Journal of Machine Learning Research 23 (2022) 1-37 Submitted 5/21; Revised 4/22; Published 11/22

More Powerful Conditional Selective Inference for Generalized Lasso by Parametric Programming

Vo Nguyen Le Duy DUY.MLLAB.NIT@GMAIL.COM
Nagoya Institute of Technology and RIKEN, Nagoya, Aichi 466-8555, Japan

Ichiro Takeuchi[*] ICHIRO.TAKEUCHI@MAE.NAGOYA-U.AC.JP
Nagoya University and RIKEN, Nagoya, Aichi 464-8601, Japan

Editor: Samuel Kaski

Abstract

Conditional selective inference (SI) has been studied intensively as a new statistical inference framework for data-driven hypotheses. The basic concept of conditional SI is to make the inference conditional on the selection event, which enables an exact and valid statistical inference to be conducted even when the hypothesis is selected based on the data. Conditional SI has mainly been studied in the context of model selection, such as vanilla lasso or generalized lasso. The main limitation of existing approaches is the low statistical power owing to over-conditioning, which is required for computational tractability. In this study, we propose a more powerful and general conditional SI method for a class of problems that can be converted into quadratic parametric programming, which includes generalized lasso. The key concept is to compute the continuum path of the optimal solution in the direction of the selected test statistic and to identify the subset of the data space that corresponds to the model selection event by following the solution path. The proposed parametric programming-based method not only avoids the aforementioned major drawback of over-conditioning, but also improves the performance and practicality of SI in various respects. We conducted several experiments to demonstrate the effectiveness and efficiency of our proposed method.

Keywords: Generalized Lasso, Model Selection, Selective Inference, Parametric Programming, Quadratic Programming

1. Introduction

As machine learning (ML) is applied to solve numerous practical problems, the quantification of the reliability of data-driven knowledge obtained by ML algorithms is becoming increasingly important. Among the various potential approaches for reliable ML, *conditional selective inference (SI)* has been recognized as a new promising method for assessing the statistical reliability of data-driven hypotheses that are selected by ML algorithms. The main concept of conditional SI is to make the inference for a data-driven hypothesis *conditional on the selection event* that the hypothesis is selected, which enables an *exact* and a *valid* inference to be conducted on the selected hypothesis. In the conditional SI framework, the statistical significance and reliability of data-driven selected hypotheses are quantified

[*] Corresponding author

by the so-called *selective p-values* and *selective confidence intervals*, which have proper false positive rates and coverage guarantees, respectively.

Lee et al. (2016) first introduced conditional SI as a statistical inference tool for the features selected by lasso (Tibshirani, 1996). Subsequently, Hyun et al. (2018a) studied SI for inference on the selected model using generalized lasso (Tibshirani and Taylor, 2011). Their basic concept was to characterize the selection event using a polytope (a set of linear inequalities) in the sample space. In general, we refer to such methods as *polytope-based SI* approaches. The practical computational methods developed by these authors can be used when the selection event can be characterized by a single polytope.

However, the application scope of such polytope-based SI is limited because it can only be used when the characterization of all relevant selection events is represented by a polytope. Therefore, in most existing polytope-based SI studies, additional conditioning is required for the selection event to be characterized as a polytope. For example, in the case of lasso (Lee et al., 2016), the set of selected features as well as their signs require conditioning. Similarly, in the case of generalized lasso (Hyun et al., 2018a), additional conditioning on the signs as well as on the history (sequential order) whereby the selected elements enter the selected model is required. Such additional conditioning results in low statistical power, which is widely recognized as a major drawback of polytope-based SI studies (Fithian et al., 2014).

Contributions. The contributions of this study are as follows:

- We go beyond the scope of polytope-based SI and propose a new SI approach based on *parametric programming (PP)*. We name the proposed method *PP-based SI*. The basic concept of PP-based SI is to compute the continuum path of the optimal solutions in the direction of the selected test statistic using PP, which is subsequently used to identify the exact sampling distribution of the test statistic with the minimum amount of conditioning. Therefore, PP-based SI can fundamentally resolve the over-conditioning problem, which is a major concern in polytope-based SI, thereby achieving high statistical power.

- We derive the proposed PP-based SI for a generic class of conditional SI problems that can be represented as parametric quadratic programs (QPs). We demonstrate that the conditional SI formulations for many practical problems, including generalized lasso, elastic net, non-negative least squares, and Huber regression with the ℓ_1 penalty, belong to this class, which means that the proposed PP-based SI can be used extensively.

- Furthermore, we discuss how the advantages of PP can be exploited for the effective performance of conditional SI in various data analysis tasks. As an example, using PP, we demonstrate that conditional SI can be conducted with minimal conditioning for regularization parameter selection by cross-validation (CV), the selection event of which is too complicated to be characterized as a single polytope, but can be fully characterized using PP[1].

[1] Loftus (2015) considered conditional SI for CV-based regularization parameter selection, but it was highly over-conditioned with additional events. We demonstrate that our PP-based SI is more powerful than this approach.

- We conducted intensive experiments on both synthetic and real-world datasets, by means of which we presented evidence that our proposed method can successfully control the false positive rate, has higher statistical power than polytope-based SI, and provides superior results in practical applications. Our code is available at

  https://github.com/vonguyenleduy/parametric_generalized_lasso_selective_inference.

A preliminary short version of this work was presented at the AI & Statistics (AISTATS2021) conference (Le Duy and Takeuchi, 2021). In the conference version, we only studied a specific case of vanilla lasso. In this study, we extended the basic concept of PP-based SI to a more general class of problems that can be formulated as parametric QPs, which includes vanilla lasso as a special case. Moreover, we extended the proposed method to various aspects and conducted intensive additional experiments to demonstrate the applicability of the generalized PP-based SI to a wider class of problems and settings.

Related works. In traditional statistical inference, it is assumed that the hypothesis is fixed in advance. That is, the hypothesis on which we wish to conduct inference is determined prior to observing the dataset. Therefore, if traditional statistical inference methods are applied to data-driven hypotheses, the inferential results will no longer be valid. This problem has been discussed extensively in the context of testing the significance of the features selected by a feature selection method, such as lasso or stepwise feature selection. Several approaches have been proposed to address this problem (Benjamini and Yekutieli, 2005; Leeb and Pötscher, 2005, 2006; Benjamini et al., 2009; Pötscher and Schneider, 2010; Berk et al., 2013; Lockhart et al., 2014; Taylor et al., 2014).

In recent years, Lee et al. (2016) proposed a practical SI framework to perform exact (non-asymptotic) inference for a set of features selected by lasso. In their work, the authors revealed that the selection event can be characterized as a polytope by conditioning on a set of selected features and their signs. Furthermore, Hyun et al. (2018a) demonstrated that polytope-based SI is applicable to generalized lasso by performing additional conditioning on the signs as well as the history (sequential order) whereby the selected elements entered the selected model. Following the seminal work of (Lee et al., 2016), conditional inference-based SI has been actively studied and applied to various problems (Fithian et al., 2015; Choi et al., 2017; Tian and Taylor, 2018; Chen and Bien, 2019; Hyun et al., 2018b; Loftus and Taylor, 2014; Loftus, 2015; Panigrahi et al., 2016; Tibshirani et al., 2016; Yang et al., 2016; Suzumura et al., 2017; Tanizaki et al., 2020; Duy et al., 2020, 2022; Sugiyama et al., 2020; Tsukurimichi et al., 2021; Duy and Takeuchi, 2022a,b).

It is desirable to conduct more powerful inference by conditioning on as little information as possible in conditional SI (Fithian et al., 2014). However, in polytope-based SI, an excessive amount of over-conditioning is required to represent the selection event using a single polytope. The authors of Lee et al. (2016) already mentioned the problem of over-conditioning and discussed the solution for removing the additional conditioning by enumerating all possible combinations of signs and taking the union over the resulting polyhedra. Unfortunately, such an enumeration for an exponentially increasing number of sign combinations is only feasible when the number of selected features is small. Loftus and Taylor (2014) extended polytope-based SI to cases in which the selection event is characterized by quadratic inequalities. Although we do not discuss quadratic inequality-based SI further, as it suffers from a similar over-conditioning issue, our proposed PP-based SI can also be applied to resolve the issue for this class of conditional SIs.

Our work was motivated by Liu et al. (2018), in which the authors proposed solutions to overcome the over-conditioning issue of polytope-based SI for vanilla lasso in certain special settings. In one of the settings, inference on the full model parameters in which conditional SI can be performed with minimal conditioning was studied, because a full model parameter is not dependent on other parameters. Moreover, our work was motivated by a discussion in the paper where the authors noted that multiple lasso fitting at a sequence of grid points may aid in alleviating over-conditioning. However, the authors did not suggest any practical computational methods to realize this concept. Furthermore, the conditional sampling distribution that is evaluated at a finite number of grid points only provides an approximation of the exact distribution, which means that the theoretical validity of the conditional SI is no longer guaranteed. Our proposed PP-based SI can be interpreted as a means of solving lasso at *infinitely* many grid points, which completely resolves these challenging problems. As another direction to resolve over-conditioning, Tian and Taylor (2018) and Terada and Shimodaira (2019) proposed methods using randomization. A drawback of these randomization-based approaches (including the simple data-splitting approach) is that further randomness is added in both the feature selection and inference stages.

PP has long been studied in the optimization field to solve a family of optimization problems that are parameterized by a scalar parameter (Ritter, 1984; Allgower and George, 1993; Gal, 1995; Best, 1996). Moreover, PP has been used in the context of the *regularization path* in ML (Osborne et al., 2000; Efron and Tibshirani, 2004; Hastie et al., 2004; Rosset and Zhu, 2007; Bach et al., 2006; Rosset and Zhu, 2007; Tsuda, 2007; Lee and Scott, 2007; Takeuchi et al., 2009; Takeuchi and Sugiyama, 2011; Karasuyama and Takeuchi, 2010; Hocking et al., 2011; Karasuyama et al., 2012; Ogawa et al., 2013; Takeuchi et al., 2013). The regularization path is a method of tracking the manner in which the optimal solution changes when the regularization parameter changes, which is useful for efficient model selection. Our main idea was to employ PP to track how the optimal solution and selected features change when the training dataset changes in the direction of the selected test statistic, which enables the exact sampling distribution of the test statistic that is conditional on the selection event to be identified. The power of the conditional SI introduced in the pioneering work of Lee et al. (2016) can be optimized using the proposed approach, which is applicable to conditional SI for a wide class of problems including generalized lasso.

We would like to note that additionally conditioning on the signs in Lee et al. (2016) and Hyun et al. (2018a) also have some utilities. For instance, in Hyun et al. (2018a), the signs are useful for one-sided tests — the "segment tests" explicitly use the signs. The usage of the signs can buy back some power compared to two-sided tests. This point is also discussed in Sec. 4.5 of Tibshirani et al. (2016). Also in Sec. 4.6 of Hyun et al. (2018a), the authors discussed a post-processing in which using the extra information (e.g. signs) can improve the power of the inference. Besides, in the cases that additionally conditioning on signs does not cause a huge lost in terms of statistical power (e.g, sample size is large or the signal is high), it can help gain efficiency in terms of computation.

In the literature of conditional SI, there were discussions about "selected model" and "saturated model". In the case of selected model, the tests can be more powerful than the ones in saturated model, but the inference are only valid under more restrictive assumptions on the underlying model (Fithian et al., 2014). In this paper, we focus on the saturated model which is mainly considered in the seminal paper of Lee et al. (2016).

2. Problem Statement

To formulate the problem, we consider a random response vector

$$\boldsymbol{Y} = (Y_1, ..., Y_n)^\top \sim \mathbb{N}(\boldsymbol{\mu}, \Sigma), \tag{1}$$

where n is the number of instances, $\boldsymbol{\mu}$ is an unknown vector, and $\Sigma \in \mathbb{R}^{n \times n}$ is a covariance matrix that is known or estimable from independent data. The goal is *statistical* quantification of the significance of the data-driven hypotheses that are obtained by applying the generalized lasso estimator to the response vector.

Generalized lasso and its selection event. We consider a linear regression model with p features $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_p \in \mathbb{R}^n$ and denote the feature matrix as $X \in \mathbb{R}^{n \times p}$, in which the features are considered as non-random. We do not make any assumption about the relationship between the $\boldsymbol{\mu}$ and p features $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_p \in \mathbb{R}^n$, but consider the case in which a linear model with generalized lasso regularization is employed to model the relationship between X and a random response vector $\boldsymbol{Y}$. Given an observed response vector $\boldsymbol{y}^{\rm obs} \in \mathbb{R}^n$ that is sampled from model (1), the generalized lasso optimization problem is expressed as

$$\hat{\boldsymbol{\beta}} = \mathop{\arg\min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2} \|\boldsymbol{y}^{\rm obs} - X\boldsymbol{\beta}\|_2^2 + \lambda \|D\boldsymbol{\beta}\|_1, \tag{2}$$

where $D \in \mathbb{R}^{m \times p}$ is a penalty matrix and $\lambda \geq 0$ is a regularization parameter. The matrix D and its number of rows are predetermined by the user to produce the desired structures in the solution $\hat{\boldsymbol{\beta}}$ in (2). Examples of matrix D are presented in Examples 1, 2, and 3.

As the optimization in (2) produces the sparsity of $D\hat{\boldsymbol{\beta}}$, we define a set of non-zero components (the active set) as follows:

$$\mathcal{M}_{\rm obs} = \mathcal{A}(\boldsymbol{y}^{\rm obs}) = \{j : (D\hat{\boldsymbol{\beta}})_j \neq 0\}, \quad j \in [m],$$

where $\mathcal{A} : \boldsymbol{Y} \mapsto \mathcal{M}$ indicates the algorithm that maps a response vector $\boldsymbol{Y}$ to a set of non-zero components $\mathcal{M}$. Thereafter, we define the selection event in which the active set for a random response vector $\boldsymbol{Y}$ is the same as the observed response vector $\boldsymbol{y}^{\rm obs}$:

$$\left\{ \mathcal{A}(\boldsymbol{Y}) = \mathcal{A}(\boldsymbol{y}^{\rm obs}) \right\}. \tag{3}$$

Statistical inference. Let $\boldsymbol{\eta}_j^\top \boldsymbol{Y}$ be a linear contrast that indicates the test statistic that we wish to consider, where $\boldsymbol{\eta}_j$ is defined depending on the problem and the $j^{\rm th}$ selected component in the observed active set.

Example 1 In the case of testing the features selected by vanilla lasso (Tibshirani, 1996), $D = I_p \in \mathbb{R}^{p \times p}$, which is the identity matrix. The test statistic $\boldsymbol{\eta}_j^\top \boldsymbol{Y} = \hat{\beta}_j$ represents the coefficient of the $j^{\rm th}$ selected feature (Lee et al., 2016), where $\boldsymbol{\eta}_j$ is defined as

$$\boldsymbol{\eta}_j = X_{\mathcal{M}_{\rm obs}} \left(X_{\mathcal{M}_{\rm obs}}^\top X_{\mathcal{M}_{\rm obs}} \right)^{-1} \boldsymbol{e}_j, \tag{4}$$

in which $\boldsymbol{e}_j \in \mathbb{R}^{|\mathcal{M}_{\rm obs}|}$ is a basis vector with 1 at the $j^{\rm th}$ position. This form of test statistic can also be applied to test the features that are selected by other regression methods, such as elastic net (Zou and Hastie, 2005), Huber regression (Huber, 1992), or non-negative least squares.

Example 2 In the context of *changepoint (CP)* detection using fused lasso, the matrix $D \in \mathbb{R}^{(p-1)\times p}$ is expressed as

$$D = \begin{pmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ & & & \cdots & & \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{pmatrix}.$$

The test statistic $\boldsymbol{\eta}_j^\top \boldsymbol{Y}$ represents the difference in the sample mean between the left and right segments of the CP at the $j^{\rm th}$ position, which was also used in Hyun et al. (2018a). In this case, $\boldsymbol{\eta}_j$ is defined as

$$\boldsymbol{\eta}_j = \frac{1}{j - j_{\rm prev}} \mathbf{1}^n_{j_{\rm prev}+1:j} - \frac{1}{j_{\rm next} - j} \mathbf{1}^n_{j+1:j_{\rm next}}, \tag{5}$$

where $j_{\rm prev} \in \mathcal{M}_{\rm obs}$ and $j_{\rm next} \in \mathcal{M}_{\rm obs}$ are the CP positions before and after the selected CP at position j, respectively, and $\mathbf{1}^n_{s:e} \in \mathbb{R}^n$ is a vector in which the elements from positions s to e are set to 1, and 0 otherwise.

Example 3 In trend filtering, the aim is to test whether a change occurs in the trend at position $j \in \mathcal{M}_{\rm obs}$. We define $\boldsymbol{\eta}_j$ as follows:

$$\boldsymbol{\eta}_j = \boldsymbol{e}_{j-1} - 2\boldsymbol{e}_j + \boldsymbol{e}_{j+1},$$

where $\boldsymbol{e}_j \in \mathbb{R}^n$. The test statistic $\boldsymbol{\eta}_j^\top \boldsymbol{Y}$ indicates that we wish to test whether the points at positions j, $j-1$, and $j+1$ lie on the same line statistically. In this case, the matrix $D \in \mathbb{R}^{(p-2)\times p}$ is expressed as

$$D = \begin{pmatrix} -1 & 2 & -1 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 2 & \cdots & 0 & 0 & 0 \\ & & & \cdots & & & \\ 0 & 0 & 0 & \cdots & -1 & 2 & -1 \end{pmatrix}.$$

For the inference, we consider the following *null hypothesis* and *alternative hypothesis*:

$$\mathrm{H}_{0,j} : \boldsymbol{\eta}_j^\top \boldsymbol{\mu} = 0 \quad \text{vs.} \quad \mathrm{H}_{1,j} : \boldsymbol{\eta}_j^\top \boldsymbol{\mu} \neq 0. \tag{6}$$

Conditional SI. Suppose that the hypotheses in (6) are fixed; that is, non-random. Thus, the *naive* (two-sided) p-value in the classical z-test is obtained by

$$P_j^{\rm naive} = \mathbb{P}_{\mathrm{H}_{0,j}} \left(|\boldsymbol{\eta}_j^\top \boldsymbol{Y}| \geq |\boldsymbol{\eta}_j^\top \boldsymbol{y}^{\rm obs}| \right). \tag{7}$$

However, as the hypotheses in (6) are not fixed in advance, the naive p-value is not *valid* in the sense that, if we reject $\mathrm{H}_{0,j}$ with a significance level α (e.g., $\alpha = 0.05$), the false positive rate (type-I error) cannot be controlled at the level α. This is because the hypotheses in (6) are *selected* by the data and *selection bias* exists.

