

## ABSTRACT

KONG, DEHAN. Penalized Regression Methods with Application to Domain Selection and Outlier Detection. (Under the direction of Howard Bondell and Yichao Wu.)

Variable selection is one of the most important problems in statistical literature, and a popular method is the penalized regression. In this dissertation, we investigate two types of variable selection problems via penalized regression. The first problem is related to the domain selection for the varying coefficient model, which identifies important regions of the varying coefficient that are related to the response. The second problem is related to variable selection problem for the linear model with the existence of outliers, and deal with variable selection and outlier detection simultaneously.

In Chapter 1, we give some introduction and background by overview of the variable selection methods, the local polynomial regression and the robust regression. In Chapter 2, we consider the varying coefficient model which allows the relationship between the predictors and response to vary across the domain of interest, such as time. In applications, it is possible that certain predictors only affect the response in particular regions and not everywhere. This corresponds to identifying the domain where the varying coefficient is nonzero. Towards this goal, we incorporate local polynomial smoothing and penalized regression into one framework. We establish asymptotic properties of our penalized estimators and show that they enjoy the oracle properties in the sense that they have the same bias and asymptotic variance as the local polynomial estimators as if the sparsity is known a priori. The choice of appropriate bandwidth and computational algorithms are discussed. In Chapter 2, we study the outlier detection and variable selection problem in linear regression. A mean shift parameter is added to the linear model to reflect the effect of the outlier, where an outlier has a nonzero shift parameter. We then apply an adaptive regularization on these shift parameters and shrink most of them to zero. For those observations with nonzero mean shift parameter estimates, they are regarded as outliers. Meanwhile, an L1 penalty is added to the regression parameters to select important predictors. We propose an efficient algorithm to solve this jointly penalized optimization problem and use the extended BIC tuning method to select the regularization parameters since the number of parameters exceeds the sample size. Theoretical results are provided in terms of high breakdown point, full efficiency as well as outlier detection consistency. We illustrate our method with simulation and real data. Our method is extended to high-dimensional problems with  $p \gg n$ . In Chapter 4, we end with some discussions and future work.

© Copyright 2013 by Dehan Kong

All Rights Reserved

Penalized Regression Methods with Application to Domain Selection and Outlier Detection

by  
Dehan Kong

A dissertation submitted to the Graduate Faculty of  
North Carolina State University  
in partial fulfillment of the  
requirements for the Degree of  
Doctor of Philosophy

Statistics

Raleigh, North Carolina

2013

APPROVED BY:

---

Yichao Wu

---

Wenbin Lu

---

Arnab Maity

---

Howard Bondell  
Chair of Advisory Committee

## DEDICATION

To my parents.

## BIOGRAPHY

Dehan Kong was born in Wuxi, Jiangsu Province, China on March 11, 1986. He obtained his Bachelor's degree in Mathematics from the Special Class in honor of Shiing-Shen Chern at Nankai University in Tianjin in 2008. Upon graduating from Nankai University, Mr. Kong was granted a full scholarship to North Carolina State University to earn his Doctorate in statistics. He earned his Master's degree in 2010 and is expected to graduate with his Doctorate in the Summer of 2013. His dissertation focuses on two topics: domain selection for the varying coefficient model and fully efficient robust estimation, outlier detection and variable selection via penalized regression. He is also interested in many other topics including high-dimensional statistical inference, functional data analysis, compressed sensing and neuroimaging.

## ACKNOWLEDGEMENTS

I would express my deepest gratitude to my advisors, Dr. Howard Bondell and Dr. Yichao Wu, for their insightful guidance, generous support and kind encouragement, without which I wouldn't have achieved what I have now. I would also like to thank Dr.s Wenbin Lu, Arnab Maity and Barbara Sherry for their service on my committee. I am very thankful to Dr. Rui Song, who attends my final oral defense.

I really appreciate the precious learning opportunity and environment provided by the Department of Statistics at North Carolina State University. I would also like to thank all my friends. We shared many great moments in the department, and with their supports and encouragements, I was able to go through many difficult situations.

I am extraordinarily grateful to the Nankai University, especially the Special Class of Mathematics in honor of Shiing-Shen Chern, where I took various basic mathematics courses, which helps me a great deal with my research.

