HIGH-DIMENSIONAL INFERENCE BASED ON THE LEAVE-ONE-COVARIATE-OUT LASSO PATH

XIANGYANG CAO, KARL GREGORY, AND DEWEI WANG

ABSTRACT. We propose a new measure of variable importance in high-dimensional regression based on the change in the LASSO solution path when one covariate is left out. The proposed procedure provides a novel way to calculate variable importance and conduct variable screening. In addition, our procedure allows for the construction of P-values for testing whether each coefficient is equal to zero as well as for testing hypotheses involving multiple regression coefficients simultaneously; bootstrap techniques are used to construct the null distribution. For low-dimensional linear models, our method can achieve higher power than the t-test. Extensive simulations are provided to show the effectiveness of our method. In the high-dimensional setting, our proposed solution path based test achieves greater power than some other recently developed high-dimensional inference methods.

1. INTRODUCTION

We consider the linear regression model

$$Y = \mathbf{X}\beta + \epsilon, \tag{1}$$

where $\mathbf{X} = [X_1^T, X_2^T, \ldots, X_n^T]^T$ with $X_i \in \mathbb{R}^p$, $Y \in \mathbb{R}^n$, $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2 \mathbf{I}_n)$, where $\mathbf{I}_n$ is the $n \times n$ identity matrix, and $\beta \in \mathbb{R}^p$ is a vector of unknown regression coefficients. We consider both the cases $p > n$ and $p \leq n$.

We propose a measure of variable importance based on the change in the LASSO solution path due to removing a covariate from the model. Regarding the LASSO solution path

$$\hat{\beta} := \hat{\beta}(\lambda) = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}}(||Y - \mathbf{X}\beta||_2^2 + \lambda||\beta||_1), \quad \lambda > 0 \tag{2}$$

as a function of λ taking values in $(0, \infty)$ and returning values $\hat{\beta}(\lambda)$ in $\mathbb{R}^p$, we propose to measure the importance of covariate X_j, for any

---

$j \in \{1, \ldots, p\}$, by comparing the path $\hat{\beta}$ to the path

$$\hat{\beta}^{(-j)} := \hat{\beta}^{(-j)}(\lambda) = \underset{\beta \in \mathbb{R}^p, \beta_j = 0}{\operatorname{argmin}} (||Y - \mathbf{X}\beta||_2^2 + \lambda ||\beta||_1), \quad \lambda > 0, \quad (3)$$

which is the LASSO solution path when the covariate X_j is removed from the model. Herein, for a vector $\mathbf{v} = (v_1, \ldots, v_K)^T$, $||\mathbf{v}||_2^2 = \sum_{k=1}^{K} v_k^2$ and $||\mathbf{v}||_1 = \sum_{k=1}^{K} |v_k|$. We will refer to $\hat{\beta}^{(-j)}$ as the leave-one-covariate-out solution path, or the LOCO path of the LASSO. It is important to note that for a given j, $\hat{\beta}_j^{(-j)}(\lambda) = 0$ for all λ. We reason that if covariate X_j is important, its importance will be reflected in a large difference between the paths $\hat{\beta}$ and $\hat{\beta}^{(-j)}$, whereas if it is not important, the difference between the paths $\hat{\beta}$ and $\hat{\beta}^{(-j)}$ will be small.

The measure of variable importance we propose, which we shall call the LOCO path statistic, can be used for variable selection and variable screening; moreover, we suggest that it can be used as a test statistic for testing the hypotheses H_0: $\beta_j = 0$ versus H_1: $\beta_j \neq 0$. We also use the LOCO solution path idea to construct a test statistic for testing more complicated hypotheses involving several coefficients, specifically hypotheses of the form

$$H_0: \beta_j = \beta_{j,0}, \text{ for all } j \in \mathcal{A} \text{ versus } H_1: \beta_j \neq \beta_{j,0} \text{ for some } j \in \mathcal{A},$$

for some $\{\beta_{j,0}, j \in \mathcal{A}\}$, where $\mathcal{A} \subset \{1, \ldots, p\}$. We propose a bootstrap procedure to calibrate the rejection regions of hypothesis tests based on the LOCO solution path.

We now place our ideas in the literature: the LASSO was introduced in 1996 in [20], and has since been one of the most popular estimators for the linear regression model of (1), particularly in the $p > n$ case. It belongs to a class of penalized estimators designed to promote sparsity among the estimated regression coefficients in order to achieve simultaneous variable selection and estimation. Implementing the LASSO requires choosing a value, usually via cross validation, of the tuning parameter λ, which governs the sparsity and shrinkage towards zero of the estimated regression coefficients. Although the LASSO is a powerful tool, the LASSO estimator has a very complicated sampling distribution, so that statistical inference based on LASSO estimators is problematic.

Other estimators for model (1) with $p > n$ have been proposed which have, under some conditions, limiting normal distributions, such as the desparsified LASSO estimator introduced by [21] and [23] as well as the estimator introduced by [13]; these methods enable inference, but

a downside is that they require the choice of an additional tuning parameter and inferences may be very sensitive to the choice of tuning parameter. The adaptive LASSO estimator of [24], under some conditions and with tuning parameters appropriately chosen, has a limiting normal distribution (for non-zero coefficients), though convergence seems to be slow; a bootstrap procedure has been shown to be consistent for the adaptive LASSO in [5]. A bootstrap method for the LASSO is proposed in [4] and [3], which is consistent for a modified LASSO and adaptive LASSO estimator. A sequential significance testing procedure for variables entering the model along the LASSO solution path was proposed in [16]. Inferential methods for the high-dimensional linear model based on sample splitting, for example in [22] and [18], have also been proposed and implemented with success.

As variable selection methods, sure independence screening (SIS) and iterative sure independence screening (ISIS) are proposed in [9] for ultra-high dimensional linear regression. Ultra-high dimensional regression focuses on the settings with $\log(p) = O(n^\varsigma)$. It has been extended to GLM [10], GAM [8] and multivariate regression models [14]. Although these methods enjoy the sure screening property [9], SIS only considers the marginal contribution of each variable to the response.

To our knowledge, however, not much work has focused on analyzing and summarizing the information contained in the entire solution path of the LASSO with respect to the importance of each variable. We propose to consider the LASSO solution path in its entirety, and then measure how it changes when we leave one covariate out.

The idea of leave-one-covariate-out (LOCO) inference is not new. The following LOCO-based procedure for measuring variable importance is described in [15]: Let $\hat{\mu}$ be an estimate of $E(Y|\mathbf{X})$ based on some training data $(\mathbf{X}, Y)$, and let $\hat{\mu}_{(-j)}$ be the same estimator based on the training data $(\mathbf{X}_{(-j)}, Y)$, where $\mathbf{X}_{(-j)}$ is the matrix $\mathbf{X}$ with column j removed. Then we measure the excess prediction error on new data (X_{new}, Y_{new}) as

$$|Y_{new} - \hat{\mu}_{(-j)}(X_{new})| - |Y_{new} - \hat{\mu}(X_{new})|,$$

where the "new" data can come from crossvalidation testing sets or from a separate testing data set. The larger the above quantity, the greater importance we assign to covariate X_j, as it measures how much worse our predictions become due to removing covariate X_j.

Permutation feature importance, introduced by [1] and generalized by [11], is a similar to the LOCO approach to measuring variable importance; instead of removing covariate X_j from the model, the observed

values of covariate X_j are randomly permuted. By this permutation, the association between covariate X_j and the response is broken and the resulting model is different from the one fit to the original data.

What we propose falls into the framework of LOCO variable importance and inference; however, rather than measuring the change in the prediction error due to removing a covariate, we consider the change in the LASSO solution path.

This paper is organized as follows: Section 2 defines our measure of variable importance based on the change in the LASSO solution path due to the removal of a covariate and discusses its use as a variable selection and variable screening tool. Section 3 explains how we propose to use the LOCO solution path idea to construct test statistics for testing hypotheses about the regression coefficients. We also describe a bootstrap procedure for estimating the null distribution of our LOCO path-based test statistics. Section 4 presents simulation results and Section 5 illustrates the method on a real data set. Section 6 provides additional discussion.