It is necessary to remove the information that has been used for the initial hypothesis generation process to correct the selection bias. This can be achieved by considering the sampling distribution of the test statistic $\boldsymbol{\eta}_j^\top \boldsymbol{Y}$ that is conditional on the selection event; that is,

$$\boldsymbol{\eta}_j^\top \boldsymbol{Y} \mid \left\{ \mathcal{A}(\boldsymbol{Y}) = \mathcal{A}(\boldsymbol{y}^{\rm obs}), \boldsymbol{q}(\boldsymbol{Y}) = \boldsymbol{q}(\boldsymbol{y}^{\rm obs}) \right\}, \tag{8}$$

where $\boldsymbol{q}(\boldsymbol{Y}) = (I_n - \boldsymbol{c}\boldsymbol{\eta}_j^\top)\boldsymbol{Y}$ with $\boldsymbol{c} = \Sigma\boldsymbol{\eta}_j(\boldsymbol{\eta}_j^\top \Sigma \boldsymbol{\eta}_j)^{-1}$ is the nuisance component. We note that $\boldsymbol{q}(\boldsymbol{Y})$ is independent of the test statistic $\boldsymbol{\eta}_j^\top \boldsymbol{Y}$ if $\boldsymbol{\eta}_j$ is fixed. In the context of conditional SI, $\boldsymbol{\eta}_j$ is considered to be fixed in the sense that we only consider a specific observed active set rather than all possible active sets. The second condition $\boldsymbol{q}(\boldsymbol{Y}) = \boldsymbol{q}(\boldsymbol{y}^{\rm obs})$ indicates that the nuisance component for a random vector $\boldsymbol{Y}$ is the same as that for $\boldsymbol{y}^{\rm obs}$ [2].

Once the selection event has been identified, the pivotal quantity can be computed:

$$F^{\mathcal{Z}}_{\boldsymbol{\eta}_j^\top \boldsymbol{\mu}, \boldsymbol{\eta}_j^\top \Sigma \boldsymbol{\eta}_j}(\boldsymbol{\eta}_j^\top \boldsymbol{Y}) \mid \left\{ \mathcal{A}(\boldsymbol{Y}) = \mathcal{A}(\boldsymbol{y}^{\rm obs}), \boldsymbol{q}(\boldsymbol{Y}) = \boldsymbol{q}(\boldsymbol{y}^{\rm obs}) \right\}, \tag{9}$$

which is the c.d.f. of the truncated normal distribution with a mean $\boldsymbol{\eta}_j^\top \boldsymbol{\mu}$, variance $\boldsymbol{\eta}_j^\top \Sigma \boldsymbol{\eta}_j$, and truncation region $\mathcal{Z}$, which is calculated based on the selection event. Based on the pivotal quantity, the *selective type-I error* or *selective p-value* (Fithian et al., 2014) can be considered in the following form:

$$P_j^{\rm selective} = 2 \ \min\{\pi_j, 1 - \pi_j\}, \tag{10}$$

where $\pi_j = 1 - F^{\mathcal{Z}}_{0, \boldsymbol{\eta}_j^\top \Sigma \boldsymbol{\eta}_j}(\boldsymbol{\eta}_j^\top \boldsymbol{Y})$, which is *valid* in the sense that

$$\text{Prob}_{\text{H}_{0,j}} \left(P_j^{\rm selective} < \alpha \right) = \alpha, \forall \alpha \in [0, 1].$$

Furthermore, to obtain a confidence level of $1 - \alpha$ for any $\alpha \in [0, 1]$, by inverting the pivotal quantity in Equation (9), we can determine the smallest and largest values of $\boldsymbol{\eta}_j^\top \boldsymbol{\mu}$ such that the value of the pivotal quantity remains in the interval $\left[\frac{\alpha}{2}, 1 - \frac{\alpha}{2}\right]$ (Lee et al., 2016).

Challenge in conditional data space characterization. The main difficulty in the above conditional SI is that the characterization of the minimal conditional data space

$$\left\{ \mathcal{A}(\boldsymbol{Y}) = \mathcal{A}(\boldsymbol{y}^{\rm obs}), \boldsymbol{q}(\boldsymbol{Y}) = \boldsymbol{q}(\boldsymbol{y}^{\rm obs}) \right\}$$

in Equation (8) is intractable. To overcome this issue, Hyun et al. (2018a) considered the inference to be conditional not only on the active set, but also on the signs and history (order) whereby the elements of $D\hat{\boldsymbol{\beta}}$ entered the active set. Unfortunately, such additional conditioning on the signs and history leads to low statistical power owing to over-conditioning [3].

[2] $\boldsymbol{q}(\boldsymbol{Y})$ corresponds to the component $\boldsymbol{z}$ in the seminal paper (see Lee et al. (2016), Section 5, Eq. 5.2 and Theorem 5.2).

[3] This over-conditioning corresponds to the additional conditioning on the signs of the selected features in the seminal conditional SI study of vanilla lasso (Lee et al., 2016).

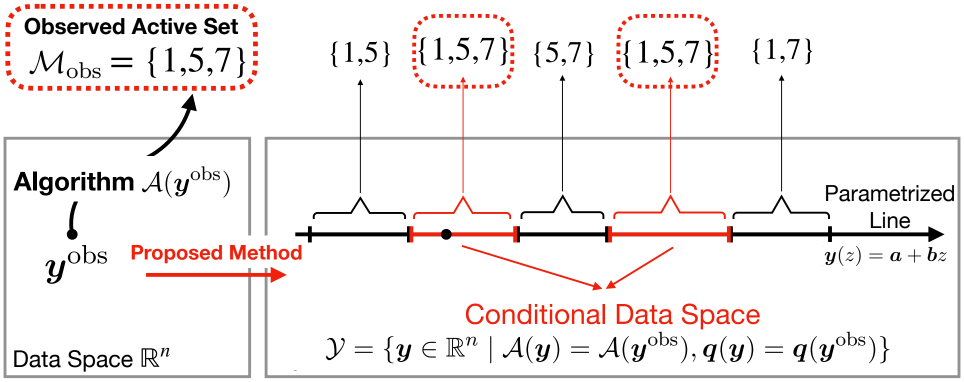

Observed Active Set
$\mathcal{M}_{\rm obs} = \{1,5,7\}$
Algorithm $\mathcal{A}(\boldsymbol{y}^{\rm obs})$
$\boldsymbol{y}^{\rm obs}$
Proposed Method
Data Space $\mathbb{R}^n$
{1,5}
{1,5,7}
{5,7}
{1,5,7}
{1,7}
Parametrized Line
$\boldsymbol{y}(z) = \boldsymbol{a} + \boldsymbol{b}z$
Conditional Data Space
$\mathcal{Y} = \{\boldsymbol{y} \in \mathbb{R}^n \mid \mathcal{A}(\boldsymbol{y}) = \mathcal{A}(\boldsymbol{y}^{\rm obs}), \boldsymbol{q}(\boldsymbol{y}) = \boldsymbol{q}(\boldsymbol{y}^{\rm obs})\}$

Figure 1: Schematic of proposed method. We obtain the observed active set $\mathcal{M}_{\rm obs}$ by applying generalized lasso to the observed data $\boldsymbol{y}^{\rm obs}$. The statistical inference for each selected element in the observed active set is conducted conditional on the subspace $\mathcal{Y}$, the data of which have the same active set as $\boldsymbol{y}^{\rm obs}$. We introduce a PP method for characterizing the conditional data space $\mathcal{Y}$ by searching the parameterized line.

In the following section, we introduce a method for identifying the minimum amount of conditioning $\{\mathcal{A}(\boldsymbol{Y}) = \mathcal{A}(\boldsymbol{y}^{\rm obs}), \boldsymbol{q}(\boldsymbol{Y}) = \boldsymbol{q}(\boldsymbol{y}^{\rm obs})\}$ that results in higher statistical power. The main concept is to compute the path of the generalized lasso solutions in the direction of interest $\boldsymbol{\eta}_j$. By focusing on the line along $\boldsymbol{\eta}_j$, the majority of irrelevant regions that do not affect the truncated normal sampling distribution can be skipped because they do not intersect with this line. Thus, we can skip the majority of combinations of signs and history that never appear when applying generalized lasso to the data on the line.

3. Proposed Method

We present the technical details of the proposed method in this section. A schematic of the method is provided in Figure 1. We first introduce a QP reformulation for the generalized lasso problem in §3.1. Thereafter, the characterization of the conditional data space is presented in §3.2. Subsequently, we propose a PP approach for identifying the conditional data space in §3.3. Finally, the detailed algorithm is presented in §3.4.

3.1 QP for Generalized Lasso

We demonstrate that the generalized lasso problem can be reformulated as a QP problem.

Lemma 1 *We denote $\boldsymbol{\xi} = D\boldsymbol{\beta}$, and the generalized lasso in (2) can be rewritten as*

$$\left(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\xi}}\right) = \mathop{\arg\min}_{\boldsymbol{\beta} \in \mathbb{R}^p, \boldsymbol{\xi} \in \mathbb{R}^m} \frac{1}{2}||\boldsymbol{y} - X\boldsymbol{\beta}||_2^2 + \lambda||\boldsymbol{\xi}||_1 \quad \textit{subject to} \quad \boldsymbol{\xi} = D\boldsymbol{\beta}. \tag{11}$$

By decomposing $\boldsymbol{\xi} = \boldsymbol{\xi}^+ - \boldsymbol{\xi}^-$, $\boldsymbol{\xi}^+, \boldsymbol{\xi}^- \geq \mathbf{0}$, *the generalized lasso problem can be formulated as the following QP problem:*

$$\left(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\xi}}^+, \hat{\boldsymbol{\xi}}^-\right) = \mathop{\arg\min}_{\boldsymbol{\beta}, \boldsymbol{\xi}^+, \boldsymbol{\xi}^-} \frac{1}{2} \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\xi}^+ \\ \boldsymbol{\xi}^- \end{pmatrix}^\top \begin{pmatrix} X^\top X & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\xi}^+ \\ \boldsymbol{\xi}^- \end{pmatrix} + \left(\lambda \begin{pmatrix} \mathbf{0}_p \\ \mathbf{1}_m \\ \mathbf{1}_m \end{pmatrix} - \begin{pmatrix} X^\top \boldsymbol{y} \\ \mathbf{0}_m \\ \mathbf{0}_m \end{pmatrix} \right)^\top \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\xi}^+ \\ \boldsymbol{\xi}^- \end{pmatrix}$$

$$s.t \begin{pmatrix} 0 & I_m & -I_m \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\xi}^+ \\ \boldsymbol{\xi}^- \end{pmatrix} = \begin{pmatrix} D & 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\xi}^+ \\ \boldsymbol{\xi}^- \end{pmatrix}, \ \boldsymbol{\xi}^+ \geq 0, \ \boldsymbol{\xi}^- \geq 0, \tag{12}$$

where $\mathbf{1}_m \in \mathbb{R}^m$ *and* $\mathbf{0}_m \in \mathbb{R}^m$ *are vectors in which all elements are set to 1 and 0, respectively, and* $I_m \in \mathbb{R}^{m \times m}$ *is an identity matrix.*

Proof The optimization problem in (11) can be rewritten as follows:

$$\left(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\xi}}\right) = \mathop{\arg\min}_{\boldsymbol{\beta}, \boldsymbol{\xi}} \frac{1}{2} \boldsymbol{\beta}^\top X^\top X \boldsymbol{\beta} - (X^\top \boldsymbol{y})^\top \boldsymbol{\beta} + \lambda ||\boldsymbol{\xi}||_1 \quad \text{s.t} \quad \boldsymbol{\xi} = D\boldsymbol{\beta}. \tag{13}$$

We obtain (12) by decomposing $\boldsymbol{\xi} = \boldsymbol{\xi}^+ - \boldsymbol{\xi}^-$ with $\boldsymbol{\xi}^+, \boldsymbol{\xi}^- \geq \mathbf{0}$. In this case, the key point is that the component $||\boldsymbol{\xi}||_1$ in (13) can be written as $||\boldsymbol{\xi}||_1 = \sum_{j \in [m]} (\xi_j^+ + \xi_j^-)$ because at least one of ξ_j^+ and ξ_j^-, $j \in [m]$, must be set to zero in the optimal solution. That is, $\hat{\xi}_j^+ > 0 \Rightarrow \hat{\xi}_j^- = 0$, and vice versa. The concept of decomposing $\boldsymbol{\xi} = \boldsymbol{\xi}^+ - \boldsymbol{\xi}^-$ is often employed to reformulate the ℓ_1-norm of $\boldsymbol{\xi}$. ■

3.2 Conditional Data Space Characterization

We define the set of $\boldsymbol{y} \in \mathbb{R}^n$ that satisfies the conditions in Equation (8):

$$\mathcal{Y} = \{\boldsymbol{y} \in \mathbb{R}^n \mid \mathcal{A}(\boldsymbol{y}) = \mathcal{A}(\boldsymbol{y}^{\text{obs}}), \boldsymbol{q}(\boldsymbol{y}) = \boldsymbol{q}(\boldsymbol{y}^{\text{obs}})\}. \tag{14}$$

According to the second condition, the data in $\mathcal{Y}$ are restricted to a line (see Section 6 in Liu et al. (2018) and Fithian et al. (2014)). Therefore, the set $\mathcal{Y}$ can be rewritten using the scalar parameter $z \in \mathbb{R}$, as follows:

$$\mathcal{Y} = \{\boldsymbol{y}(z) = \boldsymbol{a} + \boldsymbol{b}z \mid z \in \mathcal{Z}\}, \tag{15}$$

where $\boldsymbol{a} = \boldsymbol{q}(\boldsymbol{y}^{\text{obs}})$, $\boldsymbol{b} = \Sigma \boldsymbol{\eta}_j (\boldsymbol{\eta}_j^\top \Sigma \boldsymbol{\eta}_j)^{-1}$, and

$$\mathcal{Z} = \left\{ z \in \mathbb{R} \mid \mathcal{A}(\boldsymbol{y}(z)) = \mathcal{A}(\boldsymbol{y}^{\text{obs}}) \right\}. \tag{16}$$

Next, we consider a random variable $Z \in \mathbb{R}$ and its observation $z^{\text{obs}} \in \mathbb{R}$, which satisfies $\boldsymbol{Y} = \boldsymbol{a} + \boldsymbol{b}Z$ and $\boldsymbol{y}^{\text{obs}} = \boldsymbol{a} + \boldsymbol{b}z^{\text{obs}}$. The conditional inference in (8) is rewritten as the problem of characterizing the sampling distribution of

$$Z \mid \{Z \in \mathcal{Z}\}. \tag{17}$$

As $Z \sim \mathbb{N}(0, \boldsymbol{\eta}_j^\top \Sigma \boldsymbol{\eta}_j)$ under the null hypothesis, $Z \mid Z \in \mathcal{Z}$ follows a truncated normal distribution. Once the truncation region $\mathcal{Z}$ has been identified, the pivotal quantity in Equation (9) is equal to $F^{\mathcal{Z}}_{0, \boldsymbol{\eta}_j^\top \Sigma \boldsymbol{\eta}_j}(Z)$, and it can be obtained easily. Thus, the remaining task is the characterization of $\mathcal{Z}$.

