σ) varies as $(40, 3), (40, 1)$, and $(60, 1)$. It can be seen from Table 1 that when the noise level is large but the sample size is small, the performance of ABESS and ASR is close, although ASR is slightly better. When the noise level reduces, the slight advantage of ASR-SIC disappears. The fact that ABESS performs as well as the exhaustive ASR algorithm, when the setting is simple enough for the latter to be computationally feasible, demonstrates the power of ABESS in selecting the best subset.

Next, we study the computational time and computational complexity of the ASR and ABESS algorithms by adding zeros to $\boldsymbol{\beta}$ in the previous experiment to form a new $\boldsymbol{\beta}$ of a total of p coefficients. Without loss of generality, we consider the runtime of algorithms when p increases from 20 to 40 with step size 1. Fig. 1 presents the simulation results. On the one hand, from Fig. 1*A*, we can see that the difference between ASR-SIC and ABESS in the three criteria are all under control in interval $(-5 \times 10^{-3}, 5 \times 10^{-3})$, and, hence, we can conclude that ABESS and ASR have a negligible difference in this setting. On the other hand, from Fig. 1 *B* and *C*, the computational time of ASR is 20 s when dimensionality reaches 40, while that of ABESS is less than 0.03 s. More importantly, from Fig. 1*B*, the computational time of ABESS grows linearly when the dimension increases, as proven in *Theorem 2*. In contrast, from Fig. 1*C*, the runtime of ASR increases exponentially. In summary, ABESS not only can recover the support but also is computationally fast.

High-Dimensional Case. We consider the case when the dimension is in hundreds or even thousands, for which the exhaust search is computationally infeasible. It is of interest to compare ABESS and modern variable selection algorithms, including LASSO (4), SCAD (6), and MCP (7). The solutions of the three algorithms are given by the coordinate descent algorithm (22, 23) implemented in R packages *glmnet* and *ncvreg*. For all of these methods, we use SIC to select the optimal regularized parameters. We also consider cross-validation (CV), a widely used method, to select the tuning parameter. For MCP/SCAD/LASSO, the l regularized parameters to be selected are prespecified values following the default setting in R packages *glmnet* and *ncvreg*. For a fair comparison, the input argument of the ABESS algorithm, s_{max}, is also set as l. Here, l is set to be $[\frac{n}{\log p \log\log n}]$. Note that the concavity parameter γ of the SCAD and MCP penalties is fixed at 3.7 and 3, respectively (6, 7).

The dimension, p, of the explanatory variables increases as 500, 1,500, and 2,500, but only 10 randomly selected variables from them would affect the response. Among the 10 effective variables, 3 of them have a strong effect, 4 of them have a moderate effect, and the rest have a weak effect. Here, a strong/moderate/weak effect means that a coefficient is sampled from a zero-mean normal distribution with SD10/5/2. We consider two structures of Σ. The first one is the uncorrelated structure $\Sigma_{ij} = I(i = j)$, and the second one is a constant

Table 1. Simulation results in the low-dimensional setting

Method	TPR	TNR	ReErr	SLE
$n = 40, \sigma^2 = 3$				
ABESS	0.90 (0.17)	0.86 (0.15)	0.20 (0.19)	0.40 (0.89)
ASR-SIC	0.91 (0.17)	0.87 (0.15)	0.14 (0.13)	0.38 (0.85)
$n = 40, \sigma^2 = 1$				
ABESS	1.00 (0.00)	0.87 (0.14)	0.02 (0.02)	0.63 (0.72)
ASR-SIC	1.00 (0.00)	0.87 (0.14)	0.02 (0.02)	0.63 (0.72)
$n = 60, \sigma^2 = 1$				
ABESS	1.00 (0.00)	0.90 (0.13)	0.01 (0.01)	0.48 (0.64)
ASR-SIC	1.00 (0.00)	0.90 (0.13)	0.01 (0.01)	0.49 (0.64)