2. The leave-one-covariate-out path statistic

To formulate our metric for the difference between the LASSO solution path $\hat{\beta}$ defined in (2) and the LOCO solution path of the LASSO defined in (3), we define a quantity for functions taking values in $(0, \infty)$ and returning values in $\mathbb{R}^p$. Firstly, for any function g taking values in $(0, \infty)$ and returning values in $\mathbb{R}$, let

$$\|g\|_s = \begin{cases} (\int_0^\infty |g(\lambda)|^s d\lambda)^{1/s}, & 0 < s < \infty \\ \sup_{\lambda > 0} |g(\lambda)|, & s = \infty. \end{cases}$$

Secondly, for a vector $x \in \mathbb{R}^p$, let

$$\|x\|_t = \begin{cases} (\sum_{j=1}^p |x_j|^t)^{1/t}, & 0 < t < \infty \\ \max_{1 \le j \le p} |x_j|, & t = \infty. \end{cases}$$

Now, for a function f taking values in $(0, \infty)$ and returning values in $\mathbb{R}^p$ such that $f(\lambda) = (f_1(\lambda), \ldots, f_p(\lambda))^T$, define the quantity $\|f\|_{s,t}$ as

$$\|f\|_{s,t} = \|(\|f_1\|_s, \ldots, \|f_p\|_s)^T\|_t.$$

Having defined a quantity for functions taking values in $(0, \infty)$ and returning values in $\mathbb{R}^p$, we define the LOCO path statistic for covariate X_j as

$$T_j(s, t) = \|\hat{\beta} - \hat{\beta}^{(-j)}\|_{s,t},$$

which measures the change in the LASSO solution path due to removing covariate X_j from the model.

In practice, it is convenient to use $s = t$; if $s = t = q$, we have

$$
T_j(q, q) = \begin{cases} \left(\sum_{k=1}^{p} \int_{0}^{\infty} |\hat{\beta}_k(\lambda) - \hat{\beta}_k^{(-j)}(\lambda)|^q d\lambda \right)^{\frac{1}{q}} & q < \infty \\ \max_{1 \leq k \leq p} \sup_{\lambda > 0} |\hat{\beta}_k(\lambda) - \hat{\beta}_k^{(-j)}(\lambda)| & q = \infty. \end{cases}
$$

We recommend using $q = 1$ or $q = 2$ in practice. We have found that under $q = \infty$ our hypothesis test tend to have lower power, so we do not recommend this setting. We illustrate this in the simulation section.

We posit that the quantity $T_j(s, t)$ will be large if $\beta_j \neq 0$ and small if $\beta_j = 0$, for $j = 1, \ldots, p$, so that $T_j(s, t)$ may serve as a measure of variable importance for covariate X_j. Since the LASSO solution path is piecewise linear, we can calculate $T_j(s, t)$ exactly. More details about the calculation can be found in the section S.1 of the Supplementary Material.

2.1. **The LOCO path statistic as a measure of variable importance.** For the sake of illustration, let us consider one special case of $T_j(s, t)$, with $s = t = 1$. We have

$$
T_j(1, 1) = ||\hat{\beta} - \hat{\beta}^{(-j)}||_{1,1} = \sum_{k=1}^{p} \int_{0}^{\infty} |\hat{\beta}_k(\lambda) - \hat{\beta}_k^{(-j)}(\lambda)| d\lambda,
$$

which is equal to the sum of all the areas under the curves $|\hat{\beta}_k(\cdot) - \hat{\beta}_k^{(-j)}(\cdot)|$, $k = 1, \ldots, p$. We depict this for the following simple example: We generate one dataset from the linear regression model (1) with $n = 100$, $p = 4$ and $\beta = (1, 1, 0, 0)^T$, and compute the test statistics $T_1(1, 1)$ and $T_3(1, 1)$. The left and right panels of Figure 1 show the original LASSO solution path as well as the solution path after removing the first and third covariates, respectively, from the model. In each panel, the sum of the areas of the shaded regions is the value of the test statistic.

We propose to summarize the importance of the variables measured by the LOCO path statistic in the following way. After standardizing the values of $T_j(s, t)$, $j = 1, \ldots, p$, so that they sum to one, for example by defining

$$
\overline{T}_j(s, t) = T_j(s, t) \left(\sum_{k=1}^{p} T_k(s, t) \right)^{-1}, \quad j = 1, \ldots, p,
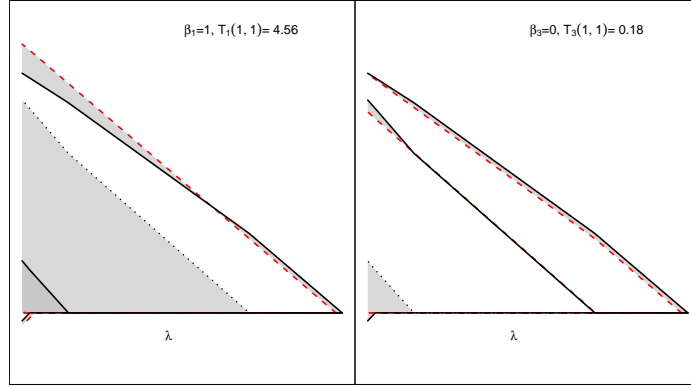$$

FIGURE 1. Shaded areas show how $T_j(1, 1)$ measures the change in LASSO path. Black solid line depicts the solution path before removal. Black dotted line depicts the solution path of the covariates being removed. Red dashed line depicts the solution path after removal. Left: $T_1(1, 1)$. Right: $T_3(1, 1)$.

we can make a plot such as the one in Figure 2, which shows the values of $\overline{T}_1(1, 1), \ldots, \overline{T}_{12}(1, 1)$, expressed as percentages. This is based on a single dataset simulated from (1) with $n = 100$, $p = 12$, $\beta = (1, 1, 1, 0, \ldots, 0)^T$, for the sake of illustration. The first three covariates are seen to have the highest importance according to the LOCO path statistic.

Furthermore, we consider attaching to the variable importance a measure of uncertainty. The LOCO path $\hat{\beta}_k^{(-j)}(\lambda)$ could be fitted by permuting variable j in $\mathbf{X}$. By permuting variable j in $\mathbf{X}$, we break the association between X_j and Y, which has an effect similar to removing variable j. By permuting the observed values of covariate j multiple times we can obtain an interval for the variable importance. Figure 2 also shows the permutation interval calculated for the importance measure of each variable.

2.2. **Variable screening in ultra-high dimensional settings.** The so-called ultra-high dimensional setting was discussed in [9], where the dimensionality p grows exponentially ($\log(p) = O(n^\zeta)$) as n grows. For ultra-high dimensional problems, preliminary variable screening is often done to reduce the dimension of the data.

Our method naturally adapts to ultra-high dimensional settings. By calculating how the removal of each variable will alter the LASSO solution path, we have a simple way to screen out variables which are likely to be irrelevant. Our method uses the information contained in
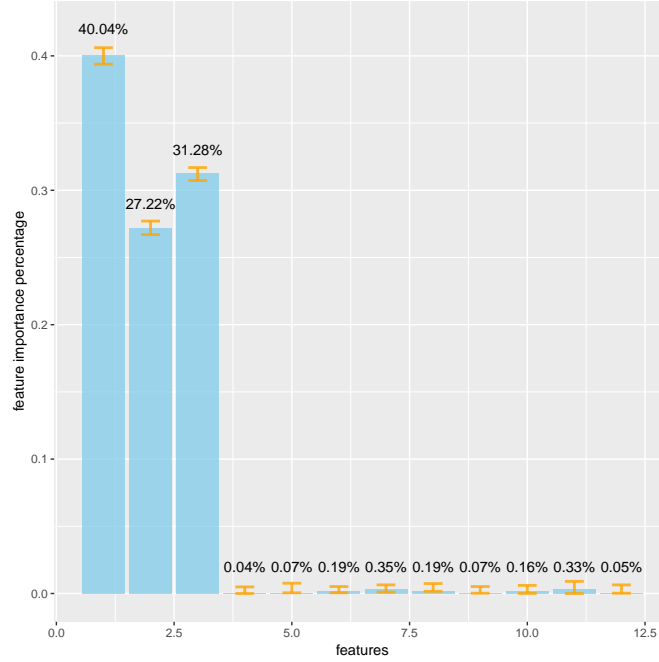
FIGURE 2. Variable importance for all variables based on the LOCO path statistic. The error bar is our permutation interval. The variable importance is also shown as percentages on top of the error bar.

the LASSO solution path, which utilizes both joint and marginal information. One interesting result of LASSO in the high-dimensional setting is that some variables never enter the model. If we take a closer look at the solution path of such variables, they are equal to 0 for all values of λ. If we were to use cross validation to select the LASSO tuning parameter and obtain the final selection results, these variables would never be selected. This means we can safely screen out these variables at the beginning.