Characterization of truncation region $\mathcal{Z}$. We introduce the optimization problem (12) with the parameterized response vectors $\boldsymbol{y}(z) = \boldsymbol{a} + \boldsymbol{b}z$ ($\boldsymbol{a}$, $\boldsymbol{b}$ that are defined in (15)) for $z \in \mathbb{R}$ as follows:

$$\begin{aligned} \hat{\boldsymbol{r}}(z) = \arg\min_{\boldsymbol{r}} \quad & \frac{1}{2}\boldsymbol{r}^\top P \boldsymbol{r} + (\boldsymbol{q}^0 + \boldsymbol{q}^1 z)^\top \boldsymbol{r} \\ \text{s.t.} \quad & G\boldsymbol{r} \leq \boldsymbol{h}^0 + \boldsymbol{h}^1 z, \end{aligned} \tag{18}$$

where $\boldsymbol{r} = (\boldsymbol{\beta}, \boldsymbol{\xi}^+, \boldsymbol{\xi}^-)^\top \in \mathbb{R}^{p+2m}$, $\boldsymbol{q}^0 = \left(-X^\top \boldsymbol{a}, \lambda \mathbf{1}_m, \lambda \mathbf{1}_m\right)^\top \in \mathbb{R}^{p+2m}$, $\boldsymbol{q}^1 = \left(-X^\top \boldsymbol{b}, \mathbf{0}, \mathbf{0}\right) \in \mathbb{R}^{p+2m}$, $P \in \mathbb{R}^{(p+2m)\times(p+2m)}$ and $G \in \mathbb{R}^{4m \times (p+2m)}$, $\boldsymbol{h}^0 = \boldsymbol{h}^1 = \mathbf{0}_{4m}$,

$$P = \begin{pmatrix} X^\top X & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad G = \begin{pmatrix} -D & D & 0 & 0 \\ I_m & -I_m & -I_m & 0 \\ -I_m & I_m & 0 & -I_m \end{pmatrix}^\top.$$

Let $\hat{\boldsymbol{u}}(z)$ be the vector of optimal Lagrange multipliers and $\text{row}(G)$ be the number of rows in matrix G. The KKT conditions of (18) are written as

$$\begin{aligned} P\hat{\boldsymbol{r}}(z) + \boldsymbol{q}^0 + \boldsymbol{q}^1 z + G^\top \hat{\boldsymbol{u}}(z) &= 0, \\ G\hat{\boldsymbol{r}}(z) - \boldsymbol{h}^0 - \boldsymbol{h}^1 z &\leq 0, \\ \hat{u}_i(z)(G\hat{\boldsymbol{r}}(z) - \boldsymbol{h}^0 - \boldsymbol{h}^1 z)_i &= 0, \quad \forall i \in [\text{row}(G)], \\ \hat{u}_i(z) &\geq 0, \quad \forall i \in [\text{row}(G)]. \end{aligned} \tag{19}$$

To construct the truncation region $\mathcal{Z}$ in Equation (16), we must 1) compute the entire path of $\hat{\boldsymbol{\xi}}(z) = \hat{\boldsymbol{\xi}}^+(z) - \hat{\boldsymbol{\xi}}^-(z)$ in (18), and 2) identify the set of intervals of z on which $\mathcal{A}(\boldsymbol{y}(z)) = \mathcal{A}(\boldsymbol{y}^{\text{obs}})$. However, it is difficult to compute $\hat{\boldsymbol{\xi}}^+(z)$ and $\hat{\boldsymbol{\xi}}^-(z)$ for infinitely many values of $z \in \mathbb{R}$. We demonstrate that the paths of $\hat{\boldsymbol{\xi}}^+(z)$ and $\hat{\boldsymbol{\xi}}^-(z)$ can be computed within *finite* operations by introducing parametric quadratic programming.

3.3 Piecewise Linear Homotopy

In this section, we demonstrate that $\hat{\boldsymbol{r}}(z)$ in (18) is a piecewise linear function of z, which also indicates that $\hat{\boldsymbol{\xi}}^+(z)$ and $\hat{\boldsymbol{\xi}}^-(z)$ are piecewise linear functions of z.

Lemma 2 *We denote $\mathcal{I}_z = \{i \in [\text{row}(G)] : \hat{u}_i(z) > 0\}$, $\mathcal{I}_z^c = [\text{row}(G)] \setminus \mathcal{I}_z$, and $G_{\mathcal{I}_z}$ as the rows of matrix G in a set $\mathcal{I}_z$. Consider two real values z and z' ($z < z'$). If $\mathcal{I}_z = \mathcal{I}_{z'}$, we obtain*

$$\hat{\boldsymbol{r}}(z') - \hat{\boldsymbol{r}}(z) = \boldsymbol{\psi}(z) \times (z' - z), \tag{20}$$

$$\hat{\boldsymbol{u}}_{\mathcal{I}_z}(z') - \hat{\boldsymbol{u}}_{\mathcal{I}_z}(z) = \boldsymbol{\gamma}(z) \times (z' - z), \tag{21}$$

where $\boldsymbol{\psi}(z) \in \mathbb{R}^{\text{row}(P)}, \boldsymbol{\gamma}(z) \in \mathbb{R}^{|\mathcal{I}_z|}$, $\begin{bmatrix} \boldsymbol{\psi}(z) \\ \boldsymbol{\gamma}(z) \end{bmatrix} = \begin{bmatrix} P & G_{\mathcal{I}_z}^\top \\ G_{\mathcal{I}_z} & 0 \end{bmatrix}^{-1} \begin{bmatrix} -\boldsymbol{q}^1 \\ \boldsymbol{h}^1_{\mathcal{I}_z} \end{bmatrix}.$

Proof From the KKT conditions in (19), we obtain

$$\begin{aligned} P\hat{\boldsymbol{r}}(z) + \boldsymbol{q}^0 + \boldsymbol{q}^1 z + G^\top \hat{\boldsymbol{u}}(z) = 0, \\ (G\hat{\boldsymbol{r}}(z) - \boldsymbol{h}^0 - \boldsymbol{h}^1 z)_i = 0, \quad \forall i \in \mathcal{I}_z, \\ (G\hat{\boldsymbol{r}}(z) - \boldsymbol{h}^0 - \boldsymbol{h}^1 z)_i \leq 0, \quad \forall i \in \mathcal{I}_z^c. \end{aligned} \tag{22}$$

According to (22), we obtain the following linear system:

$$\begin{aligned} & \begin{bmatrix} P & G_{\mathcal{I}_z}^\top \\ G_{\mathcal{I}_z} & 0 \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{r}}(z) \\ \hat{\boldsymbol{u}}_{\mathcal{I}_z}(z) \end{bmatrix} = \begin{bmatrix} -\boldsymbol{q}^0 \\ \boldsymbol{h}_{\mathcal{I}_z}^0 \end{bmatrix} + \begin{bmatrix} -\boldsymbol{q}^1 \\ \boldsymbol{h}_{\mathcal{I}_z}^1 \end{bmatrix} z \\ \Leftrightarrow \ & \begin{bmatrix} \hat{\boldsymbol{r}}(z) \\ \hat{\boldsymbol{u}}_{\mathcal{I}_z}(z) \end{bmatrix} = \begin{bmatrix} P & G_{\mathcal{I}_z}^\top \\ G_{\mathcal{I}_z} & 0 \end{bmatrix}^{-1} \begin{bmatrix} -\boldsymbol{q}^0 \\ \boldsymbol{h}_{\mathcal{I}_z}^0 \end{bmatrix} + \begin{bmatrix} P & G_{\mathcal{I}_z}^\top \\ G_{\mathcal{I}_z} & 0 \end{bmatrix}^{-1} \begin{bmatrix} -\boldsymbol{q}^1 \\ \boldsymbol{h}_{\mathcal{I}_z}^1 \end{bmatrix} z. \end{aligned} \tag{23}$$

Similarly, for z', we obtain

$$\begin{bmatrix} \hat{\boldsymbol{r}}(z') \\ \hat{\boldsymbol{u}}_{\mathcal{I}_{z'}}(z') \end{bmatrix} = \begin{bmatrix} P & G_{\mathcal{I}_{z'}}^\top \\ G_{\mathcal{I}_{z'}} & 0 \end{bmatrix}^{-1} \begin{bmatrix} -\boldsymbol{q}^0 \\ \boldsymbol{h}_{\mathcal{I}_{z'}}^0 \end{bmatrix} + \begin{bmatrix} P & G_{\mathcal{I}_{z'}}^\top \\ G_{\mathcal{I}_{z'}} & 0 \end{bmatrix}^{-1} \begin{bmatrix} -\boldsymbol{q}^1 \\ \boldsymbol{h}_{\mathcal{I}_{z'}}^1 \end{bmatrix} z'. \tag{24}$$

By subtracting (23) from (24) and $\mathcal{I}_z = \mathcal{I}_{z'}$, we can express the following:

$$\begin{bmatrix} \hat{\boldsymbol{r}}(z') \\ \hat{\boldsymbol{u}}_{\mathcal{I}_z}(z') \end{bmatrix} - \begin{bmatrix} \hat{\boldsymbol{r}}(z) \\ \hat{\boldsymbol{u}}_{\mathcal{I}_z}(z) \end{bmatrix} = \begin{bmatrix} P & G_{\mathcal{I}_z}^\top \\ G_{\mathcal{I}_z} & 0 \end{bmatrix}^{-1} \begin{bmatrix} -\boldsymbol{q}^1 \\ \boldsymbol{h}_{\mathcal{I}_z}^1 \end{bmatrix} \times (z' - z). \tag{25}$$

We denote $\begin{bmatrix} \boldsymbol{\psi}(z) \\ \boldsymbol{\gamma}(z) \end{bmatrix} = \begin{bmatrix} P & G_{\mathcal{I}_z}^\top \\ G_{\mathcal{I}_z} & 0 \end{bmatrix}^{-1} \begin{bmatrix} -\boldsymbol{q}^1 \\ \boldsymbol{h}_{\mathcal{I}_z}^1 \end{bmatrix}$ with $\boldsymbol{\psi}(z) \in \mathbb{R}^{\text{row}(P)}$ and $\boldsymbol{\gamma}(z) \in \mathbb{R}^{|\mathcal{I}_z|}$, and we subsequently achieve the results in Lemma 2. ■

For simplicity, we assume that the generalized lasso solution is unique for any $\boldsymbol{y}(z), z \in \mathbb{R}$. The uniqueness in the generalized lasso problem has been studied in Ali and Tibshirani (2019). In this case, it can be guaranteed that the matrix inverse in Equation (25) always exists. If this is not the case, we can use parametric quadratic programming for the degenerate cases in Best (1996).

Computation of breakpoint. According to Lemma 2, the solution $\hat{\boldsymbol{r}}(z)$ is a linear function of z until z reaches a breakpoint, at which one component of $\hat{\boldsymbol{u}}(z)$ enters or leaves the set $\mathcal{I}_z$. At this point, we discuss the identification of the breakpoint.

Lemma 3 *Consider a real value z. Subsequently, $\mathcal{I}_z = \mathcal{I}_{z'}$ for any real value z' in the interval $[z, z + t_z)$, where $z + t_z$ is the value of the breakpoint:*

$$t_z = \min\{t_z^1, t_z^2\}, \tag{26}$$

$$t_z^1 = \min_{j \in \mathcal{I}_z^c} \left(-\frac{(G_{\mathcal{I}_z^c}\hat{\boldsymbol{r}}(z) - \boldsymbol{h}_{\mathcal{I}_z^c}^0 - \boldsymbol{h}_{\mathcal{I}_z^c}^1 z)_j}{(G_{\mathcal{I}_z^c}\boldsymbol{\psi}(z) - \boldsymbol{h}_{\mathcal{I}_z^c}^1)_j} \right)_{++} \quad \text{and} \quad t_z^2 = \min_{j \in \mathcal{I}_z} \left(-\frac{\hat{u}_j(z)}{\gamma_j(z)} \right)_{++}. \tag{27}$$

In this case, for any $a \in \mathbb{R}$, $(a)_{++} = a$ if $a > 0$, and $(a)_{++} = \infty$ otherwise.

Proof We first illustrate how to derive t_z^1. According to (20), we obtain

$$\hat{\boldsymbol{r}}(z') = \hat{\boldsymbol{r}}(z) + \boldsymbol{\psi}(z) \times (z' - z).$$

Thereafter, we need to guarantee

$$\begin{aligned} & G_{\mathcal{I}_z^c} \hat{\boldsymbol{r}}(z') - \boldsymbol{h}^0_{\mathcal{I}_z^c} - \boldsymbol{h}^1_{\mathcal{I}_z^c} z' \leq 0 \\ \Leftrightarrow \ & G_{\mathcal{I}_z^c}(\hat{\boldsymbol{r}}(z) + \boldsymbol{\psi}(z) \times (z' - z)) - \boldsymbol{h}^0_{\mathcal{I}_z^c} - \boldsymbol{h}^1_{\mathcal{I}_z^c} \times (z' - z) - \boldsymbol{h}^1_{\mathcal{I}_z^c} z \leq 0 \\ \Leftrightarrow \ & \Big(G_{\mathcal{I}_z^c} \boldsymbol{\psi}(z) - \boldsymbol{h}^1_{\mathcal{I}_z^c}\Big) \times (z' - z) \leq -(G_{\mathcal{I}_z^c} \hat{\boldsymbol{r}}(z) - \boldsymbol{h}^0_{\mathcal{I}_z^c} - \boldsymbol{h}^1_{\mathcal{I}_z^c} z). \end{aligned} \tag{28}$$

The right-hand side of (28) is positive because $G_{\mathcal{I}_z^c} \hat{\boldsymbol{r}}(z) - \boldsymbol{h}^0_{\mathcal{I}_z^c} - \boldsymbol{h}^1_{\mathcal{I}_z^c} z \leq 0$. Therefore, to satisfy Equation (28),

$$z' - z \leq \min_{j \in \mathcal{I}_z^c} \left(- \frac{(G_{\mathcal{I}_z^c} \hat{\boldsymbol{r}}(z) - \boldsymbol{h}^0_{\mathcal{I}_z^c} - \boldsymbol{h}^1_{\mathcal{I}_z^c} z)_j}{(G_{\mathcal{I}_z^c} \boldsymbol{\psi}(z) - \boldsymbol{h}^1_{\mathcal{I}_z^c})_j} \right)_{++} = t_z^1.$$

Next, we explain how to derive t_z^2. According to (21), we obtain

$$\hat{\boldsymbol{u}}_{\mathcal{I}_z}(z') = \hat{\boldsymbol{u}}_{\mathcal{I}_z}(z) + \boldsymbol{\gamma}(z) \times (z' - z).$$

We need to guarantee

$$\hat{\boldsymbol{u}}_{\mathcal{I}_z}(z') > 0 \Leftrightarrow \hat{\boldsymbol{u}}_{\mathcal{I}_z}(z) + \boldsymbol{\gamma}(z) \times (z' - z) > 0. \tag{29}$$

Therefore, to satisfy Equation (29),

$$z' - z < \min_{j \in \mathcal{I}_z} \left(- \frac{\hat{u}_j(z)}{\gamma_j(z)} \right)_{++} = t_z^2.$$

Finally, using $t_z = \min\{t_z^1, t_z^2\}$, we obtain the interval in which $\mathcal{I}_{z'} = \mathcal{I}_z$ for any $z' \in [z, z + t_z)$. ■

3.4 Algorithm

In this section, we present the detailed algorithm of the proposed PP-based SI method. In Algorithm 1, to obtain the active set, we simply apply generalized lasso to the data $(X, \boldsymbol{y}^{\text{obs}})$, and we obtain $\mathcal{M}_{\text{obs}}$. Thereafter, we conduct SI for each observed active set. For every $j \in \mathcal{M}_{\text{obs}}$, we first obtain the direction of interest $\boldsymbol{\eta}_j$. The main task is to compute the solution path of $\hat{\boldsymbol{r}}(z)$ in Equation (18) for the parameterized response vector $\boldsymbol{y}(z)$, where the parameterized solution $\hat{\boldsymbol{r}}(z)$ varies for different $j \in \mathcal{M}_{\text{obs}}$ because the direction of interest $\boldsymbol{\eta}_j$ is dependent on j. This task can be achieved by Algorithm 2. Finally, after obtaining the path, we can easily determine the truncation region $\mathcal{Z}$, which is used to compute the selective p-value or selective confidence interval.

In Algorithm 2, a sequence of breakpoints is computed individually. The algorithm is initialized at $z_k = z_{\min}, k = 1$. At each z_k, the task is to determine the next breakpoint

Algorithm 1 `parametric_SI`

```
Input: $X, \boldsymbol{y}^{\text{obs}}, \lambda, D, [z_{\min}, z_{\max}]$
1: Obtain observed active set $\mathcal{M}_{\text{obs}} = \mathcal{A}(\boldsymbol{y}^{\text{obs}})$ for data $(X, \boldsymbol{y}^{\text{obs}})$
2: for each selected $j \in \mathcal{M}_{\text{obs}}$ do
3:     Compute $\boldsymbol{\eta}_j$, and subsequently calculate $\boldsymbol{a}$ and $\boldsymbol{b}$ based on $\boldsymbol{\eta}_j$ and $\boldsymbol{y}^{\text{obs}}$ ← Equation (15)
4:     $\mathcal{A}(\boldsymbol{y}(z))$ ← compute_solution_path ($X$, $\lambda$, $D$, $\boldsymbol{a}$, $\boldsymbol{b}$, $[z_{\min}, z_{\max}]$)
5:     Truncation region $\mathcal{Z} \leftarrow \{z : \mathcal{A}(\boldsymbol{y}(z)) = \mathcal{M}_{\text{obs}}\}$
6:     $P_j^{\text{selective}}$ ← Equation (10) (and/or selective confidence interval)
7: end for
Output: $\{P_j^{\text{selective}}\}_{j \in \mathcal{M}_{\text{obs}}}$ (and/or selective confidence intervals)
```

Algorithm 2 `compute_solution_path`

```
Input: $X, \lambda, D, \boldsymbol{a}, \boldsymbol{b}, [z_{\min}, z_{\max}]$
1: Initialization: $k = 1$, $z_k = z_{\min}$, $\mathcal{T} = z_k$
2: while $z_k < z_{\max}$ do
3:     $t_{z_k}, \mathcal{M}_{z_k}$ ← compute_step_size($X, z_k, \boldsymbol{a}, \boldsymbol{b}, \lambda, D$)
4:     $z_{k+1} = z_k + t_{z_k}$, $\mathcal{T} = \mathcal{T} \cup \{z_{k+1}\}$ ($z_{k+1}$ is the value of the next breakpoint)
5:     $k = k + 1$
6: end while
Output: $\{\mathcal{M}_{z_k}\}_{k \in [|\mathcal{T}|-1]}$
```

z_{k+1}. This task can be performed by computing the step size in Algorithm 3. This step is repeated until $z_k > z_{\max}$. By identifying all of the breakpoints $\{z_t\}_{t \in [|\mathcal{T}|]}$, the entire path of $\mathcal{M}_z$ for $z \in \mathbb{R}$ is expressed as

$$\mathcal{M}_z = \mathcal{A}(\boldsymbol{y}(z)) = \begin{cases} \mathcal{A}(\boldsymbol{y}(z_1)) & \text{if } z \in [z_1, z_2], \\ \mathcal{A}(\boldsymbol{y}(z_2)) & \text{if } z \in [z_2, z_3], \\ \quad \vdots & \\ \mathcal{A}(\boldsymbol{y}(z_{|\mathcal{T}|-1})) & \text{if } z \in [z_{|\mathcal{T}|-1}, z_{|\mathcal{T}|}]. \end{cases}$$

Selection of $[z_{\min}, z_{\max}]$. Under normality, very positive and negative values of z do not affect the inference. Therefore, it is reasonable to consider a range of values, such as $[-20\sigma, 20\sigma]$ (Liu et al., 2018) or $\left[-|\boldsymbol{\eta}_j^\top \boldsymbol{y}^{\text{obs}}| - 20\sigma, |\boldsymbol{\eta}_j^\top \boldsymbol{y}^{\text{obs}}| + 20\sigma\right]$ (Sugiyama et al., 2020), where σ is the standard deviation of the sampling distribution of the test statistic.