Last but not the least, I would like to show my deepest appreciation to my parents for their selfless love and support. Without them, I would not go to United States and pursue a doctorate degree.

# TABLE OF CONTENTS

<b>LIST OF TABLES</b> . . . . .	<b>vii</b>
<b>LIST OF FIGURES</b> . . . . .	<b>viii</b>
<b>Chapter 1 Introduction and Background</b> . . . . .	<b>1</b>
1.1 An overview of variable selection methods . . . . .	1
1.2 An overview of the local polynomial regression . . . . .	2
1.3 An overview of the robust regression . . . . .	3
<b>Chapter 2 Domain selection for the varying coefficient model via local polynomial regression</b> . . . . .	<b>4</b>
2.1 Introduction . . . . .	4
2.2 Local polynomial regression for the varying coefficient model . . . . .	6
2.2.1 Choosing the bandwidth $h$ . . . . .	7
2.3 Penalized local polynomial regression estimation . . . . .	8
2.3.1 Algorithms . . . . .	10
2.3.2 Tuning of the regularization parameter . . . . .	10
2.4 Asymptotic properties . . . . .	11
2.5 Simulation example . . . . .	12
2.5.1 Example 1 . . . . .	13
2.5.2 Example 2 . . . . .	14
2.6 Real data application . . . . .	15
<b>Chapter 3 Fully efficient robust estimation, outlier detection and variable selection via penalized regression</b> . . . . .	<b>19</b>
3.1 Introduction . . . . .	19
3.2 Methodology . . . . .	20
3.2.1 Robust Initial Estimator . . . . .	21
3.2.2 Algorithm . . . . .	22
3.2.3 Tuning parameter selection . . . . .	22
3.3 Theoretical results . . . . .	23
3.3.1 Asymptotic theory when there are no outliers . . . . .	23
3.3.2 High breakdown point . . . . .	25
3.3.3 Outlier detection consistency . . . . .	26
3.4 Simulation Studies . . . . .	28
3.5 Real data application . . . . .	29
3.6 Extension to the high dimensional case . . . . .	31
<b>Chapter 4 Discussions and Future work</b> . . . . .	<b>33</b>
4.1 Extension of the Domain selection for the varying coefficient model . . . . .	33
4.2 Extension of the fully efficient robust estimation, outlier detection and variable selection method . . . . .	34

<b>REFERENCES</b> . . . . .	<b>39</b>
<b>APPENDICES</b> . . . . .	<b>44</b>
Appendix A Proof in Chapter 2 . . . . .	45
A.1 Lemmas . . . . .	45
A.2 Proof of Lemmas . . . . .	46
A.3 Proof of Theorems and Corollary . . . . .	48
Appendix B Proof in Chapter 3 . . . . .	53
B.1 Lemmas and Proofs . . . . .	53
B.2 Proof of Theorems and Corollaries . . . . .	54

## LIST OF TABLES

Table 2.1	Simulation results for the univariate case using penalized local polynomial regression and original local polynomial regression when sample size varies from $n = 100, 200, 500$ . The entries in the table denotes mean square error (MSE) of the estimated function, the correct zero coverage (correctzero) and the estimation error (EE) for the entire model. . . . .	14
Table 2.2	Simulation results for the multivariate case using penalized local polynomial regression and original local polynomial regression when sample size varies from $n = 100, 200, 500$ . The entries in the table denotes mean square error of the estimated function $a_1(\cdot)$ (MSE (function 1)) and $a_2(\cdot)$ (MSE (function 2)), the correct zero coverage (correctzero) for $a_1(\cdot)$ (correctzero(function 1)) and $a_2(\cdot)$ (correctzero(function 2)), and the estimation error (EE) for the entire model. . . . .	15
Table 3.1	Simulation results for our method compared with the SLTS, LL and REWLS methods. . . . .	30

## LIST OF FIGURES

Figure 2.1	Plots of $a_1(u)$ (left) and $a_2(u)$ (right) for simulation example. . . . .	13
Figure 2.2	Histogram for $\sqrt{LSTAT}$ . . . . .	16
Figure 2.3	The penalized local polynomial estimates for the coefficient functions. Panels (a) (b) (c) (d) and (e) correspond to the estimates of the coefficient functions corresponding to the intercept, the variables CRIM, RM, TAX and NOX respectively. . . . .	18