Based on this intuition, we suggest the following screening procedure: Compute the solution path with all variables in the model. Then remove one variable at a time and compute the LOCO solution path; compute the values $T_1(s,t), \ldots, T_p(s,t)$, which compare the solution path based on the full set of covariates to the LOCO solution paths. Then screen out variables for which $T_j(s,t) \leq \epsilon$, where ϵ is a user-specified threshold. Choosing $\epsilon = 0$ discards only those variables which never enter the solution path. We can also rank $T_j(s,t)$ and only select the top K variables, where we might choose K to be $n-1$ and n is the sample size.

3. Hypothesis testing using the LOCO path idea

We now consider using the LOCO path idea to test hypotheses of the form

$$H_0\text{: } \beta_j = \beta_{j,0} \text{ for all } j \in \mathcal{A} \text{ versus } H_1\text{: } \beta_j \neq \beta_{j,0} \text{ for some } j \in \mathcal{A}, \quad (4)$$

for some $\{\beta_{j,0}, j \in \mathcal{A}\}$, where $\mathcal{A} \subset \{1, \ldots, p\}$. We first calculate the LASSO solution path with all variables included. Next, we compute the solution path subject to the constraint specified by the null hypothesis, which is given by

$$\hat{\beta}_0 := \hat{\beta}_0(\lambda) = \operatorname*{argmin}_{\beta \in \mathbb{R}^p, \beta_j = 0 \in \mathcal{A}} (||(Y - \mathbf{X}_{\mathcal{A}}\beta_{0,\mathcal{A}}) - \mathbf{X}\beta||_2^2 + \lambda||\beta||_1), \quad (5)$$

where $\beta_{0,\mathcal{A}} = (\beta_{j,0}, j \in \mathcal{A})^T$ and $\mathbf{X}_{\mathcal{A}}$ is the matrix constructed out of the columns of $\mathbf{X}$ with indices in $\mathcal{A}$.

We then suggest as a test statistic for testing H_0 versus H_1 the quantity

$$T_0(s, t) = ||\hat{\beta} - \hat{\beta}_0||_{s,t}, \quad (6)$$

which compares the solution paths $\hat{\beta}_0$ and $\hat{\beta}$. For testing the hypotheses

$$H_0\text{: } \beta_j = 0 \text{ versus } H_1\text{: } \beta_j \neq 0,$$

for some $j \in \{1, \ldots, p\}$, we have $\hat{\beta}_0 = \hat{\beta}^{(-j)}$, so that the test statistic $T_0(s, t)$ is equal to the LOCO path variable importance statistic $T_j(s, t) = ||\hat{\beta} - \hat{\beta}^{(-j)}||_{s,t}$.

3.1. **A bootstrap estimator of the null distribution.** In order to test the hypotheses in (4) using the test statistic $T_0(s, t)$ in (6), we need to know the distribution of $T_0(s, t)$ under H_0. We propose estimating this null distribution using a residual bootstrap procedure.

In order to obtain residuals from which to resample, we propose obtaining an initial estimator $\tilde{\beta}$, which we will discuss at the end of this section, of the vector β from which we can obtain residuals

$$\tilde{\epsilon} = Y - \mathbf{X}\tilde{\beta}.$$

Let $\tilde{Y}^*$ be the $n \times 1$ random vector with entries given by $\tilde{Y}_i^* = X_i^T\tilde{\beta} + \tilde{\epsilon}_i^*$, for $i = 1, \ldots, n$, where $\epsilon_1^*, \ldots, \epsilon_n^*$ are sampled with replacement from the entries of the residual vector $\tilde{\epsilon} = (\tilde{\epsilon}_1, \ldots, \tilde{\epsilon}_n)^T$.

For testing the hypotheses in (4), the bootstrap versions $\hat{\beta}^*$ and $\hat{\beta}_0^*$ of $\hat{\beta}$ and $\hat{\beta}_0$ are constructed as

$$\hat{\beta}^* := \hat{\beta}^*(\lambda) = \operatorname*{argmin}_{\beta \in \mathbb{R}^p}(||(\tilde{Y}^* - \mathbf{X}_{\mathcal{A}}(\tilde{\beta}_{\mathcal{A}} + \beta_{0,\mathcal{A}})) - \mathbf{X}\beta||_2^2 + \lambda||\beta||_1) \quad (7)$$

and

$$\hat{\beta}_0^* := \hat{\beta}_0^*(\lambda) = \underset{\beta \in \mathbb{R}^p, \beta_j = 0, j \in \mathcal{A}}{\operatorname{argmin}} (||(\tilde{Y}^* - \mathbf{X}_{\mathcal{A}}(\tilde{\beta}_{\mathcal{A}} + \beta_{0,\mathcal{A}})) - \mathbf{X}\beta||_2^2 + \lambda||\beta||_1),$$

(8)

respectively. Then the bootstrap version of $T_0(s,t) = ||\hat{\beta} - \hat{\beta}_0||_{s,t}$ is given by

$$T_0^*(s,t) = ||\hat{\beta}^* - \hat{\beta}_0^*||_{s,t}.$$

Given a large number B of Monte-Carlo replicates of $T_0^*(s,t)$, denoted by, say, $T_0^{*,(1)}(s,t) < \cdots < T_0^{*,(B)}(s,t)$, when ordered, our bootstrap-based test of H_0 at significance level α has decision rule

Reject H_0 if and only if $T_0(s,t) > T_0^{*,(\lfloor B(1-\alpha) \rfloor)}$,

where $T_0^{*,(\lfloor B(1-\alpha) \rfloor)}$ is the Monte-Carlo approximation to the bootstrap estimator of the upper α-quantile of the null distribution of $T_0(s,t)$, and $\lfloor \cdot \rfloor$ is the floor function. We could also obtain a bootstrapped P-value by

$$B^{-1} \sum_{i=1}^{B} I\{T_0^{*,(i)}(s,t) > T_0(s,t)\},$$

where $I(\cdot)$ is the indicator function.

For the simpler hypotheses H_0: $\beta_j = 0$ versus H_1: $\beta_j \neq 0$ for any $j = 1, \ldots, p$, we need to construct a bootstrap version of the LOCO path statistic $T_j(s,t) = ||\hat{\beta} - \hat{\beta}^{(-j)}||_{s,t}$. The bootstrap versions of $\hat{\beta}$ and $\hat{\beta}^{(-j)}$, following (7) and (8), are

$$\hat{\beta}^* := \hat{\beta}^*(\lambda) = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}}(||(\tilde{Y}^* - \mathbf{X}_j \tilde{\beta}_j) - \mathbf{X}\beta||_2^2 + \lambda||\beta||_1)$$

and

$$\hat{\beta}^{*(-j)} := \hat{\beta}^{*(-j)}(\lambda) = \underset{\beta \in \mathbb{R}^p, \beta_j = 0}{\operatorname{argmin}} (||(\tilde{Y}^* - \mathbf{X}_j \tilde{\beta}_j) - \mathbf{X}\beta||_2^2 + \lambda||\beta||_1),$$

respectively, where $\mathbf{X}_j$ is column j of the matrix $\mathbf{X}$. Then the bootstrap version of $T_j(s,t)$ is given by

$$T_j^*(s,t) = ||\hat{\beta}^* - \hat{\beta}^{*(-j)}||_{s,t}.$$

Regarding the choice of the initial estimator $\tilde{\beta}$ of β, which is used only to obtain residuals suitable for resampling, we suggest, when $p \geq n$, the adaptive LASSO estimator

$$\hat{\beta}^{Ada} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}}(||Y - \mathbf{X}\beta||_2^2 + \gamma \sum_{j=1}^{p} \hat{w}_j |\beta_j|),$$

where the tuning parameter γ is selected via 10-fold cross validation and the weights $\hat{w}_1, \ldots, \hat{w}_p$ are given by

$$\hat{w}_j = 1/|\hat{\beta}_j^L|, \quad j = 1, \ldots, p,$$

where $\hat{\beta}_1^L, \ldots, \hat{\beta}_p^L$ are the LASSO estimates of $\beta_1, \ldots, \beta_p$ from (2) under the 10-fold cross validation choice of λ. This is the initial estimator we have used in our simulation studies, and it appears to work well. For the $p < n$ case the least-squares estimator could be used, though even in the low-dimensional case, we still recommend using the adaptive LASSO estimator when p is close to n.