In Line 2 of Algorithm 3, $\hat{\boldsymbol{r}}(z)$ can be computed based on the KKT conditions. However, it is well known that numerical issues will arise (see Mairal and Yu (2012), the discussion below Algorithm 1), e.g., multiple and expensive inversion of ill-conditioned matrix that might cause the failure of the algorithm before exploring the entire path. Therefore, in our experiments, we modify the algorithm slightly to overcome these numerical problems, which is also motivated by Mairal and Yu (2012) (Lines 12-14 of Algorithm 2). In particular, we first replace Line 1 in Algorithm 3 with $\boldsymbol{y}(z) = \boldsymbol{a} + \boldsymbol{b}(z + \Delta z)$, where Δz is a small value such that $z_k + \Delta z < z_{k+1}$ for all $k \in [|\mathcal{T}| - 1]$. Subsequently, at Line 2 of Algorithm 3, we simply obtain $\hat{\boldsymbol{r}}(z)$ by applying the QP solver to $\boldsymbol{y}(z)$. We can confirm whether Δz is sufficiently small by verifying whether exactly one component in the vector of the optimal

Algorithm 3 `compute_step_size`

```
Input: $X, z, \boldsymbol{a}, \boldsymbol{b}, \lambda, D$
1: $\boldsymbol{y}(z) = \boldsymbol{a} + \boldsymbol{b}z$
2: Compute $\hat{\boldsymbol{r}}(z)$ for data $(X, \boldsymbol{y}(z))$ and calculate $\hat{\boldsymbol{\xi}}(z) = \hat{\boldsymbol{\xi}}^+(z) - \hat{\boldsymbol{\xi}}^-(z)$ based on $\hat{\boldsymbol{r}}(z)$
3: Obtain $\mathcal{M}_z = \mathcal{A}(\boldsymbol{y}(z)) = \{j : \hat{\xi}_j(z) \neq 0\}$
4: Compute $t_z \leftarrow$ Lemma 3
Output: $t_z, \mathcal{M}_z$
```

Lagrange multipliers $\hat{\boldsymbol{u}}(z)$ has already entered or left the set $\mathcal{I}_z$. This condition is satisfied in all of the experiments by simply setting $\Delta z = 0.0001$.

The complexity of Algorithm 1 is dependent on the number of breakpoints. In the literature on PP, the worst-case complexity increases exponentially with the problem size (Ritter, 1984; Allgower and George, 1993; Gal, 1995; Best, 1996; Mairal and Yu, 2012). However, in practice, it has been reported that the number of breakpoints is approximately linear with the problem size and it does not actually increase as much as in the theoretical worst case. This has also been noted in regularization path studies (Osborne et al., 2000; Efron and Tibshirani, 2004; Hastie et al., 2004; Mairal and Yu, 2012). In fact, in all of the experiments in §6, the number of breakpoints is within a reasonable size and the computational cost is not a major problem of the proposed method.

4. Generality of Proposed Method

Although we only focused on generalized lasso in §3, the parametric QP formulation in (18) is more general and the forms of matrices P and G as well as vectors $\boldsymbol{q}^0, \boldsymbol{q}^1, \boldsymbol{h}^0$, and $\boldsymbol{h}^1$ can be changed depending on the problem. That is, the method proposed method in §3 is flexible and can be applied to any problem that can be converted into a parametric QP in the form of (18). In this section, we demonstrate the extensions of the proposed method for testing the statistical significance of the features that are selected by various feature selection algorithms.

When applying a feature selection algorithm $\mathcal{A}$ to the observed response vector $\boldsymbol{y}^{\text{obs}}$, the observed active set can be defined as follows:

$$\mathcal{M}_{\text{obs}} = \mathcal{A}(\boldsymbol{y}^{\text{obs}}) = \{j : \hat{\beta}_j \neq 0\}.$$

In this setting, $D = I_p$. To test the selected features in $\mathcal{M}_{\text{obs}}$, the conditional inference is the same as that defined in (8) and the characterization of the truncation region $\mathcal{Z}$ is the same as (16). To identify $\mathcal{Z}$, the remaining task is to compute the solution path $\hat{\boldsymbol{\beta}}(z)$ for $z \in \mathbb{R}$ and to identify the intervals of z in which we obtain the same active set as $\boldsymbol{y}^{\text{obs}}$. In the following sections, we present the parametric QP formulations for vanilla lasso, elastic net, non-negative least squares, and Huber regression. As all of these regression problems can be converted into the form of (18), the path of $\hat{\boldsymbol{\beta}}(z)$ can be computed within finite operations by using PP, as demonstrated in §3.3.

Parametric QP for vanilla lasso. Vanilla lasso with a parameterized response vector $\boldsymbol{y}(z)$ for $z \in \mathbb{R}$ is defined as

$$\hat{\boldsymbol{\beta}}_{\text{lasso}}(z) = \mathop{\arg\min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2}||\boldsymbol{y}(z) - X\boldsymbol{\beta}||_2^2 + \lambda||\boldsymbol{\beta}||_1. \tag{30}$$

Lemma 4 *By decomposing $\boldsymbol{\beta} = \boldsymbol{\beta}^+ - \boldsymbol{\beta}^-$, $\boldsymbol{\beta}^+, \boldsymbol{\beta}^- \geq \mathbf{0}_p$, the lasso problem in (30) can be solved by the following parametric QP:*

$$\begin{aligned}
\left(\hat{\boldsymbol{\beta}}^+_{\text{lasso}}(z), \hat{\boldsymbol{\beta}}^-_{\text{lasso}}(z)\right) = \mathop{\arg\min}_{\boldsymbol{\beta}^+, \boldsymbol{\beta}^- \in \mathbb{R}^p} \quad & \frac{1}{2}\begin{pmatrix}\boldsymbol{\beta}^+ \\ \boldsymbol{\beta}^-\end{pmatrix}^\top \begin{pmatrix} X^\top X & -X^\top X \\ -X^\top X & X^\top X \end{pmatrix} \begin{pmatrix}\boldsymbol{\beta}^+ \\ \boldsymbol{\beta}^-\end{pmatrix} \\
& + \left(\lambda \mathbf{1}_{2p} - \begin{pmatrix} X^\top \boldsymbol{y}(z) \\ -X^\top \boldsymbol{y}(z)\end{pmatrix}\right)^\top \begin{pmatrix}\boldsymbol{\beta}^+ \\ \boldsymbol{\beta}^-\end{pmatrix} \\
s.t. \quad & \boldsymbol{\beta}^+ \geq \mathbf{0}_p, \boldsymbol{\beta}^- \geq \mathbf{0}_p.
\end{aligned}$$

Proof According to (30), we obtain

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_{\text{lasso}}(z) &= \mathop{\arg\min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2}||\boldsymbol{y}(z) - X\boldsymbol{\beta}||_2^2 + \lambda||\boldsymbol{\beta}||_1 \\
&= \mathop{\arg\min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2}\left(\boldsymbol{y}(z)^\top \boldsymbol{y}(z) - \boldsymbol{\beta}^\top X^\top \boldsymbol{y}(z) - \boldsymbol{y}(z)^\top X\boldsymbol{\beta} + \boldsymbol{\beta}^\top X^\top X \boldsymbol{\beta}\right) + \lambda||\boldsymbol{\beta}||_1. \quad (31)
\end{aligned}$$

Similar to the proof of Lemma 1, as the component $\frac{1}{2}\boldsymbol{y}(z)^\top \boldsymbol{y}(z)$ does not affect the optimal solution $\hat{\boldsymbol{\beta}}_{\text{lasso}}(z)$, we can rewrite (31) as

$$\hat{\boldsymbol{\beta}}_{\text{lasso}}(z) = \mathop{\arg\min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2}\boldsymbol{\beta}^\top X^\top X \boldsymbol{\beta} - (X^\top \boldsymbol{y}(z))^\top \boldsymbol{\beta} + \lambda||\boldsymbol{\beta}||_1. \tag{32}$$

Finally, by decomposing $\boldsymbol{\beta} = \boldsymbol{\beta}^+ - \boldsymbol{\beta}^-$, $\boldsymbol{\beta}^+, \boldsymbol{\beta}^- \geq \mathbf{0}_p$ and $||\boldsymbol{\beta}||_1 = \sum_{j \in [p]}(\beta_j^+ + \beta_j^-)$, we obtain the result in Lemma 4. ■

In our preliminary conference paper (Le Duy and Takeuchi, 2021), we initially presented the idea of introducing PP for vanilla lasso in a slightly different (but essentially the same) manner.

Parametric QP for elastic net. The elastic net with a parameterized response vector $\boldsymbol{y}(z)$ for $z \in \mathbb{R}$ is defined as

$$\hat{\boldsymbol{\beta}}_{\text{elastic}}(z) = \mathop{\arg\min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2n}||\boldsymbol{y}(z) - X\boldsymbol{\beta}||_2^2 + \lambda||\boldsymbol{\beta}||_1 + \frac{1}{2}\zeta||\boldsymbol{\beta}||_2^2. \tag{33}$$

Lemma 5 *By decomposing $\boldsymbol{\beta} = \boldsymbol{\beta}^+ - \boldsymbol{\beta}^-$, $\boldsymbol{\beta}^+, \boldsymbol{\beta}^- \geq \mathbf{0}_p$, the elastic net problem in (33) can be solved using the following parametric QP:*

$$\begin{aligned}\left(\hat{\boldsymbol{\beta}}^+_{\text{elastic}}(z), \hat{\boldsymbol{\beta}}^-_{\text{elastic}}(z)\right) = \mathop{\arg\min}_{\boldsymbol{\beta}^+, \boldsymbol{\beta}^- \in \mathbb{R}^p} \quad & \frac{1}{2n} \begin{pmatrix} \boldsymbol{\beta}^+ \\ \boldsymbol{\beta}^- \end{pmatrix}^\top \begin{pmatrix} X^\top X & -X^\top X \\ -X^\top X & X^\top X \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta}^+ \\ \boldsymbol{\beta}^- \end{pmatrix} \\ & + \zeta \cdot \frac{1}{2} \begin{pmatrix} \boldsymbol{\beta}^+ \\ \boldsymbol{\beta}^- \end{pmatrix}^\top \begin{pmatrix} I_p & -I_p \\ -I_p & I_p \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta}^+ \\ \boldsymbol{\beta}^- \end{pmatrix} \\ & + \left(\lambda \mathbf{1}_{2p} - \frac{1}{n} \begin{pmatrix} X^\top \boldsymbol{y}(z) \\ -X^\top \boldsymbol{y}(z) \end{pmatrix} \right)^\top \begin{pmatrix} \boldsymbol{\beta}^+ \\ \boldsymbol{\beta}^- \end{pmatrix} \\ \textit{s.t.} \quad & \boldsymbol{\beta}^+ \geq \mathbf{0}_p, \boldsymbol{\beta}^- \geq \mathbf{0}_p.\end{aligned}$$

Proof The proof of Lemma 5 is similar to the proof of Lemma 4. ■

Parametric QP for non-negative least squares. The non-negative least squares problem with a parametrized response vector $\boldsymbol{y}(z)$ for $z \in \mathbb{R}$ is defined as

$$\hat{\boldsymbol{\beta}}_{\text{non-negative}}(z) = \mathop{\arg\min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2} ||\boldsymbol{y}(z) - X\boldsymbol{\beta}||_2^2 \quad \text{s.t} \quad \boldsymbol{\beta} \geq 0.$$

The above problem can be formulated as the following parametric QP:

$$\hat{\boldsymbol{\beta}}_{\text{non-negative}}(z) = \mathop{\arg\min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2} \boldsymbol{\beta}^\top X^\top X \boldsymbol{\beta} - \left(X^\top \boldsymbol{y}(z) \right)^\top \boldsymbol{\beta} \quad \text{s.t} \quad \boldsymbol{\beta} \geq 0.$$

Parametric QP for Huber regression with ℓ_1 penalty. The Huber regression with the ℓ_1 penalty for a parameterized response vector $\boldsymbol{y}(z)$ is formulated as

$$\hat{\boldsymbol{\beta}}_{\text{huber}}(z) = \mathop{\arg\min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \sum_{i=1}^n L_\delta(y_i(z) - \boldsymbol{x}_i^\top \boldsymbol{\beta}) + \lambda ||\boldsymbol{\beta}||_1, \tag{34}$$

with

$$L_\delta(e) = \begin{cases} \frac{1}{2} e^2 & \text{if } |e| \leq \delta, \\ \delta(|e| - \frac{1}{2}\delta) & \text{otherwise,} \end{cases}$$

where $\delta > 0$ is also a predetermined tuning parameter. Let $\boldsymbol{e} = \boldsymbol{y}(z) - X\boldsymbol{\beta}$, $\boldsymbol{\phi} \in \mathbb{R}^n$, and $\boldsymbol{\nu} \in \mathbb{R}^n$ be the vectors in which the i^{th} element ϕ_i and ν_i are respectively defined as

$$\phi_i = \min\{|e_i|, \delta\}, \quad \nu_i = \max\{|e_i| - \delta, 0\},$$

where e_i is the i^{th} element of $\boldsymbol{e}$. Obviously, $|e_i| = \phi_i + \nu_i$. Subsequently, by decomposing $\boldsymbol{\beta} = \boldsymbol{\beta}^+ - \boldsymbol{\beta}^-$, $\boldsymbol{\beta}^+, \boldsymbol{\beta}^- \geq \mathbf{0}_p$, the problem in (34) can be solved by the following parametric QP:

$$\begin{aligned}\left(\hat{\boldsymbol{\phi}}(z), \hat{\boldsymbol{\nu}}(z), \hat{\boldsymbol{\beta}}^+_{\text{huber}}(z), \hat{\boldsymbol{\beta}}^-_{\text{huber}}(z)\right) = \mathop{\arg\min}_{\boldsymbol{\phi}, \boldsymbol{\nu}, \boldsymbol{\beta}^+, \boldsymbol{\beta}^-} & \frac{1}{2} \boldsymbol{\phi}^\top \boldsymbol{\phi} + \delta \cdot \mathbf{1}_n^\top \boldsymbol{\nu} + \sum_j \left(\beta_j^+ + \beta_j^- \right) \\ \text{s.t} \ & -\boldsymbol{\phi} - \boldsymbol{\nu} \leq \boldsymbol{y}(z) - X\left(\boldsymbol{\beta}^+ - \boldsymbol{\beta}^-\right) \leq \boldsymbol{\phi} + \boldsymbol{\nu}, \\ & \mathbf{0}_n \leq \boldsymbol{u} \leq \delta \cdot \mathbf{1}_n, \ \boldsymbol{v} \geq \mathbf{0}_n, \\ & \boldsymbol{\beta}^+ \geq \mathbf{0}_p, \ \boldsymbol{\beta}^- \geq \mathbf{0}_p.\end{aligned}$$

The lasso-like formulation of the Huber regression has been discussed in She and Owen (2011). Conditional SI for outliers was studied in Chen and Bien (2019), and we recently investigated its PP version in Tsukurimichi et al. (2021).

5. Characterization of CV-Based Tuning Parameter Selection Event

Various ML tasks involve careful tuning of a regularization parameter λ that controls the balance between an empirical loss term and a regularization term; for example, this is commonly achieved by CV. However, the majority of the current SI studies have assumed a pre-specified λ and have ignored the fact that λ is selected based on the data, because it is difficult to characterize the CV selection event. Loftus (2015) and Markovic et al. (2017) proposed solutions to incorporate CV events. However, the former work required additional conditioning on all intermediate models, which led to a loss of power, whereas the latter considered a randomization version of CV instead of vanilla CV.

In this section, we introduce a new means of characterizing the *minimal* selection event in which λ is selected based on the data; for example, via CV [4]. For notational simplicity, we consider the case in which the data are divided into training and validation sets, and the latter is used for selecting λ. The following discussion can easily be extended to CV scenarios. We rewrite the observed data as follows:

$$\{X, \boldsymbol{y}^{\text{obs}}\} = \left\{(X_{\text{train}}\ X_{\text{val}})^\top \in \mathbb{R}^{n \times p}, (\boldsymbol{y}^{\text{obs}}_{\text{train}}\ \boldsymbol{y}^{\text{obs}}_{\text{val}})^\top \in \mathbb{R}^n\right\}.$$

Given a set of regularization parameter candidates Λ, the process of selecting λ is as follows:

1. For each $\lambda \in \Lambda$, we first obtain $\hat{\boldsymbol{\beta}}_\lambda$ using the training data

   $$\hat{\boldsymbol{\beta}}_\lambda \in \arg\min_{\boldsymbol{\beta}} \frac{1}{2}\|\boldsymbol{y}^{\text{obs}}_{\text{train}} - X_{\text{train}}\boldsymbol{\beta}\|_2^2 + \lambda\|D\boldsymbol{\beta}\|_1.$$

   Thereafter, the validation error is defined as

   $$E_\lambda = \frac{1}{2}\|\boldsymbol{y}^{\text{obs}}_{\text{val}} - X_{\text{val}}\hat{\boldsymbol{\beta}}_\lambda\|_2^2.$$

2. We select $\lambda^{\text{obs}} = \lambda \in \Lambda$, which has the corresponding smallest validation error E_λ.

The selection event of the above validation process is defined as

$$\{\mathcal{V}(\boldsymbol{Y}) = \mathcal{V}(\boldsymbol{y}^{\text{obs}})\}, \tag{35}$$

where $\mathcal{V}(\boldsymbol{y}^{\text{obs}}) = \lambda^{\text{obs}} \in \Lambda$ is the event that λ^{obs} is selected when validation is performed on $\boldsymbol{y}^{\text{obs}}$.