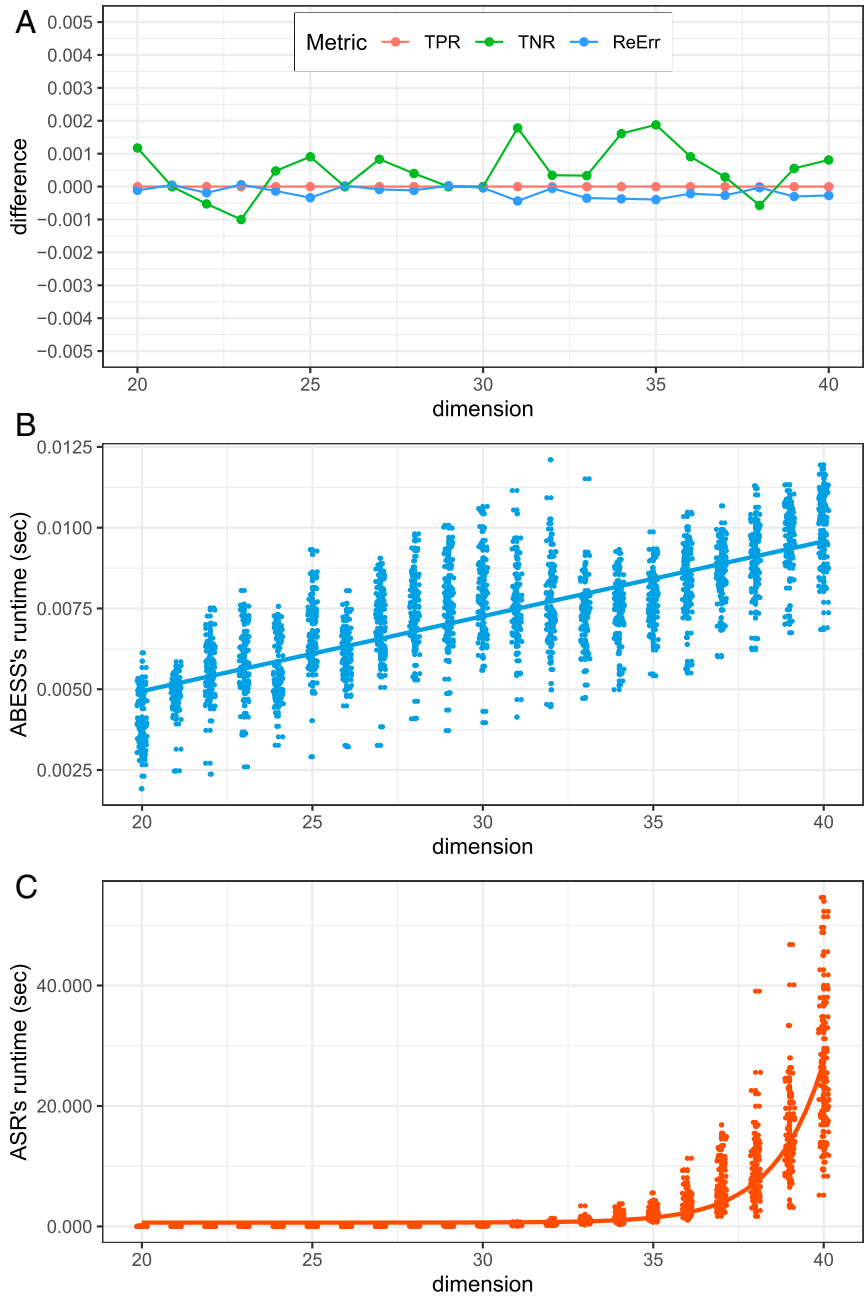

Fig. 1. (*A*) For each of the three metrics (TPR, TNR, and ReErr), the difference (*y* axis) is calculated by subtracting an ABESS metric from its corresponding ASR metric. Different colors correspond to different metrics. (*B*) Dimension (*x* axis) versus ABESS's runtime (*y* axis) scatterplot. The blue straight line is characterized by equation $y = a + bx$. (*C*) Dimension (*x* axis) versus ASR's runtime (*y* axis) scatterplot. The red curve is $y = a + b2^x$. In *B* and *C*, the coefficients *a*, *b* are estimated by the ordinary least squares.

structure $\Sigma_{ij} = 0.8^{I(i \neq j)}$, corresponding to the case that any two explanatory variables are highly correlated. The sample size n is fixed at 500, and the noise level σ^2 is fixed at 1.

The simulation results are presented in Fig. 2 *A* and *B*. A few observations are noteworthy. First, among all of the methods, ABESS or the CV-based LASSO estimator have the best performance for correctly identifying the true effective variables; moreover, ABESS can reasonably control the false-positive rate at a low level like SCAD and MCP. Second, SIC helps ABESS efficiently detect the true model size and its SLE approaches to 0. In conjunction with the first point, the empirical results demonstrate ABESS's performance as proven in *Theorem 4*. In contrast, the MCP and SCAD underestimate the model size, whereas LASSO overestimates it. Also, we see that like BIC (24), SIC avoids overfitting (see additional simulation studies in *SI Appendix*). Finally, the parameter estimation of ABESS is superior to the other algorithms because ABESS not only effectively recovers the support set but also yields an unbiased parameter estimate. Fig. 2*C* compares the runtime. We see that ABESS's runtime is computationally efficient. Furthermore, as expected, ABESS is much faster than the CV-based LASSO/SCAD/MCP methods.