3.2. Justification of the bootstrap for a simple case.

Finding the sampling distribution of $T_0(s, t)$ in general is a very hard problem which we do not attempt to solve. However, we do provide in this section an argument for why the bootstrap method described in the previous section will work in a simple case: the low-dimensional case, with $p < n$, with a design matrix having orthonormal columns. We focus on the null distribution of the test statistic $T_j(1, 1) = \|\hat{\beta} - \hat{\beta}^{(-j)}\|_{1,1}$ for testing H_0: $\beta_j = 0$ versus H_1: $\beta_j \neq 0$ for some $j \in \{1, \ldots, p\}$.

In low-dimension, if the design matrix $\mathbf{X}$ satisfies $\mathbf{X}^T\mathbf{X} = \mathbf{I}_p$, where $\mathbf{I}_n$ is the $n \times n$ identity matrix, the LASSO solution path $\hat{\beta}$ has entries given by

$$\hat{\beta}_k(\lambda) = S_\lambda(\hat{\beta}_k^{\mathrm{LS}}), \quad k = 1, \ldots, p,$$

where $\hat{\beta}^{\mathrm{LS}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^TY = \mathbf{X}^TY$ is the least-squares estimator of β and $S_\lambda(\cdot)$ is the soft-thresholding operator defined by

$$S_\lambda(x) = \begin{cases} x - \lambda, & x > \lambda \\ 0, & -\lambda < x < \lambda \\ x + \lambda, & x < -\lambda \end{cases}$$

for $\lambda \geq 0$. The solution path $\hat{\beta}^{(-j)}$ has entries given by

$$\hat{\beta}_k^{(-j)}(\lambda) = \begin{cases} 0 & k = j \\ S_\lambda(\hat{\beta}_k^{\mathrm{LS}}) & k \neq j \end{cases}$$

for $k = 1, \ldots, p$.

In this case, the LOCO path statistic $T_j(1, 1)$ is given by

$$T_j(1, 1) = \|\hat{\beta} - \hat{\beta}^{(-j)}\|_{1,1} = \sum_{k=1}^p \int_0^\infty |\hat{\beta}_k(\lambda) - \hat{\beta}_k^{(-j)}(\lambda)| d\lambda$$

$$= \int_0^{|\hat{\beta}_j^{\mathrm{LS}}|} (|\hat{\beta}_j^{\mathrm{LS}}| - \lambda) d\lambda = \frac{1}{2}|\hat{\beta}_j^{\mathrm{LS}}|^2.$$

So, our test statistic is merely a 1-to-1 mapping of the least-squares estimator. Hence, under H_0: $\beta_j = 0$,

$$nT_j(1,1) = \frac{n}{2}|\hat{\beta}_j^{\mathrm{LS}}|^2 \sim W\frac{\sigma^2}{2},$$

where $W \sim \chi_1^2$.

Now consider the bootstrap version $T_j^*(1,1)$ of $T_j(1,1)$ in the $p < n$ and orthonormal design case; we assume that the least-squares estimator is used as the initial estimator from which the residuals are obtained. Let $\hat{\beta}^{*,\mathrm{LS}} = \mathbf{X}^T \tilde{Y}^*$ be the bootstrap version of $\hat{\beta}^{\mathrm{LS}}$. Now, we can write the entries of

$$\hat{\beta}^*(\lambda) = \operatorname*{argmin}_{\beta \in \mathbb{R}^p}(||(\tilde{Y}^* - \mathbf{X}_j\hat{\beta}_j^{\mathrm{LS}}) - \mathbf{X}\beta||_2^2 + \lambda||\beta||_1)$$

as

$$\hat{\beta}_k^*(\lambda) = \begin{cases} S_\lambda(\hat{\beta}_k^{*,\mathrm{LS}} - \hat{\beta}_k^{\mathrm{LS}}), & k = j \\ S_\lambda(\hat{\beta}_k^{*,\mathrm{LS}}), & k \neq j \end{cases} \quad \text{for } k = 1, \ldots, p,$$

using the fact that

$$\mathbf{X}_k^T(\tilde{Y}^* - \mathbf{X}_j\hat{\beta}_j^{\mathrm{LS}}) = \begin{cases} \hat{\beta}_k^{*,\mathrm{LS}} - \hat{\beta}_k^{\mathrm{LS}}, & k = j \\ \hat{\beta}_k^{*,\mathrm{LS}} & k \neq j, \end{cases} \quad \text{for } k = 1, \ldots, p.$$

In addition, we can write the entries of

$$\hat{\beta}^{*(-j)}(\lambda) = \operatorname*{argmin}_{\beta \in \mathbb{R}^p, \beta_j = 0} (||(\tilde{Y}^* - \mathbf{X}_j\hat{\beta}_j^{\mathrm{LS}}) - \mathbf{X}\beta||_2^2 + \lambda||\beta||_1)$$

as

$$\hat{\beta}_k^{*(-j)}(\lambda) = \begin{cases} 0, & k = j \\ S_\lambda(\hat{\beta}_k^{*,\mathrm{LS}}), & k \neq j \end{cases} \quad \text{for } k = 1, \ldots, p.$$

So we have

$$T_j^*(1,1) = \|\hat{\beta}^* - \hat{\beta}^{*(-j)}\|_{1,1} = \sum_{k=1}^p \int_0^\infty |\hat{\beta}_k^{*(-j)}(\lambda) - \hat{\beta}_k^*(\lambda)|d\lambda$$

$$= \int_0^{|\hat{\beta}_k^{*,\mathrm{LS}} - \hat{\beta}_k^{\mathrm{LS}}|} (|\hat{\beta}_k^{*,\mathrm{LS}} - \hat{\beta}_k^{\mathrm{LS}}| - \lambda)d\lambda = \frac{1}{2}|\hat{\beta}_j^{*,\mathrm{LS}} - \hat{\beta}_j^{\mathrm{LS}}|^2.$$

It can be established that

$$\sup_{x\in\mathbb{R}} \left| P_* \left(\frac{n}{2}|\hat{\beta}_j^{*,\mathrm{LS}} - \hat{\beta}_j^{\mathrm{LS}}|^2 < x\right) - P\left(\frac{n}{2}|\hat{\beta}_j^{\mathrm{LS}} - \beta_j|^2 < x\right)\right| \xrightarrow{p} 0,$$

as $n \to \infty$, where P_* denotes probability conditional on the observed data [17]. This means our bootstrap works in the low-dimensional orthonormal design case. In the high-dimensional case, or even in the low-dimensional case without the assumption of an orthogonal design, (2) does not admit a simple solution, and in this setting the derivation

of the distribution of the test statistic would be very difficult. Our simulation studies, however, suggest that our bootstrap procedure can consistently estimate the null distributions of the test statistics in the non-orthogonal design and high-dimensional cases.

4. Simulation studies

We now study via simulation the effectiveness of the LOCO path statistic as a variable screening tool as well as the properties of our proposed LOCO-path-based tests of hypotheses which use the residual bootstrap to estimate the null distributions of the test statistics. An R package `LOCOpath` that implements all of our proposed methods is publicly available at `http://github.com/devcao/LOCOpath`. We first present the variable screening results.