After selecting λ^{obs}, we can obtain the observed active set $\mathcal{M}_{\text{obs}}$ by applying generalized lasso on $\boldsymbol{y}^{\text{obs}}$ with the selected λ^{obs}, which can be defined as in (3). However, to conduct a statistical test for each element in $\mathcal{M}_{\text{obs}}$, the conditional inference will be different from

[4] We note that the following discussion is only applicable when the number of features p is independent of n.

(8), because we must incorporate the selection event of the validation process. Therefore, for each $j \in \mathcal{M}_{\rm obs}$, we consider the following conditional inference:

$$\boldsymbol{\eta}_j^\top \boldsymbol{Y} \mid \{\mathcal{A}(\boldsymbol{Y}) = \mathcal{A}(\boldsymbol{y}^{\rm obs}), \mathcal{V}(\boldsymbol{Y}) = \mathcal{V}(\boldsymbol{y}^{\rm obs}), \boldsymbol{q}(\boldsymbol{Y}) = \boldsymbol{q}(\boldsymbol{y}^{\rm obs})\}. \tag{36}$$

According to the third condition, the data are restricted on the line, as discussed in §3.2. Therefore, the conditional data space in (36) can be rewritten as:

$$\mathcal{Y}_{\rm CV} = \{\boldsymbol{y}(z) = \boldsymbol{a} + \boldsymbol{b}z \mid z \in \mathcal{Z}_{\rm CV}\},$$

where

$$\mathcal{Z}_{\rm CV} = \{z \in \mathbb{R} \mid \mathcal{A}(\boldsymbol{y}(z)) = \mathcal{A}(\boldsymbol{y}^{\rm obs}), \mathcal{V}(\boldsymbol{y}(z)) = \mathcal{V}(\boldsymbol{y}^{\rm obs})\}.$$

The remaining task to conduct the inference is to identify $\mathcal{Z}_{\rm CV}$. We can decompose $\mathcal{Z}_{\rm CV}$ into two separate sets

$$\mathcal{Z}_{\rm CV} = \mathcal{Z}_1 \cap \mathcal{Z}_2,$$

where $\mathcal{Z}_1 = \{z \in \mathbb{R} \mid \mathcal{A}(\boldsymbol{y}(z)) = \mathcal{A}(\boldsymbol{y}^{\rm obs})\}$ and $\mathcal{Z}_2 = \{z \in \mathbb{R} \mid \mathcal{V}(\boldsymbol{y}(z)) = \mathcal{V}(\boldsymbol{y}^{\rm obs})\}$. The set $\mathcal{Z}_1$ can easily be constructed using the method proposed in the previous sections. The remaining challenge is to identify $\mathcal{Z}_2$.

To construct $\mathcal{Z}_2$, it is necessary to identify the intervals of z on which $\lambda^{\rm obs}$ has the smallest validation error. That is, we can redefine

$$\mathcal{Z}_2 = \{z \in \mathbb{R} \mid E_{\lambda^{\rm obs}}(z) \leq E_\lambda(z) \text{ for any } \lambda \in \Lambda\},$$

where

$$E_\lambda(z) = \frac{1}{2}\|\boldsymbol{y}_{\rm val}(z) - X_{\rm val}\hat{\boldsymbol{\beta}}_\lambda(z)\|_2^2, , \tag{37}$$

$$\hat{\boldsymbol{\beta}}_\lambda(z) \in \mathop{\arg\min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2}\|\boldsymbol{y}_{\rm train}(z) - X_{\rm train}\boldsymbol{\beta}\|_2^2 + \lambda\|D\boldsymbol{\beta}\|_1. \tag{38}$$

For each $\lambda \in \Lambda$, although it appears to be intractable to compute $E_\lambda(z)$ for infinitely many values of $z \in \mathbb{R}$, the task can be completed within finite operations using PP.

As demonstrated in §3.3, the optimal solution $\hat{\boldsymbol{\beta}}_\lambda(z)$ in (38) is a piecewise linear function of z. That is,

$$\hat{\boldsymbol{\beta}}_\lambda(z) = \begin{cases} \hat{\boldsymbol{\beta}}_\lambda(z_1) + \boldsymbol{s}_\lambda^1(z - z_1) & \text{if } z \in [z_1, z_2], \\ \hat{\boldsymbol{\beta}}_\lambda(z_2) + \boldsymbol{s}_\lambda^2(z - z_2) & \text{if } z \in [z_2, z_3], \\ \quad \vdots & \\ \hat{\boldsymbol{\beta}}_\lambda(z_{|\mathcal{T}_\lambda|-1}) + \boldsymbol{s}_\lambda^{|\mathcal{T}_\lambda|-1}(z - z_{|\mathcal{T}_\lambda|-1}) & \text{if } z \in [z_{|\mathcal{T}_\lambda|-1}, z_{|\mathcal{T}_\lambda|}], \end{cases} \tag{39}$$

where $\boldsymbol{s}_\lambda^{k\in|\mathcal{T}_\lambda|}$ is the slope vector, the elements of which are dependent on $\boldsymbol{\psi}(z_{k\in|\mathcal{T}_\lambda|})$ in (20). The subscript λ in $\hat{\boldsymbol{\beta}}_\lambda(z), \boldsymbol{s}_\lambda^k, \mathcal{T}_\lambda$ indicates that these components are dependent on λ.

Algorithm 4 `SI_with_K_fold_cross_validation`

```
Input: $X, \boldsymbol{y}^{\text{obs}}, \Lambda, D, K, [z_{\min}, z_{\max}]$
Conduct $K$-fold CV to select $\lambda^{\text{obs}}$
Obtain $\mathcal{M}_{\text{obs}} = \mathcal{A}(\boldsymbol{y}^{\text{obs}})$ for data $(X, \boldsymbol{y}^{\text{obs}})$ with the selected $\lambda^{\text{obs}}$
for each $j \in \mathcal{M}_{\text{obs}}$ do
    Compute $\boldsymbol{\eta}_j$, and subsequently calculate $\boldsymbol{a}$ and $\boldsymbol{b}$ based on $\boldsymbol{\eta}_j$ and $\boldsymbol{y}^{\text{obs}} \leftarrow$ Equation (15)
    Obtain $\mathcal{A}(\boldsymbol{y}(z))$ using Algorithm 2 $\leftarrow$ compute_solution_path ($X$, $\lambda^{\text{obs}}$, $D$, $\boldsymbol{a}$, $\boldsymbol{b}$, $[z_{\min}, z_{\max}]$)
    $\mathcal{Z}_1 \leftarrow \{z : \mathcal{A}(\boldsymbol{y}(z)) = \mathcal{M}_{\text{obs}}\}$    (characterize model selection event)
    $\mathcal{Z}_2 \leftarrow$ compute_$\mathcal{Z}_2(X, \boldsymbol{a}, \boldsymbol{b}, \Lambda, D, K)$    (characterize CV selection event)
    $\mathcal{Z}_{\text{CV}} = \mathcal{Z}_1 \cap \mathcal{Z}_2$
    Compute $P_j^{\text{selective}}$ in Equation (10) with truncation region $\mathcal{Z}_{\text{CV}}$
end for
Output: $\{P_j^{\text{selective}}\}_{j \in \mathcal{M}_{\text{obs}}}$
```

Moreover, because we decompose $\boldsymbol{y}(z) = (\boldsymbol{y}_{\text{train}}(z), \boldsymbol{y}_{\text{val}}(z))^\top$, the vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ can be rewritten as follows:

$$\boldsymbol{a} = (\boldsymbol{a}_{\text{train}} \ \boldsymbol{a}_{\text{val}})^\top, \quad \boldsymbol{b} = (\boldsymbol{b}_{\text{train}} \ \boldsymbol{b}_{\text{val}})^\top.$$

Therefore, we can write

$$\boldsymbol{y}_{\text{val}}(z) = \boldsymbol{a}_{\text{val}} + \boldsymbol{b}_{\text{val}} z. \tag{40}$$

According to the piecewise linearity of $\hat{\boldsymbol{\beta}}_\lambda(z)$ in (39) and the linearity of $\boldsymbol{y}_{\text{val}}(z)$ in (40), the validation error $E_\lambda(z)$ in (37) is a *piecewise quadratic* function of z, which can be expressed as follows:

$$E_\lambda(z) = \begin{cases} (1/2)||(\boldsymbol{a}_{\text{val}} - X_{\text{val}}\hat{\boldsymbol{\beta}}_\lambda(z_1) + X_{\text{val}}\boldsymbol{s}_\lambda^1 z_1) + (\boldsymbol{b}_{\text{val}} - X_{\text{val}}\boldsymbol{s}_\lambda^1)z||_2^2 & \text{if } z \in [z_1, z_2], \\ (1/2)||(\boldsymbol{a}_{\text{val}} - X_{\text{val}}\hat{\boldsymbol{\beta}}_\lambda(z_2) + X_{\text{val}}\boldsymbol{s}_\lambda^2 z_2) + (\boldsymbol{b}_{\text{val}} - X_{\text{val}}\boldsymbol{s}_\lambda^2)z||_2^2 & \text{if } z \in [z_2, z_3], \\ \vdots & \\ (1/2)||(\boldsymbol{a}_{\text{val}} - X_{\text{val}}\hat{\boldsymbol{\beta}}_\lambda(z_{|\mathcal{T}_\lambda|-1}) + X_{\text{val}}\boldsymbol{s}_\lambda^{|\mathcal{T}_\lambda|-1} z_{|\mathcal{T}_\lambda|-1}). & \\ \qquad + (\boldsymbol{b}_{\text{val}} - X_{\text{val}}\boldsymbol{s}_\lambda^{|\mathcal{T}_\lambda|-1})z||_2^2 & \text{if } z \in [z_{|\mathcal{T}_\lambda|-1}, z_{|\mathcal{T}_\lambda|}]. \end{cases}$$

At this point, for each $\lambda \in \Lambda$, we have a corresponding validation error $E_\lambda(z)$, which is a piecewise quadratic function of z. Finally, $\mathcal{Z}_2$ can be identified by determining the intervals of z in which the validation error $E_{\lambda^{\text{obs}}}(z)$ corresponding to λ^{obs} is the minimum among a set of piecewise quadratic functions. The procedure for the K-fold CV case is presented in Algorithm 4.

6. Experiments

In this section, we discuss the performance evaluation of the proposed method. First, the experimental setup is presented in §6.1. Thereafter, the results on synthetic and real data are outlined in §6.2 and §6.3, respectively.

Algorithm 5 `compute_`$\mathcal{Z}_2$

```
Input: $X, \boldsymbol{a}, \boldsymbol{b}, \Lambda, D, K$
for $\lambda \in \Lambda$ do
    for $k \leftarrow 1$ to $K$ do
        Compute $\hat{\boldsymbol{\beta}}^k_\lambda(z)$ as in (38) for fold $k$
        Compute validation error $E^k_\lambda(z)$ as in (37) for fold $k$
    end for
    Compute mean error $\bar{E}_\lambda(z) = \frac{1}{K}\sum_{k=1}^{K} E^k_\lambda(z)$ among $K$ folds for current $\lambda$ candidate
end for
$\mathcal{Z}_2 = \{z \in \mathbb{R} \mid \bar{E}_{\lambda^{\rm obs}}(z) \leq \bar{E}_\lambda(z)$ for any $\lambda \in \Lambda\}$
Output: $\mathcal{Z}_2$
```

6.1 Experimental Setup

We conducted experiments on fused lasso, which is one of the most commonly studied cases of generalized lasso, as well as on feature selection methods including vanilla lasso, elastic net, non-negative least squares, and Huber regression + ℓ_1 penalty. We executed the code on an Intel(R) Xeon(R) CPU E5-2687W v4 @ 3.00 GHz. In all the experiments, the significance level α was set to 0.05.

Methods for comparison. We present the false positive rate (FPR), true positive rate (TPR), and confidence interval (CI) for the following conditional inferences:

- Proposed method: Conditional inference *without* extra conditioning, as defined in (8), which was the main focus of this study.
- Over-conditioning (OC): Conditional inference with extra conditioning. Regarding fused lasso, OC is the method that was proposed in Hyun et al. (2018a). Regarding vanilla lasso, elastic net, and Huber regression + ℓ_1 penalty, OC is the polytope-based SI that was proposed in Lee et al. (2016).
- Data splitting (DS) (Cox, 1975): DS is the commonly used procedure for selection bias correction. In this approach, the data are randomly divided into two halves: one half is used for model selection and the other half is used for inference.

Synthetic data generation for fused lasso. We set $X = I_n$, and the matrix $D \in \mathbb{R}^{(n-1)\times n}$ was defined as

$$\begin{pmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ & & & \cdots & & \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{pmatrix}.$$

Regarding the FPR experiments, we generated 100 null data $\boldsymbol{y} = (y_1, ..., y_n)^\top$, where $y_{i\in[n]} \sim \mathbb{N}(0, 1)$ for each $n \in \{70, 80, 90, 100\}$. To test the TPR, we generated $\boldsymbol{y} = (y_1, ..., y_n)^\top$ with $n = 60$, in which

$$y_i \sim \mathbb{N}(\mu_i, 1), \quad \mu_i = \begin{cases} 0, & \text{if } 1 \leq i \leq 20 \text{ or } 41 \leq i \leq 60, \\ 0 + \Delta_\mu, & \text{if } 21 \leq i \leq 40. \end{cases}$$

We used Bonferroni correction to account for the multiplicity in all of the experiments. If v selected features (hypotheses) are tested simultaneously, Bonferroni correction tests each individual hypothesis at $\alpha^* = \alpha / v$. We ran 100 trials for each $\Delta_\mu \in \{1, 2, 3, 4\}$ and we repeated the experiment 10 times.

Synthetic data generation for feature selection methods. We generated n outcomes as $y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta} + \varepsilon_i$, $i = 1, ..., n$, where $\boldsymbol{x}_i \sim \mathbb{N}(0, I_p)$, in which $p = 5$, and $\varepsilon_i \sim \mathbb{N}(0, 1)$. We set the regularization parameter $\lambda = 1$. Bonferroni correction was also applied. For the FPR experiments, all of the elements of $\boldsymbol{\beta}$ were set to 0. For the TPR experiments, the first two elements of $\boldsymbol{\beta}$ were set to 0.25. We ran 100 trials for each $n \in \{50, 100, 150, 200\}$ and we repeated this experiment 10 times. For the CI experiments, we set $n = 100, p = 5$ and ran 100 trials.

Definition of TPR. In SI, statistical testing is only conducted when at least one hypothesis is discovered by the algorithm. Therefore, the definition of the TPR, which is also known as the *conditional power*, is as follows:

$$\text{TPR} = \frac{\text{\# correctly detected \& rejected}}{\text{\# correctly detected}}.$$

In the case of fused lasso, a detection is considered as correct if it is within $L = \pm 2$ of the true CP locations. This is because it is often difficult to identify the exact CPs accurately in the presence of noise. Many existing studies considered a detection to be correct if it was within L positions of the true CP locations (Truong et al., 2020). In the case of feature selection, # correctly detected indicates the number of truly positive features that are selected by the algorithm (e.g., lasso), whereas # rejected indicates the number of truly positive features for which the null hypothesis is rejected by the SI.

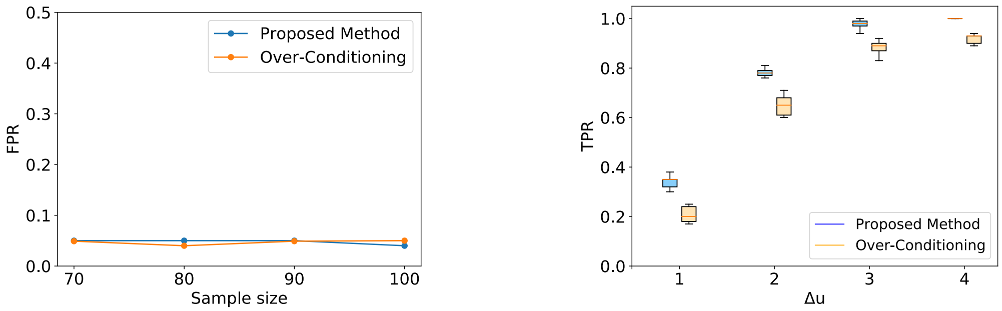
6.2 Numerical Results

FPR, TPR, and CI results. The fused lasso results are presented in Figure 2. The results for the vanilla lasso, elastic net, non-negative least squares, and Huber regression + ℓ_1 norm are depicted in Figures 3, 4, 5, and 6, respectively. No over-conditioning occurred in the case of the non-negative least squares as we had already restricted the coefficients to be positive. In summary, although all of the methods could properly control the FPR at a significance level of α, the proposed method had the highest power among the methods. The CI results were also consistent with the TPR results. That is, the shortest CI for the proposed method indicated that it exhibited the highest power.

Robustness of proposed method in terms of FPR control. We demonstrated the robustness of our method in terms of FPR control by considering the following cases:

- Non-normal noise: We considered the noise following the Laplace distribution, skew normal distribution (skewness coefficient: 10), and t_{20} distribution.
- Unknown σ^2: We considered the case in which the variance was estimated from the data.