Summary

We present an iterative splicing method that distinguishes the active set from the inactive set iteratively in variable selection.

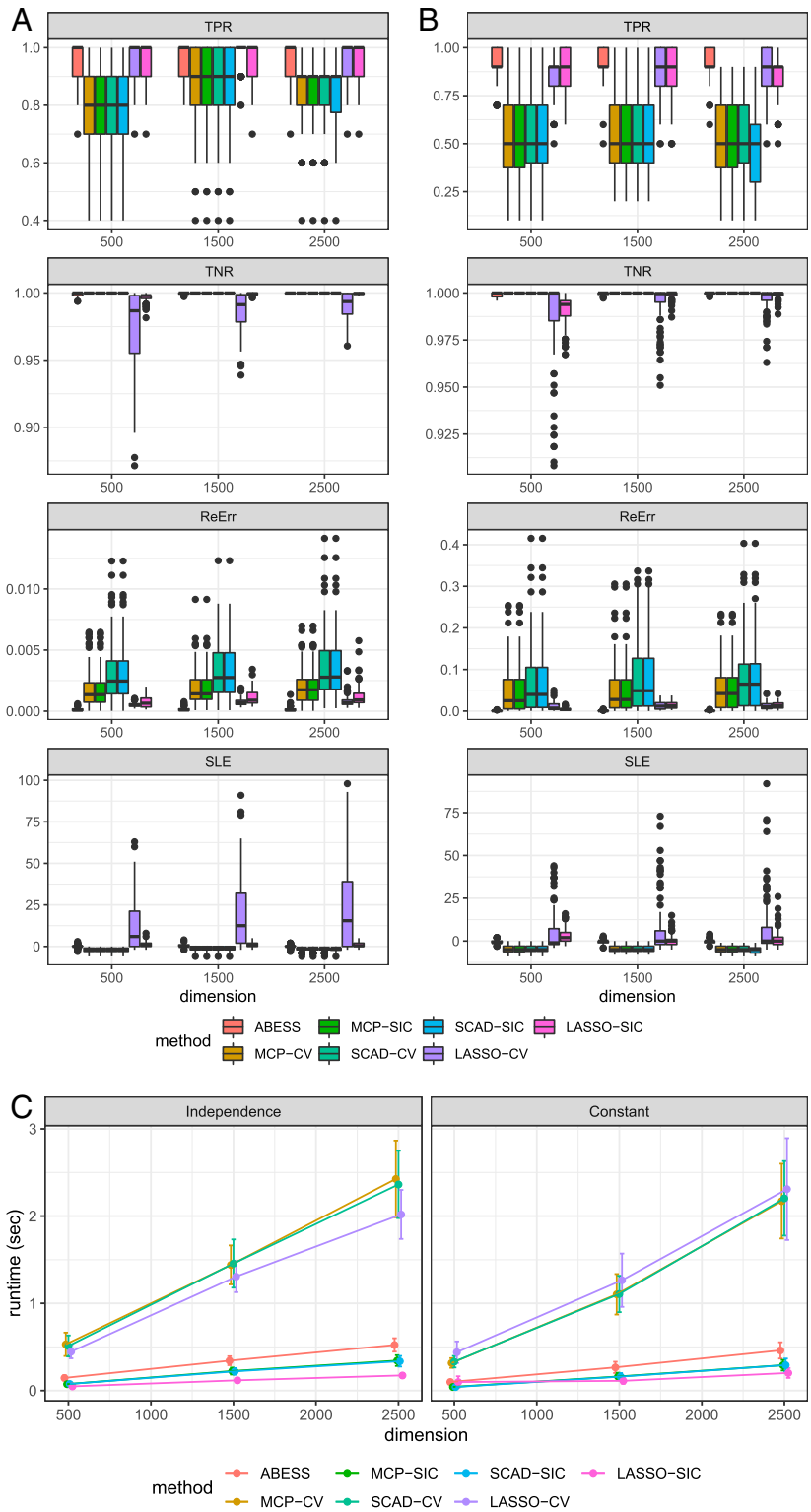

Fig. 2. (*A* and *B*) The boxplots of TPR, TNR, ReErr, and SLE of different algorithms in the high-dimensional setting when any two covariates have no correlation (*Left*) and constant correlation 0.8 (*Right*). (*C*) Average runtime comparison under two correlation structure settings: uncorrelated and constant. The runtime (*y* axis) is measured in seconds.

The estimated active set is shown to contain the true active set when the given support size is no less than the true size, or to be included in the true active set when the given support size is less than the true size. We also introduce a selection information criterion to adaptively determine the sparsity level, which can guarantee to choose the true active set with a high probability. We show that our solution is globally optimal for the Lagrangian of Eq. **2** with SIC and has the oracle properties with a high probability. Numerical results demonstrate the theoretical properties of ABESS. However, when there are a large number of weak

effects, the ambiguity makes it challenging for us to detect signals. ABESS as well as other methods such as LASSO, SCAD, and MCP face a similar difficulty. How to perform an effective subset selection with many weak effects warrants further research.