4.1. **Variable screening.** To assess the performance of the LOCO-path-based variable screening procedure described in Section 2, we follow the simulation example in [9], generating data from the model

$$Y = \beta X_1 + \beta X_2 + \beta X_3 + \epsilon,$$

where $\epsilon \sim \mathcal{N}(0, 1)$, with a total of p predictors $X_1, \ldots, X_p$ in the model. The rows of the design matrix are generated as independent multivariate normal random vectors with covariance matrix $\Sigma = (\rho^{|i-j|})_{1 \leq i,j \leq p}$, where $\rho = 0, 0.1, 0.5$ and 0.9. Models with $\beta = 1, 2, 3$, $p = 100$, $n = 20$, and $p = 1000$, $n = 50$ are considered. We simulated 200 data sets for each model. To compare with SIS and ISIS, we utilized the R package `SIS` [19]. We simulated 200 data sets and for each model we calculate $T_j(1,1)$ and $T_j(2,2)$ for $j = 1, 2, \ldots, p$ and select the top $n-1$ covariates, selecting the same number of covariates with SIS and ISIS in order to make a fair comparison. For our method, we utilized the R package `lars` [12] with LASSO modification to calculate our test statistic.

In Table 1 we show the proportion of times that the true model is contained in the set of selected covariates for our method and for the SIS and ISIS variable screening methods. In most cases, the model selected by our LOCO-path-based method contains the true model with greater frequency than that of the SIS and ISIS methods. We note that our method achieves this without any need for selecting tuning parameters, whereas the ISIS methods involves iterated LASSO fits for which the strength of the sparsity penalty must be chosen.

4.2. **Study of power and size of LOCO path tests of hypotheses.**

Setting	β	$T_j(1,1)$	$T_j(2,2)$	SIS	ISIS
$p = 1000, n = 50, \Sigma = \mathbf{I}_p$	1	0.995	0.995	0.900	0.945
	2	1	1	0.945	1
	3	1	1	0.990	1
$p = 1000, n = 50, \Sigma = (0.1^{|i-j|})_{1 \le i,j \le p}$	1	0.990	0.990	0.960	0.960
	2	1	1	0.995	1
	3	1	1	0.990	1
$p = 1000, n = 50, \Sigma = (0.5^{|i-j|})_{1 \le i,j \le p}$	1	1	1	1	0.890
	2	1	1	1	1
	3	1	1	1	1
$p = 1000, n = 50, \Sigma = (0.9^{|i-j|})_{1 \le i,j \le p}$	1	0.980	0.975	1	0.535
	2	1	1	1	0.825
	3	1	1	1	0.965
$p = 100, n = 20, \Sigma = \mathbf{I}_p$	1	0.630	0.630	0.560	0.440
	2	0.915	0.920	0.700	0.860
	3	0.955	0.955	0.710	0.905
$p = 100, n = 20, \Sigma = (0.1^{|i-j|})_{1 \le i,j \le p}$	1	0.705	0.700	0.685	0.495
	2	0.960	0.965	0.810	0.890
	3	0.970	0.970	0.845	0.970
$p = 100, n = 20, \Sigma = (0.5^{|i-j|})_{1 \le i,j \le p}$	1	0.940	0.940	0.945	0.505
	2	1	1	0.990	0.940
	3	1	1	0.995	0.975
$p = 100, n = 20, \Sigma = (0.9^{|i-j|})_{1 \le i,j \le p}$	1	0.745	0.74	1	0.465
	2	0.995	0.995	1	0.635
	3	1	1	1	0.805

TABLE 1. Proportion of times SIS, ISIS and our method selected a set of covariates containing $\{X_1, X_2, X_3\}$.

4.2.1. *Test involving a single coefficient.* We first study the size and power of the LOCO path test for testing the hypotheses H_0: $\beta_j = 0$ versus H_1: $\beta_j \ne 0$ for some $j \in \{1, \ldots, p\}$, where the rejection region of the test is calibrated using the residual bootstrap procedure described in Section 3. We consider the test statistics $T_j(1,1)$, $T_j(2,2)$, and $T_j(\infty, \infty)$. In high-dimensional ($p \ge n$) settings, we compare the empirical size and power of our test based on these statistics with the test based on the desparsified LASSO estimator of [21]. We use the R package `hdi` [6] to obtain the P-value based on the desparsified LASSO estimator using default settings [7]. And we utilize the R package

`lars` [12] with lasso modification to implement our method. In low-dimensional $(p < n)$ settings, we compare the performance of our tests to that of the classical t-test.

We generate data according to the model

$$Y = \mathbf{X}\beta + \epsilon,$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I}_n)$ and consider three cases with $n = 100$, $p = 80$ and $p = 1000$. For $p = 1000$, we set $\beta = (\beta_1, \dots, \beta_p)^T$ such that $\beta_2 = \cdots = \beta_{10} = 1$, $\beta_{11} = \cdots = \beta_{1000} = 0$. For $p = 80$, we set $\beta_2 = \beta_3 = 1$, $\beta_4 = \cdots = \beta_{80} = 0$. To simulate the power curve, we take different values of $\beta_1 \in \{0/10, 1/10, \dots, 1\}$. Each row of $\mathbf{X}$ is generated independently from the multivariate normal distribution $\mathcal{N}(0, \Sigma)$, where we consider different choices of the $p \times p$ covariance matrix Σ. For each choice of Σ and for each value of $\beta_1 \in \{0/10, 1/10, \dots, 1\}$, we generate $N = 500$ data sets and with each data set we test H_0: $\beta_1 = 0$ versus H_1: $\beta_1 \neq 0$. For each data set, we draw $B = 500$ bootstrap samples to estimate the null distribution. We record the proportion of rejections of H_0 at the $\alpha = 0.05$ significance level.

The empirical size of the simulation for H_0: $\beta_1 = 0$ under $p = 1000$, is given in Table 2 under different choices of Σ. We also recorded the empirical size of the test based on the desparsified LASSO estimator. It is clear that our method nicely controlled the size under different choices of Σ and different quantities $T_1(1, 1)$, $T_1(2, 2)$ and $T_1(\infty, \infty)$. The desparsified LASSO does not control the size in many cases.

The empirical power curves of our test based on the LOCO path statistics $T_1(1, 1)$ and $T_1(\infty, \infty)$ as well as of the test based on the desparsified LASSO under settings $n = 1000$ and $p = 80$ over the values $\beta_1 \in \{0/10, 1/10, \dots, 1\}$ are depicted in Figures 3 and 4. For most cases, $T_1(1, 1)$ have the highest power, while $T_1(\infty, \infty)$ loses a lot of power under the correlated design. Under different designs, our method outperformed desparsified LASSO using quantity $T_1(1, 1)$. It is interesting to see that the desparsified LASSO appears to outperform our method under the design $\Sigma = (0.9^{|i-j|})_{1 \le i, j \le p}$. However, since its size is inflated in that case, we dismiss its power curve. Overall, our methods achieves comparable or higher power, with size well-controlled, compared to the desparsified LASSO method.