# Chapter 1

## Introduction and Background

### 1.1 An overview of variable selection methods

Variable selection is an important topic in statistical applications. Different classical methods have been introduced such as best subset selection, forward selection, backward elimination and stepwise selection. These methods provide some candidate models through different algorithms and then compare the performance of these models through some criteria, for example the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC). These classical selection methods are intuitive, simple to implement and work well in practice. However, they also suffer from several drawbacks. First, they are not stable, i.e. a small perturbation of the data may result in very different models being selected. Second, it is difficult to establish the asymptotic properties of the estimator. Third, these methods are computationally expensive for many modern statistical problems, especially for the high dimensional data where the dimension of the predictor is higher than the sample size.

Recently, the shrinkage method is developed that revolutionizes the variable selection. Suppose we have data  $\mathbf{y} = (y_1, \dots, y_n)^\top$ ,  $\mathbf{X} = (\mathbf{X}_1^\top, \dots, \mathbf{X}_n^\top)^\top$  where  $\mathbf{X}_i$  is a  $p$ -dimensional vector. Let  $L(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})$  be the loss function for the data  $(\mathbf{y}, \mathbf{X})$  with some  $p$ -dimensional parameter  $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)^\top$ , we solve  $\min_{\boldsymbol{\beta}} [L(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X}) + P_\lambda(\boldsymbol{\beta})]$ , where  $P_\lambda$  is called the penalty function.

The most popular shrinkage method used for variable selection is the least absolute shrinkage and selection operator (Lasso) [50], which adopts an L-1 penalty function  $P_\lambda(\boldsymbol{\beta}) = \sum_{j=1}^p P_\lambda(\beta_j) = \lambda \sum_{j=1}^p |\beta_j|$ . The Lasso penalty may shrink some of the components of  $\boldsymbol{\beta}$  to zero, which achieves variable selection. However, Lasso estimator would create bias for nonzero components. To remedy this issue, [15] proposed the Smoothly Clipped Absolute Deviation (SCAD) penalty. For each  $\beta_j$ , the derivative of the SCAD penalty function is given by

$$P'_\lambda(\beta_j) = \lambda \{I(|\beta_j| \leq \lambda) + \frac{(a\lambda - |\beta_j|)_+}{(a-1)\lambda} I(|\beta_j| > \lambda)\} \text{ for some } a > 2,$$

and the penalty function itself satisfies  $P_\lambda(0) = 0$ . Unlike the Lasso penalty, SCAD penalty functions have flat tails, which reduce the biases. The estimator possesses three good properties: sparsity, unbiasedness and continuity, which are also called the oracle property. Another variable selection technique enjoying the oracle property is Adaptive Lasso, proposed by [61]. Unlike the Lasso penalty term  $\lambda \sum_{j=1}^p |\beta_j|$ , they introduced the weighted penalized term  $\lambda \sum_{j=1}^p w_j |\beta_j|$  and recommended using  $w_j = 1/|\hat{\beta}_j|^\gamma$  with some  $\gamma \geq 0$ , where  $\hat{\beta}_j$  is the least square estimate. There are some other penalized regression methods such as the nonnegative garrote [3], the bridge regression [21], the elastic net [62] and the minimax concave penalty (MCP) [58], among others.

For all these penalized regression methods, they involve a tuning parameter  $\lambda$ , which controls the size of the model. There are several criteria such as cross validation (CV), generalized cross validation (GCV), AIC, BIC and so on. For the SCAD penalty, [51] pointed out that CV, GCV or AIC may result in an overfitting. They advocated the BIC tuning parameter selector, which could identify the true sparse model consistently. Consequently, BIC is preferred for the SCAD penalty, which is also used in Chapter 2 of the thesis when we apply the SCAD penalty.

## 1.2 An overview of the local polynomial regression

Local polynomial regression is a useful technique for nonparametric smoothing. Simply speaking, local polynomial regression fits a weighted polynomial regression in some neighborhood of the point  $x_0$  to obtain the function estimate at  $x_0$ . We will have an brief overview of the local polynomial regression here, and for a complete review, see the well written book [13].