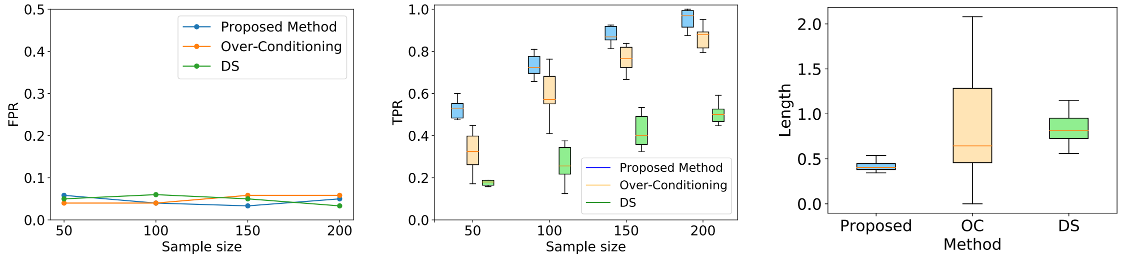
We generated n outcomes: $y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta} + \varepsilon_i$, $i = 1, ..., n$, where $p = 5, \boldsymbol{x}_i \sim \mathbb{N}(0, I_p)$, and ε_i follows a Laplace distribution, skew normal distribution, or t_{20} distribution with a zero mean and the standard deviation set to 1. In the case of the estimated σ^2, $\varepsilon_i \sim \mathbb{N}(0, 1)$.



(a) FPR

(b) TPR

Figure 2: Results of FPR and TPR for fused lasso.

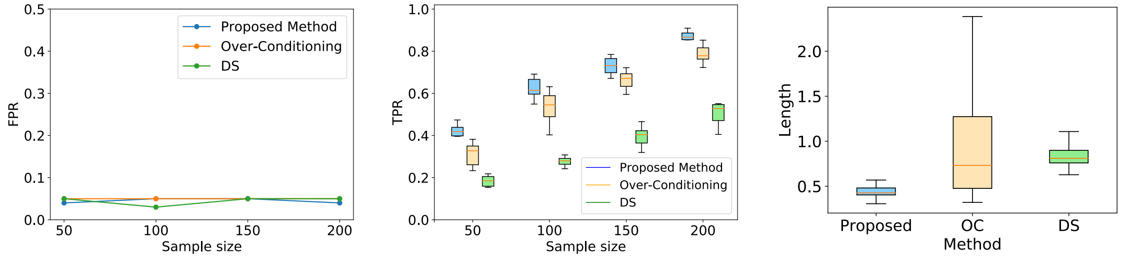


(a) FPR

(b) TPR

(c) CI

Figure 3: Results of FPR, TPR, and CI for vanilla lasso.



(a) FPR

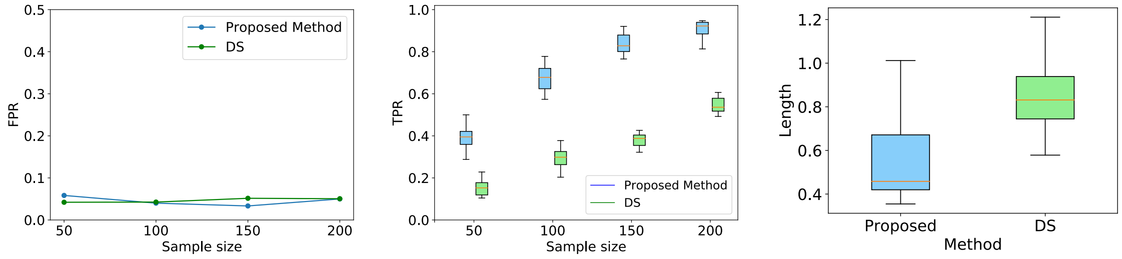
(b) TPR

(c) CI

Figure 4: Results of FPR, TPR, and CI for elastic net.

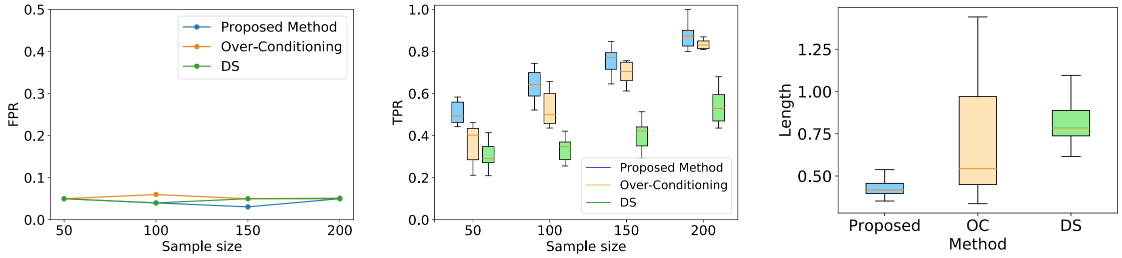
We set all elements of $\boldsymbol{\beta}$ to 0 and set $\lambda = 0.5$. For every case, we ran 1,200 trials for each $n \in \{50, 100, 150, 200\}$. We confirmed that our method maintained high performance on FPR control. The results are presented in Figures 7, 8, 9, 10, and 11.

Results when accounting for CV selection event. We conducted a comparison of the TPRs between the proposed method and the OC version that was proposed in Loftus (2015) when λ was selected from $\Lambda_1 = \{2^{-1}, 2^0, 2^1\}$ or $\Lambda_2 = \{2^{-10}, 2^{-9}, ..., 2^9, 2^{10}\}$. The results are presented in Figure 12. The existing method had lower power because additional



(a) FPR (b) TPR (c) CI

Figure 5: Results of FPR, TPR, and CI for non-negative least squares.



(a) FPR (b) TPR (c) CI

Figure 6: FPR, TPR, and CI results for Huber regression with ℓ_1 penalty.

conditioning on all intermediate models was required, which was also discussed in Markovic et al. (2017). Our method exhibited higher power as we could characterize the minimum conditioning amount.

Efficiency of proposed method. Our main purpose was to demonstrate that the proposed method has not only high statistical power, but also reasonable computational costs. We conducted experiments on feature selection by lasso. The computational time of the proposed method was almost linear with respect to the number of active features, as illustrated in Figure 13a. Figure 13b depicts the efficiency of our method compared to that of Lee et al. (2016), in which the authors mentioned the *naive* method for removing sign conditioning by enumerating all possible combinations of signs $2^{|\mathcal{M}_{\rm obs}|}$. We additionally report the number of breakpoints. The results are shown in Figure 14. The number of breakpoints is linearly increasing w.r.t $|\mathcal{M}_{\rm obs}|$ rather than exponentially increasing ($2^{|\mathcal{M}_{\rm obs}|}$).

Comparison with Liu et al. (2018). Furthermore, we demonstrated the efficiency of our method compared to the two methods proposed in §3 (inference for partial regression targets) of Liu et al. (2018). In this work, only stable features were allowed to influence the formation of the test statistic. Stable features are those with very strong signals and that cannot missed. In the first method, which we refer to as TN-ℓ_1, the stable features were selected by setting a higher value of λ. In the second method, which we refer to as TN-Custom, the stable features were selected by setting a cutoff value. The details of these two methods are presented in Appendix B. In general, to perform SI with these two methods,

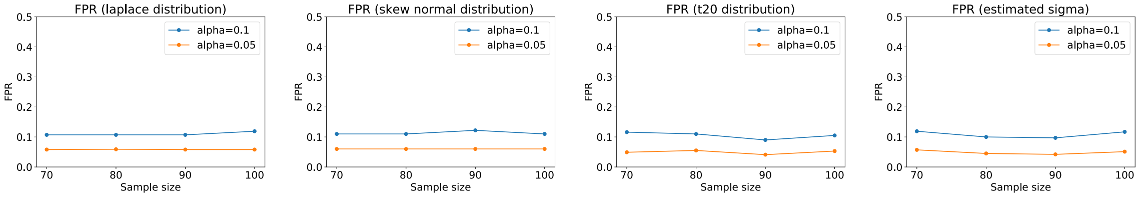


Figure 7: Robustness of proposed method for fused lasso.

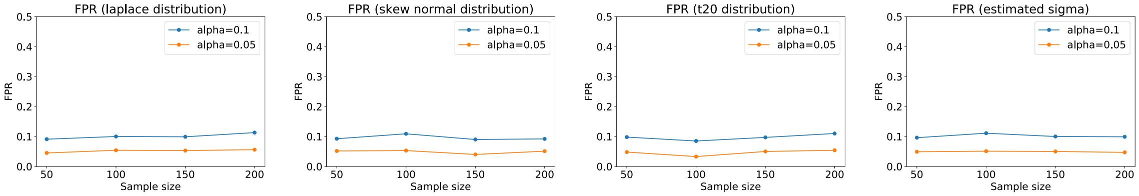


Figure 8: Robustness of proposed method for vanilla lasso.

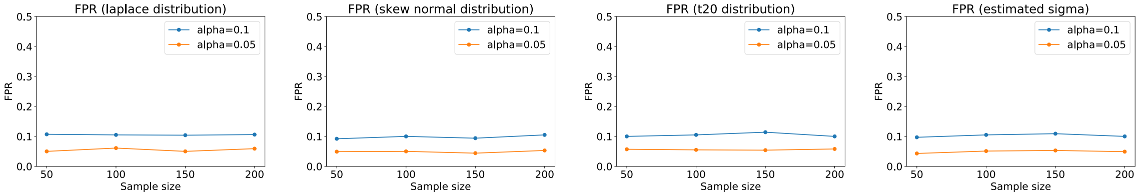


Figure 9: Robustness of proposed method for elastic net.

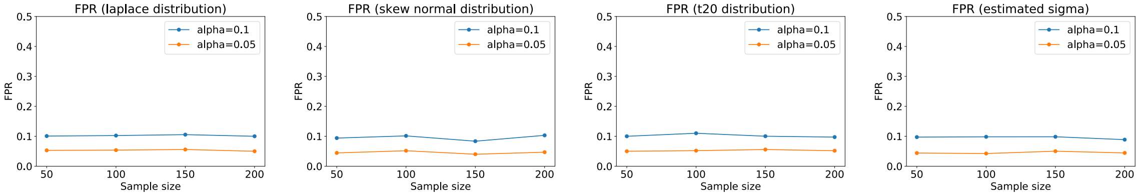


Figure 10: Robustness of proposed method for non-negative least squares.

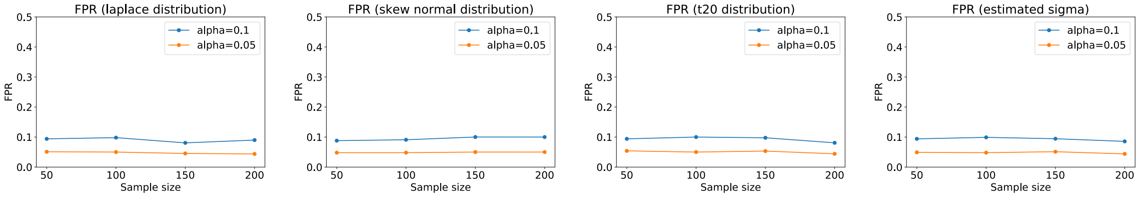
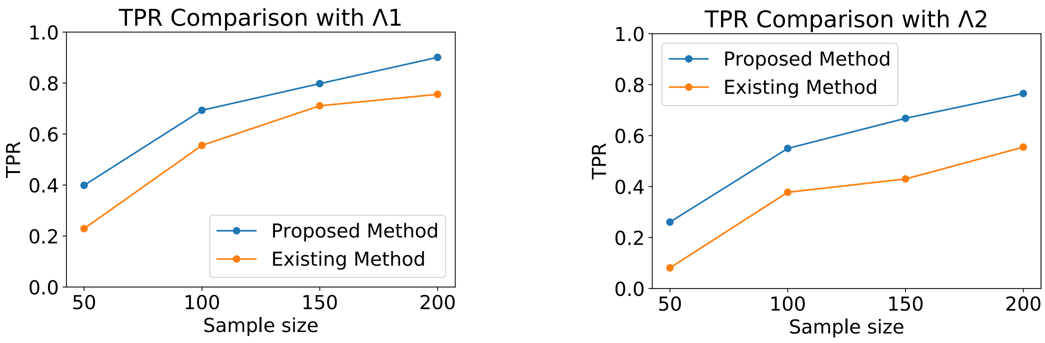


Figure 11: Robustness of proposed method for Huber regression with ℓ_1 penalty.

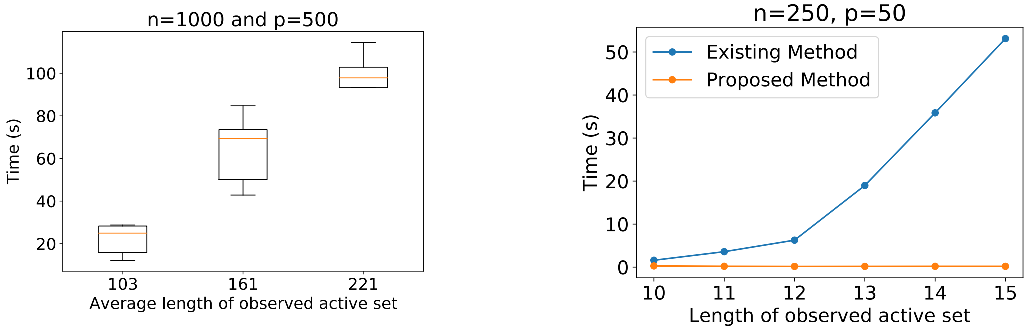
all possible combinations of signs, which increase exponentially, still need to be naively



(a) $\Lambda_1 = \{2^{-1}, 2^0, 2^1\}$

(b) $\Lambda_2 = \{2^{-10}, ..., 2^{10}\}$

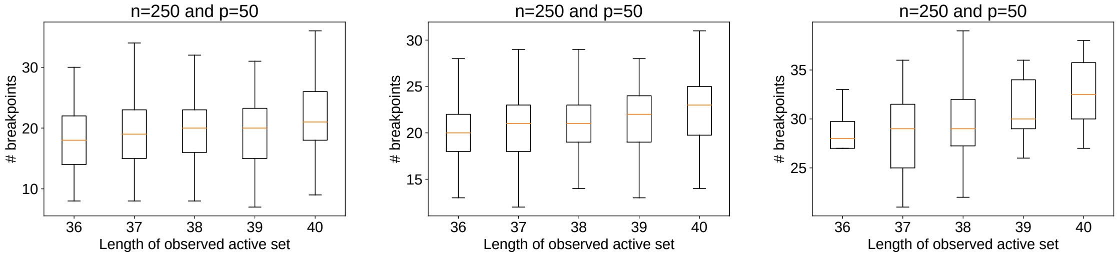
Figure 12: TPR comparison with existing method (Loftus, 2015) when accounting for CV selection event.



(a) Computational time of proposed method.

(b) Comparison with Lee et al. (2016).

Figure 13: Efficiency of proposed method.



(a) Lasso

(b) Elastic net

(c) Non-negative least squares

Figure 14: Number of breakpoints.

enumerated. The proposed method can be used to solve this computational bottleneck. A comparison of the computational costs is illustrated in Figure 15.

Settings that create larger or smaller differences between the proposed method and OC in terms of TPR. We consider the following two settings:

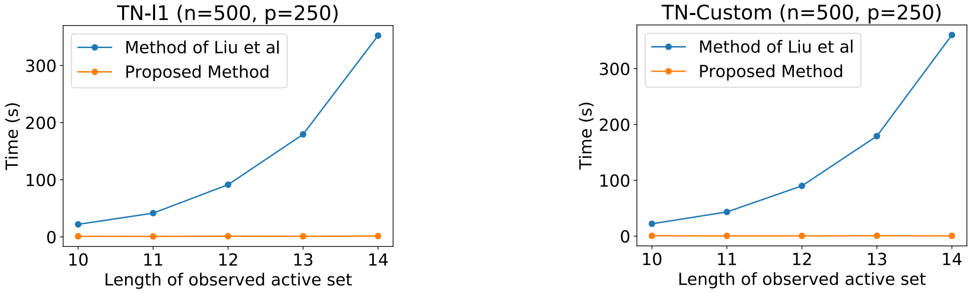
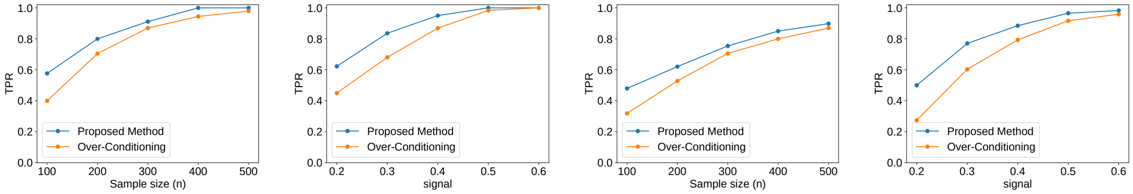


Figure 15: Comparison between proposed method and methods in Liu et al. (2018), in which an exponentially increasing number of all possible sign combinations is still required.



(a) Uncorrelated features

(b) Correlated features

Figure 16: Settings that create larger or smaller differences between the proposed method and OC in terms of TPR.

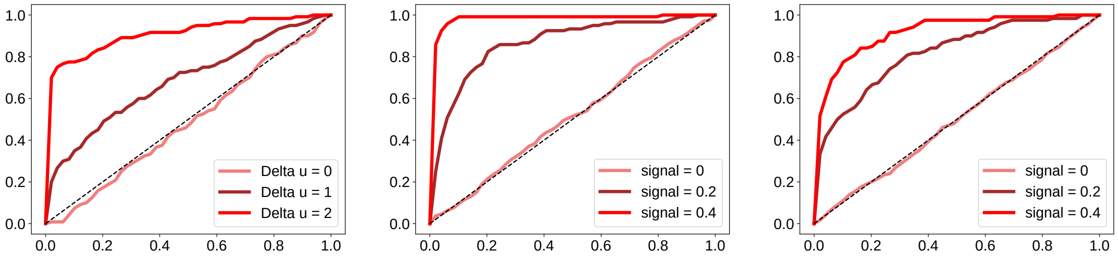
- Setting 1: we set $\boldsymbol{\beta} = \{\text{signal}, \text{signal}, 0, 0, 0\}$ where signal $= 0.2$ and consider different values of $n \in \{100, 200, 300, 400, 500\}$.
- Setting 2: we set $n = 100$ and consider $\boldsymbol{\beta} = \{\text{signal}, \text{signal}, 0, 0, 0\}$ with different values of signal $\in \{0.2, 0.3, 0.4, 0.5, 0.6\}$.