For the $p = 80$ case, we will compare our method to the classical t-test. From the power curve in Figure 4, it is clear that our method achieved considerably greater power than the t-test using both $T_1(1, 1)$ and $T_1(\infty, \infty)$, while controlling the size at the same time.
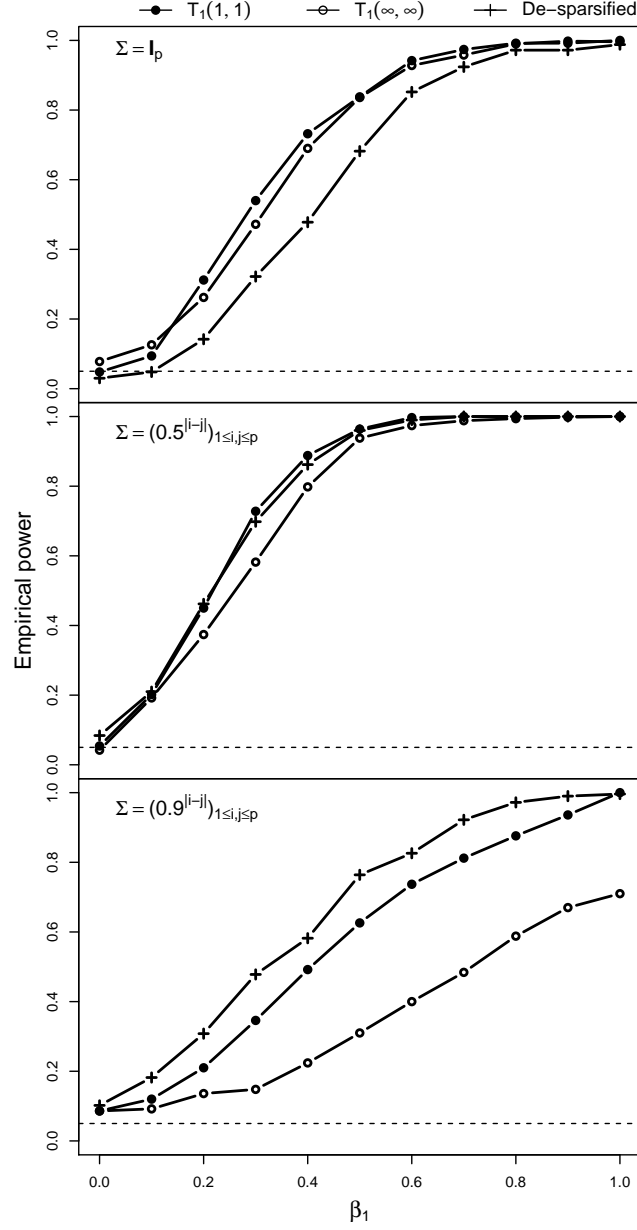
FIGURE 3. Empirical power for testing H_0: $\beta_1 = 0$ vs H_1: $\beta_1 \neq 0$ under different correlation design with $n = 100$, $p = 1000$ (from top to bottom: $\Sigma = \mathbf{I}_p$, $\Sigma = (0.5^{|i-j|})_{1 \leq i,j \leq p}$, and $\Sigma = (0.9^{|i-j|})_{1 \leq i,j \leq p}$).
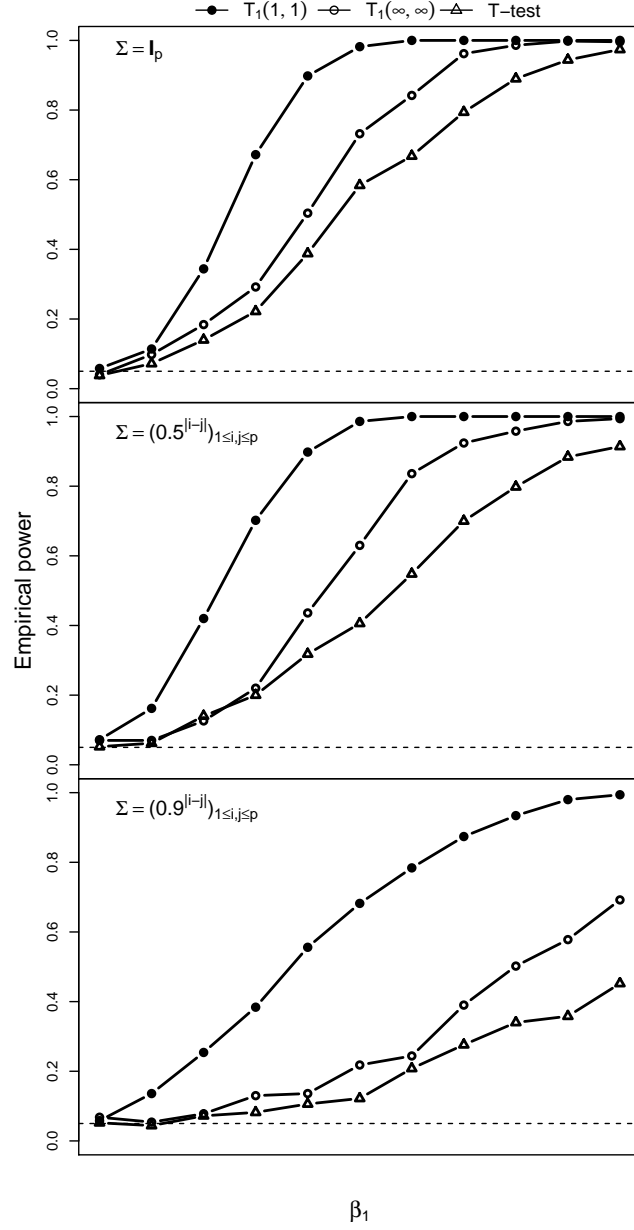
FIGURE 4. Empirical power for testing H_0: $\beta_1 = 0$ vs H_1: $\beta_1 \neq 0$ under different correlation design with $n = 100$, $p = 80$ (from top to bottom: $\Sigma = \mathbf{I}_p$, $\Sigma = (0.5^{|i-j|})_{1 \leq i,j \leq p}$, and $\Sigma = (0.9^{|i-j|})_{1 \leq i,j \leq p}$).

Design	Method	$\alpha = 0.20$	$\alpha = 0.10$	$\alpha = 0.05$	$\alpha = 0.01$
$\Sigma = \mathbf{I}_p$	$T_1(1,1)$	0.194	0.106	0.048	0.008
	$T_1(2,2)$	0.186	0.110	0.056	0.016
	$T_1(\infty,\infty)$	0.230	0.140	0.078	0.012
	Desparsified	0.138	0.058	0.030	0.010
$\Sigma = (0.5^{|i-j|})_{1 \leq i,j \leq p}$	$T_1(1,1)$	0.226	0.110	0.054	0.018
	$T_1(2,2)$	0.192	0.084	0.040	0.004
	$T_1(\infty,\infty)$	0.196	0.090	0.042	0.008
	Desparsified	0.222	0.138	0.084	0.020
$\Sigma = (0.9^{|i-j|})_{1 \leq i,j \leq p}$	$T_1(1,1)$	0.214	0.116	0.086	0.024
	$T_1(2,2)$	0.238	0.124	0.076	0.030
	$T_1(\infty,\infty)$	0.264	0.160	0.086	0.018
	Desparsified	0.274	0.162	0.102	0.054
$\Sigma = (0.5^{\mathbf{1}(i \neq j)})_{1 \leq i,j \leq p}$	$T_1(1,1)$	0.194	0.126	0.064	0.018
	$T_1(2,2)$	0.212	0.098	0.050	0.008
	$T_1(\infty,\infty)$	0.180	0.102	0.050	0.014
	Desparsified	0.126	0.048	0.028	0.004
$\Sigma = (0.8^{\mathbf{1}(i \neq j)})_{1 \leq i,j \leq p}$	$T_1(1,1)$	0.242	0.116	0.056	0.010
	$T_1(2,2)$	0.182	0.086	0.040	0.010
	$T_1(\infty,\infty)$	0.198	0.084	0.050	0.008
	Desparsified	0.070	0.022	0.010	0.002

TABLE 2. Empirical size of the test under different Σ with $n = 100$, $p = 1000$.

4.2.2. *Test involving multiple coefficients.* For the simultaneous test, we consider similar settings. For $p = 1000$, We will test

H_0: $\beta_1 = 1$, $\beta_{11} = 0$, $\beta_{12} = 0$ vs H_1: $\beta_1 \neq 1$ or $\beta_{11} \neq 0$ or $\beta_{12} \neq 0$.

and for $p = 80$, we will test

H_0: $\beta_1 = 1$, $\beta_4 = 0$, $\beta_5 = 0$ vs H_1: $\beta_1 \neq 1$ or $\beta_4 \neq 0$ or $\beta_5 \neq 0$.

We generate data according to the model

$$Y = \mathbf{X}\beta + \epsilon,$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I}_n)$ with $n = 100$ and $\beta = (\beta_1, \ldots, \beta_p)^T$. For $p = 1000$, we set $\beta_2 = \cdots = \beta_{10} = 1$, $\beta_{11} = \cdots = \beta_{1000} = 0$, and $\beta_1 \in \{1, 11/10, \ldots, 2\}$. For $p = 80$, we set $\beta_2 = \beta_3 = 1$. Other settings remain the same as those under which we tested H_0: $\beta_1 = 0$ versus H_1: $\beta_1 \neq 0$.

For the $p = 1000$ case, Figure 5 shows the power curves of the tests under different choices of Σ. The size is well controlled when H_0 is true, and $T_1(1,1)$ achieved higher power than $T_1(\infty, \infty)$ as the correlation increases..