Data Availability. All study data are included in the article and *SI Appendix*.

ACKNOWLEDGMENTS. X.W.'s research is partially supported by National Key Research and Development Program of China Grant 2018YFC1315400, Natural Science Foundation of China (NSFC) Grants 71991474 and 11771462, and Key Research and Development Program of Guangdong, China Grant 2019B020228001. H.Z.'s research is supported in part by US NIH Grants R01HG010171 and R01MH116527 and NSF Grant DMS1722544. C.W.'s research is partially supported by NSFC Grant 11801540, Natural Science Foundation of Anhui Grant BJ2040170017, and Fundamental Research Funds for the Central Universities Grant WK2040000016.

1. H. Akaike, "Information theory and an extension of the maximum likelihood principle" in *Selected Papers of Hirotugu Akaike*, E. Parzen, K. Tanabe, G. Kitagawa, Eds. (Springer, 1998), pp. 199–213.
2. G. Schwarz, Estimating the dimension of a model. *Ann. Stat.* **6**, 461–464 (1978).
3. C. L. Mallows, Some comments on Cp. *Technometrics* **15**, 661–675 (1973).
4. R. Tibshirani, Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc. B* **58**, 267–288 (1996).
5. H. Zou, The adaptive lasso and its oracle properties. *J. Am. Stat. Assoc.* **101**, 1418–1429 (2006).
6. J. Fan, R. Li, Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Am. Stat. Assoc.* **96**, 1348–1360 (2001).
7. C. Zhang, Nearly unbiased variable selection under minimax concave penalty. *Ann. Stat.* **38**, 894–942 (2010).
8. H. Hazimeh, R. Mazumder, Fast best subset selection: Coordinate descent and local combinatorial optimization algorithms. *Oper. Res.*, in press.
9. B. K. Natarajan, Sparse approximate solutions to linear systems. *SIAM J. Comput.* **24**, 227–234 (1995).
10. T. Blumensath, M. E. Davies, Iterative hard thresholding for compressed sensing. *Appl. Comput. Harmon. Anal.* **27**, 265–274 (2009).
11. M. Hintermüller, K. Ito, K. Kunisch, The primal-dual active set strategy as a semismooth Newton method. *SIAM J. Optim.* **13**, 865–888 (2002).
12. D. Bertsimas, A. King, R. Mazumder, Best subset selection via a modern optimization lens. *Ann. Stat.* **44**, 813–852 (2016).
13. L. Wang, Y. Kim, R. Li, Calibrating non-convex penalized regression in ultra-high dimension. *Ann. Stat.* **41**, 2505–2536 (2013).
14. J. Chen, Z. Chen, Extended Bayesian information criteria for model selection with large model spaces. *Biometrika* **95**, 759–771 (2008).
15. C. Zhang, J. Huang, The sparsity and bias of the lasso selection in high-dimensional linear regression. *Ann. Stat.* **36**, 1567–1594 (2008).
16. P. J. Bickel, Y. Ritov, A. B. Tsybakov, Simultaneous analysis of Lasso and Dantzig selector. *Ann. Stat.* **37**, 1705–1732 (2009).
17. G. Raskutti, M. J. Wainwright, B. Yu, Restricted eigenvalue properties for correlated Gaussian designs. *J. Mach. Learn. Res.* **11**, 2241–2259 (2010).
18. E. J. Candes, T. Tao, Decoding by linear programming. *IEEE Trans. Inf. Theor.* **51**, 4203–4215 (2005).
19. J. Huang, Y. Jiao, Y. Liu, X. Lu, A constructive approach to l_0 penalized regression. *J. Mach. Learn. Res.* **19**, 1–37 (2018).
20. Z. Zheng, M. T. Bahadori, Y. Liu, J. Lv, Scalable interpretable multi-response regression via seed. *J. Mach. Learn. Res.* **20**, 1–34 (2019).
21. A. Miller, *Subset Selection in Regression* (CRC Press, 2002).
22. J. Friedman, T. Hastie, R. Tibshirani, Regularization paths for generalized linear models via coordinate descent. *J. Stat. Software* **33**, 1–22 (2010).
23. P. Breheny, J. Huang, Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *Ann. Appl. Stat.* **5**, 232–253 (2011).
24. Y. Zhang, R. Li, C. L. Tsai, Regularization parameter selections via generalized information criterion. *J. Am. Stat. Assoc.* **105**, 312–323 (2010).