We repeated the experiments for the above two settings in two scenarios:

- Uncorrelated features: $\boldsymbol{x}_i \sim \mathbb{N}(0, \Sigma)$ where $\Sigma = I_p \in \mathbb{R}^{p \times p}$.
- Correlated features: $\boldsymbol{x}_i \sim \mathbb{N}(0, \Sigma)$ where $\Sigma = \left[0.5^{|j-k|}\right]_{jk} \in \mathbb{R}^{p \times p}$.

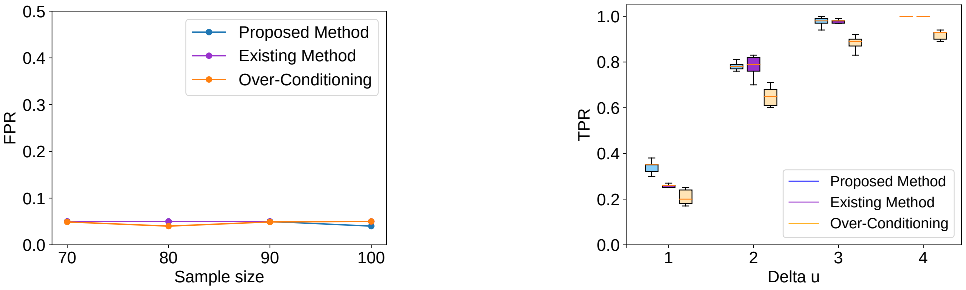
The results are shown in Figure 16. When the sample size n is small or the signal is low, the difference between the proposed method and OC is large. The difference becomes smaller when increasing n or signal.

Empirical CDF of the selective p-value. We examine the empirical CDF of the selective p-value to empirically verify the correctness of the proposed method under the null hypothesis as well as seeing the changes in the distribution of the p-value when the alternative hypothesis is true. Regarding the fused lasso, we set $\Delta_\mu \in \{0, 1, 2\}$. In regard to the lasso and elastic net, we set $\boldsymbol{\beta} = \{\text{signal}, \text{signal}, 0, 0, 0\}$ with signal $\in \{0, 0.2, 0.4\}$. The results are shown in Figure 17. Under the null hypothesis, the selective p-value is uniformly distributed. We can also see that the TPR increases when increasing Δ_μ or signal.



(a) Fused lasso (b) Lasso (c) Elastic net

Figure 17: Empirical CDF of the selective p-value.

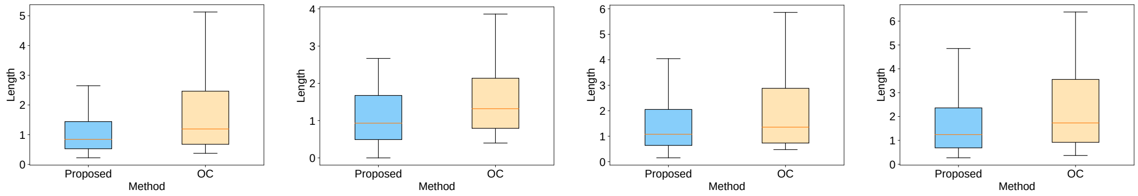


(a) FPR (b) TPR

Figure 18: Comparison with Jewell et al. (2019) (Existing Method).

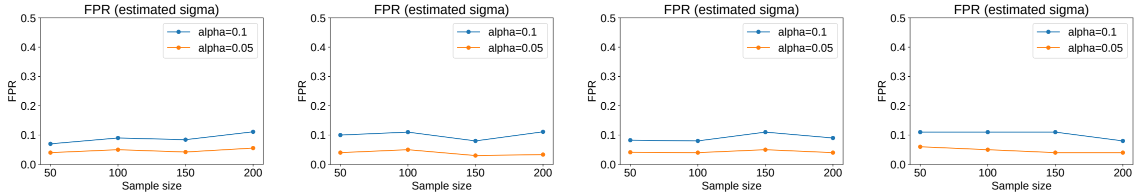
Comparison with Jewell et al. (2019). In the context of 1-dimensional changepoint detection with ℓ_0 penalty, the authors proposed an approach to remove unnecessary parts of the conditioning to conduct more powerful inference. The difference between our method and their method is that we explore the (piecewise-linear) solution path of the optimal solution in terms of an explicit scalar parameter z to characterize the inference, while Jewell et al. (2019) utilize the recursive property of solving ℓ_0 segmentation and apply it on the n-length perturbed data and its segmentation cost to characterize their inference. We additionally conducted comparison experiment in terms of FPR and TPR. The experimental setting is the same as the one described in §6.1. The results are shown in Figure 18. When $\Delta_\mu = 1$, our method has higher power. When $\Delta_\mu = 2$, the existing method has higher power. Both methods have competitive power when $\Delta_\mu \in \{3, 4\}$. In all the cases, the proposed method and existing method have higher power than the OC. This is reasonable because both methods try to remove unnecessary parts of the conditioning.

Experiments for larger p. We considered $p \in \{50, 60, 70, 80\}$. The first five elements of $\boldsymbol{\beta}$ were set to 0.25. The sample size n is set to 100. We used lasso to conduct the experiment. We compared the length of CI between the proposed method and OC. We ran 100 trials for each value of p. The results are shown in Figure 19. In all the cases, the proposed method has shorter CI than the one obtained by OC.



(a) $p = 50$ (b) $p = 60$ (c) $p = 70$ (d) $p = 80$

Figure 19: Experiments for larger $p \in \{50, 60, 70, 80\}$.



(a) $p = 250$ (b) $p = 500$ (c) $p = 750$ (d) $p = 1000$

Figure 20: Robustness of the proposed method when the variance is estimated and $p >> n$

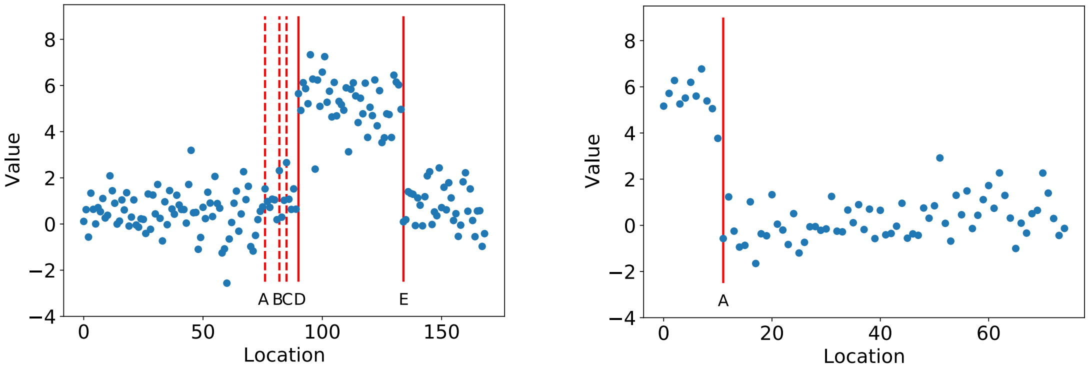
Additional study on the robustness of the proposed method when the variance is estimated and $p >> n$. We generated n outcomes as $y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta} + \varepsilon_i$, $i = 1, ..., n$, where $\boldsymbol{x}_i \sim \mathbb{N}(0, I_p)$ and $\varepsilon_i \sim \mathbb{N}(0, 1)$. We considered $p \in \{250, 500, 750, 1000\}$ and $n \in \{50, 100, 150, 200\}$. We used elastic net to conduct the experiment. We ran 100 trial for each pair (p, n). The results are shown in Figure 20. In all the cases, the proposed method still maintains good performance in terms of FPR control.

6.3 Results on Real-World Datasets

Array CGH data. Array CGH analyses enable the detection of changes in copy numbers across the genome. We applied the proposed method and OC to the dataset with the ground truth provided in Snijders et al. (2001). The results of the detected CPs and tables of p-values are presented in Figures 21 and 22. The solid red line denotes the significant CPs, which had a p-value that was smaller than the significance level following Bonferroni correction. All of the results were consistent with those of Snijders et al. (2001). Moreover, we compared the p-values of the proposed method and OC. The p-values of the proposed method were smaller than or equal to those of OC for all true CPs, which indicates that the proposed method had higher power than OC.

The boxplots of the distribution of the p-values for the proposed method and OC on the real-world dataset are illustrated in Figure 23. We used the *jointseg* package (Pierre-Jean et al., 2014) to generate realistic DNA copy number profiles of cancer samples with “known” truths. Two datasets consisting of 1,000 profiles, each with a length of $n = 60$, were created, as follows:

- Dataset 1: Resampled from GSE11976 with tumor fraction = 1.



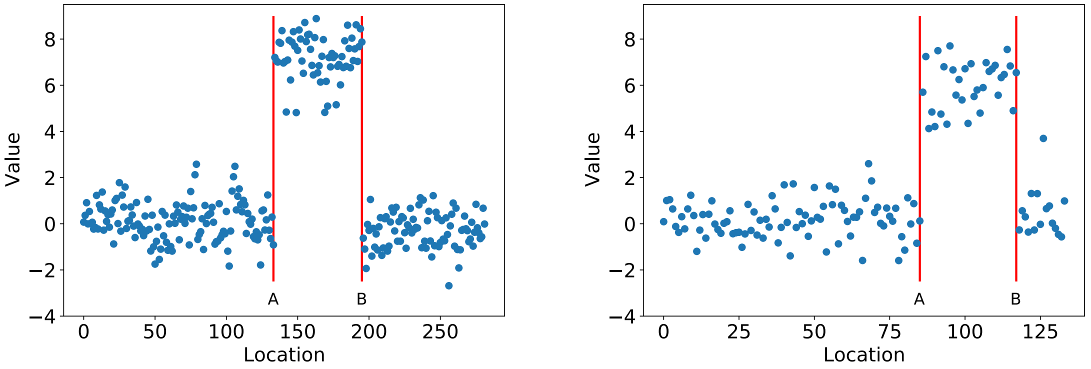
	A	B	C	D	E
Proposed	0.9	0.6	0.3	1.4×10^{-15}	0.0
OC	0.9	0.3	0.3	6.0×10^{-5}	1.2×10^{-11}

(a) Chromosomes 17, 18, and 19 ($n = p = 192$).

	A
Proposed	0.0
OC	0.0

(b) Chromosome 14 ($n = p = 79$).

Figure 21: Experimental results for cell lines GM00143 and GM01750.



	A	B
Proposed	2.7×10^{-320}	0.0
OC	5.3×10^{-98}	0.0

(a) Chromosomes 1, 2, and 3 ($n = p = 300$).

	A	B
Proposed	1.1×10^{-41}	9.1×10^{-14}
OC	1.1×10^{-41}	9.1×10^{-14}

(b) Chromosomes 20, 21, and 22 ($n = p = 151$).

Figure 22: Experimental results for cell line GM03576.

- Dataset 2: Resampled from GSE29172 with tumor fraction = 1.

Nile data. These data contain the annual flow volume of the Nile River at Aswan from 1871 to 1970 (100 years). In this case, the interest lies in unexpected events such as natural disasters. According to Figure 24, the proposed algorithm identified a CP at the 28th position, corresponding to the year 1899. This result was consistent with that of Jung et al. (2017).

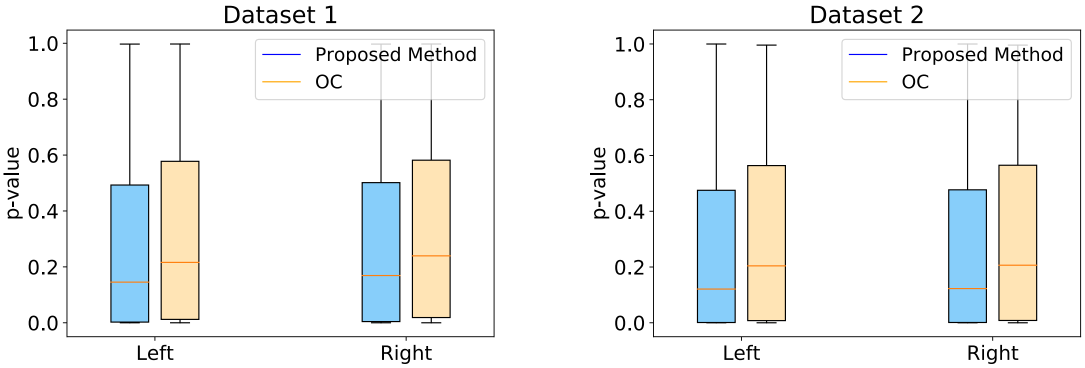


Figure 23: Boxplots of p-values. The left plot in each figure depicts the distributions of the p-values, whereas the right plot displays the distributions of the p-values for the cases in which the two p-values of the proposed method and OC differed. In Dataset 1, the percentage of the p-value of the proposed method was 55.81% smaller than that of OC. In Dataset 2, the percentage of the p-value of the proposed method was 54.24% smaller than that of OC. In general, the p-value of the proposed method tended to be smaller than that of OC, which indicates that the proposed method had higher statistical power than OC.

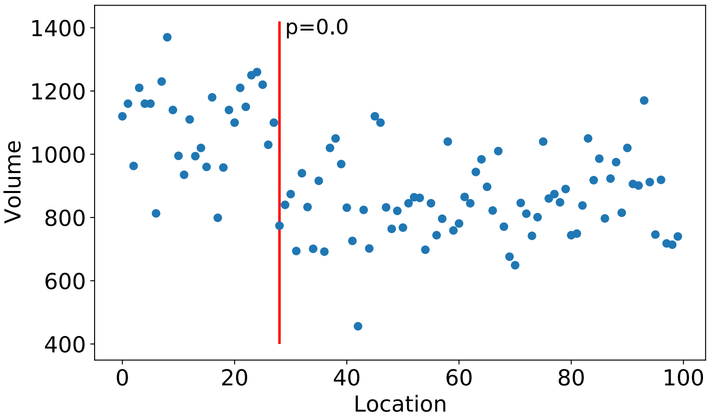


Figure 24: Experimental result for Nile data. A CP was detected at the 28^{th} position, which indicates that a change in the volume level occurred in 1899 ($n = p = 100$).

Prostate data. We applied our proposed method for lasso to the prostate dataset from Hastie et al. (2009). These data come from a study that examined the correlation between the level of prostate specific antigen and a number of clinical measures in men who were about to receive a radical prostatectomy. This dataset contains $n = 97$ rows and $p = 9$ columns. As $p < n$ for this dataset, we could estimate σ^2 using the residual sum of squares from the full regression model with all p predictors. We set $\lambda = 5$. Figure 25 depicts the 95% CIs for the features that were selected by both lasso and DS.

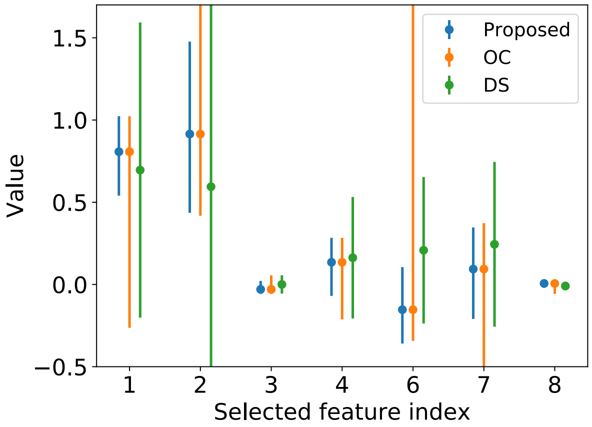


Figure 25: Experimental results for prostate data. The indices of the features were 1: lcavol, 2: lweight, 3: age, 4: lbph, 6: lcp, 7: gleason, and 8: pgg45 ($p = 9, n = 97$).

Table 1: Results on high-dimensional real-world bioinformatics related datasets.

	n	p	$\lvert\mathcal{M}_{\text{obs}}\rvert$	Avg. Time (s)
Dataset 1	89	5787	600	0.374
Dataset 2	76	5144	621	0.344
Dataset 3	133	5787	660	0.342

Other bioinformatics related datasets where $p >> n$. We demonstrate the efficiency of the proposed method by applying it on high-dimensional real-world bioinformatics related datasets, which is available at `http://www.coepra.org/CoEPrA_regr.html`. In datasets 1 and 3, n is the number of nona-peptides. Each amino acid in a nona-peptide is described by 643 descriptors, for a total of $p = 643 \times 9 = 5787$ descriptors. In dataset 2, n is the number of octa-peptides. Each amino acid in a octa-peptide is described by 643 descriptors, for a total of $p = 643 \times 8 = 5144$ descriptors. For these experiments, we used elastic net instead of lasso to obtain large $\mathcal{M}_{\text{obs}}$. The results are shown in Table 1. The time shown in the table is the average time to compute p-value for a selected feature.

7. Conclusion

We have proposed a method for characterizing the selection events of generalized lasso SI by introducing a piecewise linear PP approach. Moreover, we demonstrated that the proposed method is generally applicable to a wider class of problems that can be converted into parametric quadratic programming. The proposed method can overcomes the drawbacks of existing methods, and also improves the performance and practicality of SI in various respects. Our concept is general and can be applied to circumvent the over-conditioning problem that has occurred in many conditional SI studies. We conducted experiments on both synthetic and real-world datasets to demonstrate the effectiveness and efficiency of the proposed method.