For the $p = 80$ case, we will compare our method to the classical F-test. From the power curve in Figure 6, it is clear our method achieved considerably greater power than the F-test both $T_1(1,1)$ and $T_1(\infty, \infty)$, while controlling the size at the same time.

5. Real data analysis

To provide a concrete example, we consider a dataset about riboflavin (vitamin B2) production in Bacillus subtilis with 71 observations and 4088 variables [2] [7] [21]. The response variable measures the logarithm of the riboflavin production rate and the predictors are logarithm of the expression level of 4088 genes. We will model the data with a high-dimensional linear model and carry out variable screening and inferences with the LOCO path statistic.

We use $T(1,1)$ in this part and obtained bootstrap P-values for each gene after variable screening. We screened in 342 genes with $T_j(1,1) > 0$, $j = 1, \ldots, 4088$. Based on our bootstrapped P-values, our method found the following 9 significant genes at 0.05 significance level: ARGF_at, XHLA_at, XHLB_at, XTRA_at, YCKE_at, YEBC_at, YOAB_at, YXLD_at and YYBG_at. Using the P-values based on the desparsified LASSO results in 0 significant genes [7]. Figure 7 shows the variable importance for a small portion of genes. We will see only a few genes have large variable importance, while most genes have variable importance less than 1%.

Table 3 shows all variables with importance 1%, where YXLD_at and YOAB_at have the largest variable importance. Both genes are also tested significant using our bootstrap procedure.

6. Discussion

Our LOCO path statistic provides a new way to do variable screening and statistical inference in linear models. For variable screening, our method does not require the selection of tuning parameters and can achieve a greater probability of selecting a set of covariates that contains the true model than both SIS and ISIS. For statistical inference, our method provides reliable P-values in both high and low-dimensional settings. Overall, the proposed bootstrap method controls the size and in some cases achieves higher power than the desparsified LASSO of
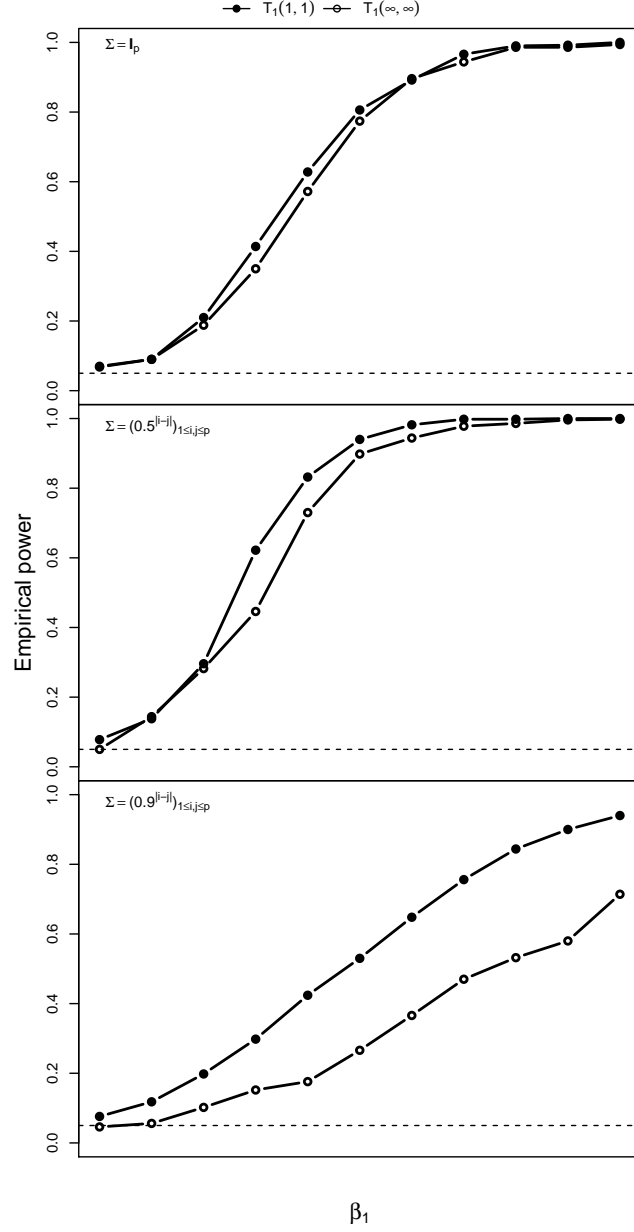
FIGURE 5. Multiple testing empirical power under different correlation design with $n = 100$, $p = 1000$ (from top to bottom: $\Sigma = \mathbf{I}_p$, $\Sigma = (0.5^{|i-j|})_{1 \leq i,j \leq p}$, and $\Sigma = (0.9^{|i-j|})_{1 \leq i,j \leq p}$).
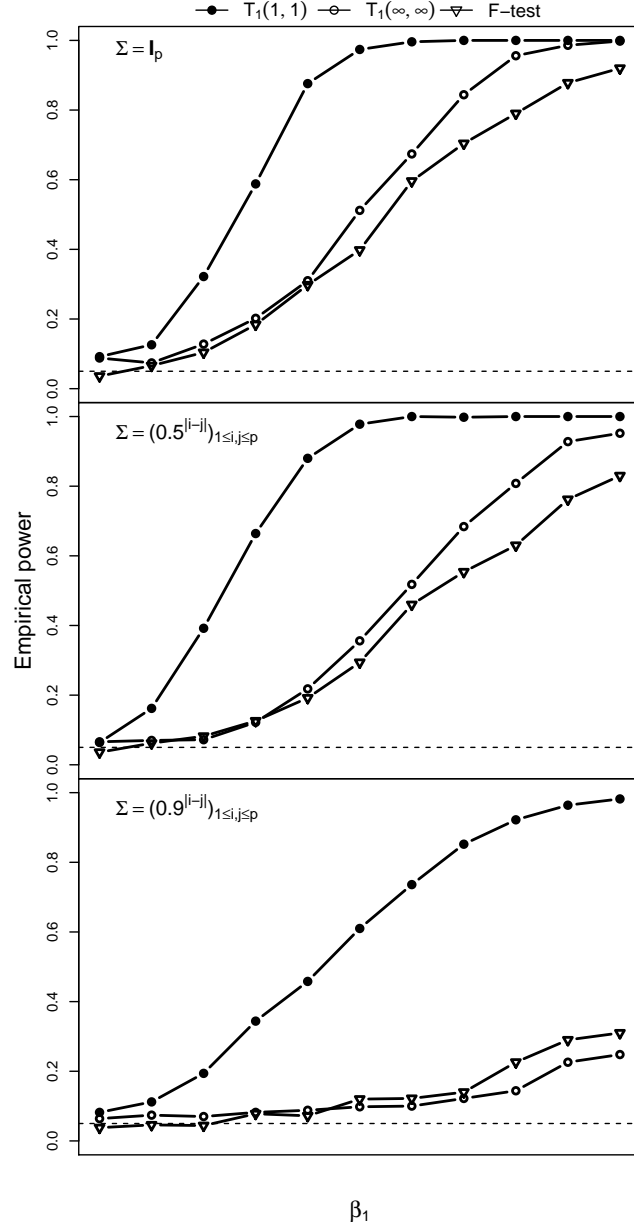
FIGURE 6. Multiple testing empirical power under different correlation design with $n = 100$, $p = 80$ (from top to bottom: $\Sigma = \mathbf{I}_p$, $\Sigma = (0.5^{|i-j|})_{1 \leq i,j \leq p}$, and $\Sigma = (0.9^{|i-j|})_{1 \leq i,j \leq p}$).
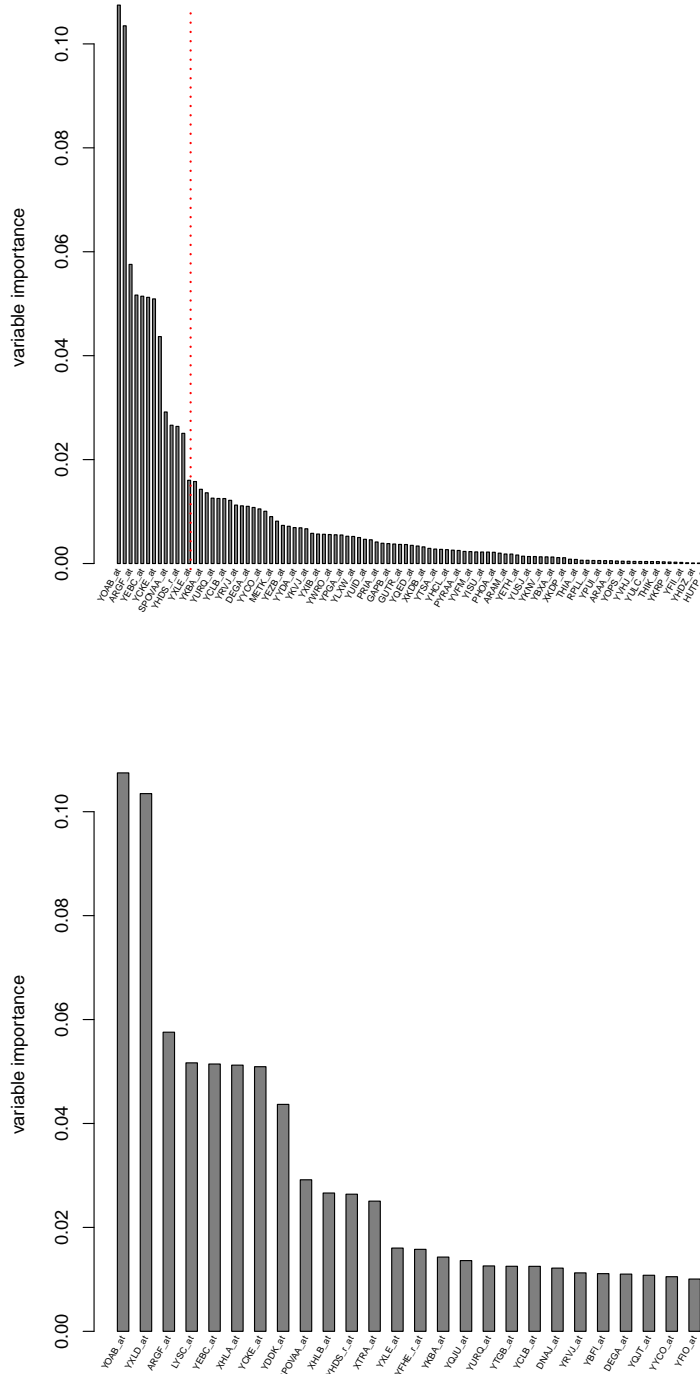
FIGURE 7. Top: The first 100 most important genes. The vertical dotted line marks the variable importance at 1%. Bottom: All genes with variable importance > 1%.

Genes	Importance	P-value
YOAB_at	10.7%	0.0084
YXLD_at	10.3%	0.0084
ARGF_at	5.8%	0.0168
LYSC_at	5.2%	0.0924
YEBC_at	5.2%	0.0616
XHLA_at	5.1%	0.0140
YCKE_at	5.1%	0.0084
YDDK_at	4.4%	0.0560
SPOVAA_at	2.9%	0.1482
XHLB_at	2.7%	0.0194

TABLE 3. The first 10 most important genes.

[21]. Moreover, our method can be used to test hypothesis simultaneously involving multiple coefficients. We believe the LOCO path idea can be readily extended to other settings.

Consider the regularization optimization problem

$$\hat{\beta} = \hat{\beta}(\lambda) := \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \, L(Y, \mathbf{X}\beta) + \lambda J(\beta), \tag{9}$$

where $L(\cdot)$ is a pre-defined loss function, $\lambda > 0$ is a tuning parameter which controls the level of regularization, and $J(\cdot)$ is a penalty function on β. The solution path $\hat{\beta}(\lambda)$ could be viewed as a 1-to-p mapping $\lambda \mapsto \hat{\beta}(\lambda)$ taking values in $(0, \infty)$ and returning values in $\mathbb{R}^p$. Since our measure of feature importance and variable screening procedure relies on the solution path only, we can easily adapt our method to (9), which includes logistic regression, Poisson regression and Cox models. Appropriate bootstrap methods for calibrating hypothesis tests would have to be worked out under each setting, which we leave to future work.

Supplementary Material

Supplementary material related to this article can be found in our submission.

References

[1] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[2] Peter Bühlmann, Markus Kalisch, and Lukas Meier. High-dimensional statistics with a view toward applications in biology. *Computational Statistics*, 29:407–430, 2014.

[3] Arindam Chatterjee, Soumendra N Lahiri, et al. Rates of convergence of the adaptive lasso estimators to the oracle distribution and higher order refinements by the bootstrap. *The Annals of Statistics*, 41(3):1232–1259, 2013.

[4] Arindam Chatterjee and Soumendra Nath Lahiri. Bootstrapping lasso estimators. *Journal of the American Statistical Association*, 106(494):608–625, 2011.

[5] Debraj Das, Karl Gregory, SN Lahiri, et al. Perturbation bootstrap in adaptive lasso. *The Annals of Statistics*, 47(4):2080–2116, 2019.

[6] Ruben Dezeure, Peter Bühlmann, Lukas Meier, and Nicolai Meinshausen. High-dimensional inference: Confidence intervals, p-values and R-software hdi. *Statistical Science*, 30(4):533–558, 2015.

[7] Ruben Dezeure, Peter Bühlmann, Lukas Meier, Nicolai Meinshausen, et al. High-dimensional inference: Confidence intervals, *p*-values and r-software hdi. *Statistical Science*, 30(4):533–558, 2015.

[8] Jianqing Fan, Yang Feng, and Rui Song. Nonparametric independence screening in sparse ultra-high-dimensional additive models. *Journal of the American Statistical Association*, 106(494):544–557, 2011.

[9] Jianqing Fan and Jinchi Lv. Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5):849–911, 2008.

[10] Jianqing Fan, Rui Song, et al. Sure independence screening in generalized linear models with np-dimensionality. *The Annals of Statistics*, 38(6):3567–3604, 2010.

[11] Aaron Fisher, Cynthia Rudin, and Francesca Dominici. All models are wrong but many are useful: Variable importance for black-box, proprietary, or misspecified prediction models, using model class reliance. *arXiv preprint arXiv:1801.01489*, 2018.

[12] Trevor Hastie and Brad Efron. *lars: Least Angle Regression, Lasso and Forward Stagewise*, 2013. R package version 1.2.

[13] Adel Javanmard and Andrea Montanari. Confidence intervals and hypothesis testing for high-dimensional regression. *The Journal of Machine Learning Research*, 15(1):2869–2909, 2014.

[14] Tracy Ke, Jiashun Jin, and Jianqing Fan. Covariance assisted screening and estimation. *Annals of Statistics*, 42(6):2202–2242, 2014.

[15] Jing Lei, Max GSell, Alessandro Rinaldo, Ryan J Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111, 2018.

[16] Richard Lockhart, Jonathan Taylor, Ryan J Tibshirani, and Robert Tibshirani. A significance test for the lasso. *Annals of Statistics*, 42(2):413–468, 2014.

[17] Enno Mammen. *When does bootstrap work?: asymptotic results and simulations*, volume 77. Springer Science & Business Media, 2012.

[18] Nicolai Meinshausen, Lukas Meier, and Peter Bühlmann. P-values for high-dimensional regression. *Journal of the American Statistical Association*, 104(488):1671–1681, 2009.

[19] Diego Franco Saldana and Yang Feng. SIS: An R package for sure independence screening in ultrahigh-dimensional statistical models. *Journal of Statistical Software*, 83(2):1–25, 2018.

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

[21] Sara Van de Geer, Peter Bühlmann, Yaacov Ritov, Ruben Dezeure, et al. On asymptotically optimal confidence regions and tests for high-dimensional models. *The Annals of Statistics*, 42(3):1166–1202, 2014.

[22] Larry Wasserman and Kathryn Roeder. High dimensional variable selection. *Annals of Statistics*, 37(5A):2178–2201, 2009.

[23] Cun-Hui Zhang and Stephanie S Zhang. Confidence intervals for low dimensional parameters in high dimensional linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(1):217–242, 2014.

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