Acknowledgements

The authors are very grateful to the anonymous reviewers and editor for their comments. This work was partially supported by MEXT KAKENHI (20H00601, 16H06538), JST CREST (JPMJCR21D3), JST Moonshot R&D (JPMJMS2033-05), NEDO (JPNP18002, JPNP20006), RIKEN Center for Advanced Intelligence Project, and RIKEN Junior Research Associate Program.

References

A. Ali and R. J. Tibshirani. The generalized lasso problem and uniqueness. *Electronic Journal of Statistics*, 13(2):2307–2347, 2019.

E. L. Allgower and K. George. Continuation and path following. *Acta Numerica*, 2:1–63, 1993.

F. R. Bach, D. Heckerman, and E. Horvits. Considering cost asymmetry in learning classifiers. *Journal of Machine Learning Research*, 7:1713–41, 2006.

Y. Benjamini and D. Yekutieli. False discovery rate–adjusted multiple confidence intervals for selected parameters. *Journal of the American Statistical Association*, 100(469):71–81, 2005.

Y. Benjamini, R. Heller, and D. Yekutieli. Selective inference in complex research. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1906):4255–4271, 2009.

R. Berk, L. Brown, A. Buja, K. Zhang, and L. Zhao. Valid post-selection inference. *The Annals of Statistics*, 41(2):802–837, 2013.

M. J. Best. An algorithm for the solution of the parametric quadratic programming problem. *Applied Mathemetics and Parallel Computing*, pages 57–76, 1996.

S. Chen and J. Bien. Valid inference corrected for outlier removal. *Journal of Computational and Graphical Statistics*, pages 1–12, 2019.

Y. Choi, J. Taylor, and R. Tibshirani. Selecting the number of principal components: Estimation of the true rank of a noisy matrix. *The Annals of Statistics*, 45(6):2590–2617, 2017.

D. R. Cox. A note on data-splitting for the evaluation of significance levels. *Biometrika*, 62 (2):441–444, 1975.

V. N. L. Duy and I. Takeuchi. Exact statistical inference for the wasserstein distance by selective inference. *Annals of the Institute of Statistical Mathematics*, pages 1–31, 2022a.

V. N. L. Duy and I. Takeuchi. Exact statistical inference for time series similarity using dynamic time warping by selective inference. *arXiv preprint arXiv:2202.06593*, 2022b.

V. N. L. Duy, H. Toda, R. Sugiyama, and I. Takeuchi. Computing valid p-value for optimal changepoint by selective inference using dynamic programming. In *Advances in Neural Information Processing Systems*, 2020.

V. N. L. Duy, S. Iwazaki, and I. Takeuchi. Quantifying statistical significance of neural network-based image segmentation by selective inference. In *Advances in Neural Information Processing Systems*, 2022.

B. Efron and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32(2):407–499, 2004.

W. Fithian, D. Sun, and J. Taylor. Optimal inference after model selection. *arXiv preprint arXiv:1410.2597*, 2014.

W. Fithian, J. Taylor, R. Tibshirani, and R. Tibshirani. Selective sequential model selection. *arXiv preprint arXiv:1512.02565*, 2015.

T. Gal. *Postoptimal Analysis, Parametric Programming, and Related Topics.* Walter de Gruyter, 1995.

T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu. The entire regularization path for the support vector machine. *Journal of Machine Learning Research*, 5:1391–415, 2004.

T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning: data mining, inference, and prediction.* Springer Science & Business Media, 2009.

T. Hocking, j. P. Vert, F. Bach, and A. Joulin. Clusterpath: an algorithm for clustering using convex fusion penalties. In *Proceedings of the 28th International Conference on Machine Learning*, pages 745–752, 2011.

P. J. Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics*, pages 492–518. Springer, 1992.

S. Hyun, M. G'sell, and R. J. Tibshirani. Exact post-selection inference for the generalized lasso path. *Electronic Journal of Statistics*, 12(1):1053–1097, 2018a.

S. Hyun, K. Lin, M. G'Sell, and R. J. Tibshirani. Post-selection inference for changepoint detection algorithms with application to copy number variation data. *arXiv preprint arXiv:1812.03644*, 2018b.

S. Jewell, P. Fearnhead, and D. Witten. Testing for a change in mean after changepoint detection. *arXiv preprint arXiv:1910.04291*, 2019.

M. Jung, S. Song, and Y. Chung. Bayesian change-point problem using bayes factor with hierarchical prior distribution. *Communications in Statistics-Theory and Methods*, 46(3): 1352–1366, 2017.

M. Karasuyama and I. Takeuchi. Nonlinear regularization path for quadratic loss support vector machines. *IEEE Transactions on Neural Networks*, 22(10):1613–1625, 2010.

M. Karasuyama, N. Harada, M. Sugiyama, and I. Takeuchi. Multi-parametric solution-path algorithm for instance-weighted support vector machines. *Machine Learning*, 88(3): 297–330, 2012.

V. N. Le Duy and I. Takeuchi. Parametric programming approach for more powerful and general lasso selective inference. In *International Conference on Artificial Intelligence and Statistics*, pages 901–909. PMLR, 2021.

G. Lee and C. Scott. The one class support vector machine solution path. In *Proc. of ICASSP 2007*, pages II521–II524, 2007.

J. D. Lee, D. L. Sun, Y. Sun, and J. E. Taylor. Exact post-selection inference, with application to the lasso. *The Annals of Statistics*, 44(3):907–927, 2016.

H. Leeb and B. M. Pötscher. Model selection and inference: Facts and fiction. *Econometric Theory*, pages 21–59, 2005.

H. Leeb and B. M. Pötscher. Can one estimate the conditional distribution of post-model-selection estimators? *The Annals of Statistics*, 34(5):2554–2591, 2006.

K. Liu, J. Markovic, and R. Tibshirani. More powerful post-selection inference, with application to the lasso. *arXiv preprint arXiv:1801.09037*, 2018.

R. Lockhart, J. Taylor, R. J. Tibshirani, and R. Tibshirani. A significance test for the lasso. *Annals of statistics*, 42(2):413, 2014.

J. R. Loftus. Selective inference after cross-validation. *arXiv preprint arXiv:1511.08866*, 2015.

J. R. Loftus and J. E. Taylor. A significance test for forward stepwise model selection. *arXiv preprint arXiv:1405.3920*, 2014.

J. Mairal and B. Yu. Complexity analysis of the lasso regularization path. *arXiv preprint arXiv:1205.0079*, 2012.

J. Markovic, L. Xia, and J. Taylor. Unifying approach to selective inference with applications to cross-validation. *arXiv preprint arXiv:1703.06559*, 2017.

K. Ogawa, M. Imamura, I. Takeuchi, and M. Sugiyama. Infinitesimal annealing for training semi-supervised support vector machines. In *International Conference on Machine Learning*, pages 897–905, 2013.

M. R. Osborne, B. Presnell, and B. A. Turlach. A new approach to variable selection in least squares problems. *IMA journal of numerical analysis*, 20(3):389–403, 2000.

S. Panigrahi, J. Taylor, and A. Weinstein. Bayesian post-selection inference in the linear model. *arXiv preprint arXiv:1605.08824*, 28, 2016.

M. Pierre-Jean, G. Rigaill, and P. Neuvial. Performance evaluation of dna copy number segmentation methods. *Briefings in bioinformatics*, 16(4):600–615, 2014.

B. M. Pötscher and U. Schneider. Confidence sets based on penalized maximum likelihood estimators in gaussian regression. *Electronic Journal of Statistics*, 4:334–360, 2010.

K. Ritter. On parametric linear and quadratic programming problems. *mathematical Programming: Proceedings of the International Congress on Mathematical Programming*, pages 307–335, 1984.

S. Rosset and J. Zhu. Piecewise linear regularized solution paths. *Annals of Statistics*, 35: 1012–1030, 2007.

Y. She and A. B. Owen. Outlier detection using nonconvex penalized regression. *Journal of the American Statistical Association*, 106(494):626–639, 2011.

A. M. Snijders, N. Nowak, R. Segraves, S. Blackwood, N. Brown, J. Conroy, G. Hamilton, A. K. Hindle, B. Huey, and K. Kimura. Assembly of microarrays for genome-wide measurement of dna copy number. *Nature genetics*, 29(3):263, 2001.

K. Sugiyama, V. N. L. Duy, and I. Takeuchi. More powerful and general selective inference for stepwise feature selection using the homotopy continuation approach. *arXiv preprint arXiv:2012.13545*, 2020.

S. Suzumura, K. Nakagawa, Y. Umezu, K. Tsuda, and I. Takeuchi. Selective inference for sparse high-order interaction models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3338–3347. JMLR. org, 2017.

I. Takeuchi and M. Sugiyama. Target neighbor consistent feature weighting for nearest neighbor classification. In *Advances in neural information processing systems*, pages 576–584, 2011.

I. Takeuchi, K. Nomura, and T. Kanamori. Nonparametric conditional density estimation using piecewise-linear solution path of kernel quantile regression. *Neural Computation*, 21(2):539–559, 2009.

I. Takeuchi, T. Hongo, M. Sugiyama, and S. Nakajima. Parametric task learning. *Advances in Neural Information Processing Systems*, 26:1358–1366, 2013.

K. Tanizaki, N. Hashimoto, Y. Inatsu, H. Hontani, and I. Takeuchi. Computing valid p-values for image segmentation by selective inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9553–9562, 2020.

J. Taylor, R. Lockhart, R. J. Tibshirani, and R. Tibshirani. Post-selection adaptive inference for least angle regression and the lasso. *arXiv preprint arXiv:1401.3889*, 354, 2014.

Y. Terada and H. Shimodaira. Selective inference after variable selection via multiscale bootstrap. *arXiv preprint arXiv:1905.10573*, 2019.

X. Tian and J. Taylor. Selective inference with a randomized response. *The Annals of Statistics*, 46(2):679–710, 2018.

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

R. J. Tibshirani and J. Taylor. The solution path of the generalized lasso. *The annals of statistics*, 39(3):1335–1371, 2011.

R. J. Tibshirani, J. Taylor, R. Lockhart, and R. Tibshirani. Exact post-selection inference for sequential regression procedures. *Journal of the American Statistical Association*, 111 (514):600–620, 2016.

C. Truong, L. Oudre, and N. Vayatis. Selective review of offline change point detection methods. *Signal Processing*, 167:107299, 2020.

K. Tsuda. Entire regularization paths for graph data. In *In Proc. of ICML 2007*, pages 919–925, 2007.

T. Tsukurimichi, Y. Inatsu, V. N. L. Duy, and I. Takeuchi. Conditional selective inference for robust regression and outlier detection using piecewise-linear homotopy continuation. *arXiv preprint arXiv:2104.10840*, 2021.

F. Yang, R. F. Barber, P. Jain, and J. Lafferty. Selective inference for group-sparse linear models. In *Advances in Neural Information Processing Systems*, pages 2469–2477, 2016.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.

Appendix A. Full Target Case for Lasso in Liu et al. (2018)

In the full target case, as discussed in Liu et al. (2018), the data are used to select the interesting features, but they are *not* used to summarize the relation between the response and the selected features. Therefore, *all* features can be used to define the direction of interest.

$$\boldsymbol{\eta}_j = X(X^\top X)^{-1}\boldsymbol{e}_j,$$

where $\boldsymbol{e}_j \in \mathbb{R}^p$ is a zero vector with 1 at the j^{th} coordinate. The conditional inference is defined as

$$\boldsymbol{\eta}_j^\top \boldsymbol{Y} \mid \left\{ j \in \mathcal{A}(\boldsymbol{Y}), \boldsymbol{q}(\boldsymbol{Y}) = \boldsymbol{q}(\boldsymbol{y}^{\text{obs}}) \right\}. \tag{41}$$

In Liu et al. (2018), the authors proposed a solution for conducting conditional inference for a specific case when $p < n$, and there was no solution for the case when $p > n$. This problem can be solved with the proposed PP method. First, we rewrite the conditional inference in (41) as the problem of characterizing the sampling distribution of

$$Z \mid \{Z \in \mathcal{Z}\} \text{ where } \mathcal{Z} = \{z \in \mathbb{R} \mid j \in \mathcal{A}(\boldsymbol{y}(z))\}. \tag{42}$$

$\boldsymbol{y}(z)$ in (42) is defined as in (15). Thereafter, to identify $\mathcal{Z}$, only the path of the lasso solution $\hat{\boldsymbol{\beta}}(z)$ needs to be obtained, as proposed in §3, and the intervals in which j is an element of the active set corresponding to $\hat{\boldsymbol{\beta}}(z)$ simply need to be verified along the path. Finally, after obtaining $\mathcal{Z}$, we can easily compute the selective p-value or selective CI.

Appendix B. Stable Partial Target Case for Lasso in Liu et al. (2018)

In the stable partial target case, as discussed in Liu et al. (2018), only stable features are allowed to influence the formation of the test statistic. Stable features are those with very strong signals that we do not wish to omit. We select a set $\mathcal{H}_{\text{obs}}$ of stable features. Subsequently, for any $j \in \mathcal{H}_{\text{obs}}, j \in \mathcal{M}_{\text{obs}}$,

$$\boldsymbol{\eta}_j = X_{\mathcal{H}_{\text{obs}}}(X^\top_{\mathcal{H}_{\text{obs}}} X_{\mathcal{H}_{\text{obs}}})^{-1}\boldsymbol{e}_j.$$

For any $j \notin \mathcal{H}_{\text{obs}}, j \in \mathcal{M}_{\text{obs}}$,

$$\boldsymbol{\eta}_j = X_{\mathcal{H}_{\text{obs}} \cup \{j\}}(X^\top_{\mathcal{H}_{\text{obs}} \cup \{j\}} X_{\mathcal{H}_{\text{obs}} \cup \{j\}})^{-1}\boldsymbol{e}_j.$$

Next, we demonstrate how to construct $\mathcal{H}_{\text{obs}}$ according to Liu et al. (2018).

Stable target formation by setting higher value of λ (TN-ℓ_1). In this case, $\mathcal{H}_{\text{obs}}$ is the lasso active set, but with a higher value of λ than that used to select $\mathcal{M}_{\text{obs}}$. We denote $\mathcal{H}_{\text{obs}} = \mathcal{H}(\boldsymbol{y}^{\text{obs}})$, and subsequently, the conditional inference is defined as

$$\boldsymbol{\eta}_j^\top \boldsymbol{Y} \mid \left\{ j \in \mathcal{A}(\boldsymbol{Y}), \mathcal{H}(\boldsymbol{Y}) = \mathcal{H}(\boldsymbol{y}^{\text{obs}}), \boldsymbol{q}(\boldsymbol{Y}) = \boldsymbol{q}(\boldsymbol{y}^{\text{obs}}) \right\}. \tag{43}$$

The main drawback of the method in Liu et al. (2018) is that all $2^{|\mathcal{H}_{\text{obs}}|}$ sign vectors must be considered, which requires substantial computation time when $|\mathcal{H}_{\text{obs}}|$ is large. This limitation can easily be overcome using our piecewise linear homotopy computation. First, we rewrite the conditional inference in (43) as the problem of characterizing the sampling distribution of

$$Z \mid \{Z \in \mathcal{Z}\} \text{ where } \mathcal{Z} = \{z \in \mathbb{R} \mid j \in \mathcal{A}(\boldsymbol{y}(z)), \mathcal{H}(\boldsymbol{y}(z)) = \mathcal{H}(\boldsymbol{y}^{\text{obs}})\}.$$

Thereafter, we can easily identify $\mathcal{Z} = \mathcal{Z}_1 \cap \mathcal{Z}_2$, where $\mathcal{Z}_1 = \{z \in \mathbb{R} \mid j \in \mathcal{A}(\boldsymbol{y}(z))\}$ and $\mathcal{Z}_2 = \{z \in \mathbb{R} \mid \mathcal{H}(\boldsymbol{y}(z)) = \mathcal{H}(\boldsymbol{y}^{\text{obs}})\}$, which we can simply obtain using the method proposed in §3 of the main paper.

Stable target formation by setting cutoff value c (TN-Custom). In this case, we determine $\mathcal{H}_{\text{obs}}$ by setting a cutoff value c to select β_j, such that $|\beta_j| \geq c$ [5]. The set $\mathcal{H}_{\text{obs}}$ is defined as

$$\mathcal{H}_{\text{obs}} = \{j \in \mathcal{M}_{\text{obs}}, |\beta_j| \geq c\},$$

where $\beta_j = \boldsymbol{e}_j^\top (X^\top_{\mathcal{M}_{\text{obs}}} X_{\mathcal{M}_{\text{obs}}})^{-1} X^\top_{\mathcal{M}_{\text{obs}}} \boldsymbol{y}^{\text{obs}}$. We denote $\mathcal{H}_{\text{obs}} = \mathcal{H}(\mathcal{M}_{\text{obs}}) \subset \mathcal{M}_{\text{obs}}$, and subsequently, the conditional inference is formulated as

$$\boldsymbol{\eta}_j^\top \boldsymbol{Y} \mid \{\mathcal{H}(\mathcal{A}(\boldsymbol{Y})) = \mathcal{H}(\mathcal{M}_{\text{obs}}), \mathcal{A}(\boldsymbol{Y}) = \mathcal{M}_{\text{obs}}\}. \tag{44}$$

The main drawback of the method in Liu et al. (2018) is that it still requires conditioning on $\{\mathcal{A}(\boldsymbol{Y}) = \mathcal{M}_{\text{obs}}\}$, which is computationally intractable when $|\mathcal{M}_{\text{obs}}|$ is large, because the enumeration of $2^{|\mathcal{M}_{\text{obs}}|}$ sign vectors is required. This limitation can easily be overcome using our proposed method.

[5] Our formulation is slightly different from but more general than that in Liu et al. (2018).