Suppose  $\{(x_i, y_i), 1 \leq i \leq n\}$  are pairs of data following

$$y_i = f(x_i) + \sigma(x_i)\epsilon_i,$$

where  $\epsilon_i$  are assumed to be independent and identically distributed (iid) standard normal. To estimate  $f(x)$  at a fixed point  $x_0$ , we may use Taylor's expansion:

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0) + \dots + \frac{f^{(p)}(x_0)}{p!}(x - x_0)^p$$

We can estimate  $f(x_0), f'(x_0), \dots, f^{(p)}(x_0)$  by minimizing the following objective function

$$\min_{\beta_0, \beta_1, \dots, \beta_p} \sum_{i=1}^n \left\{ y_i - \sum_{j=0}^p \beta_j (x_i - x_0)^j \right\}^2 K_h(x_i - x_0), \quad (1.1)$$

where  $h$  is called the bandwidth controlling the neighborhood around  $x_0$ . The function  $K_h(\cdot) =$

$K(\cdot/h)/h$ , where  $K(\cdot)$  is a kernel function. The kernel function is defined as a nonnegative symmetric density function satisfying  $\int K(t)dt = 1$ . There are numerous choices for the kernel function, examples include Gaussian kernel, Epanechnikov kernel and others.

Denote the solution of (1.1) as  $\hat{\beta} = (\hat{\beta}_1, \dots, \hat{\beta}_p)^\top$ , we have  $\hat{f}^{(r)}(x_0) = r!\hat{\beta}_r$  for  $0 \leq r \leq p$ . Denote  $\mathbf{X}$  as a  $n \times (p+1)$  matrix with  $ij$ th element  $(x_i - x_0)^{j-1}$ ,  $\mathbf{W}$  be a  $n \times n$  diagonal matrix with  $i$ th element  $K_h(x_i - x_0)$  and  $\mathbf{y} = (y_1, \dots, y_n)^\top$ , we have  $\hat{\beta} = (\mathbf{X}^\top \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{W} \mathbf{y}$ .

As the diagonal matrix  $\mathbf{W}$  is related to  $h$ , the estimates of the function depend on the choice of the bandwidth. Since the bandwidth controls the size of the neighborhood, it is essential in local polynomial fitting. There are various literatures on how to select the appropriate bandwidth. The basic idea is to minimize the Mean Integrated Squared Error (MISE), which is calculated by integrating the conditional Mean Square Error (MSE) over the domain of  $\{x_i, 1 \leq i \leq n\}$ . Various approaches can be used including cross validation, rule of thumb and some multi-stage methods. We will discuss some of these methods in Chapter 2, and for a complete review, we direct the readers to [13].

### 1.3 An overview of the robust regression

Consider the linear regression model

$$y_i = \mathbf{X}_i \beta + \epsilon_i,$$

where  $\mathbf{X}_i$  is a  $p$  dimensional predictor,  $\beta$  is a  $p \times 1$  parameter, and  $\epsilon_i$  are the random error iid with mean zero. The Ordinary Least Squares (OLS) estimates are fully efficient when the error follows a normal distribution. However, when the errors deviate from the normal distribution or some outliers exist, OLS estimates may perform very badly. To remedy this issue, various robust regression methods can be used. Basically, these methods work better than OLS estimates as they are not influenced much by the outliers.

There are many robust regression methods in literature, including the M-estimates [33], the Least Median of Squares estimates [49], the Least Trimmed Squares estimates [46], S-estimates [44], Generalized S-estimates [9], MM-estimates [56], Mallows Generalized M-estimates [40], the Robust and Efficient Weighted Least Squares Estimators [23] and the robust regression via two-stage generalized empirical likelihood method [2], among others. These robust regression methods have their advantages and disadvantages in terms of the robust measure properties and the relative efficiency to the OLS estimates when the errors are normally distributed. We will have detail discussions and comparisons of these methods in Chapter 3.

## Chapter 2

# Domain selection for the varying coefficient model via local polynomial regression

### 2.1 Introduction

In this paper, we consider the varying coefficient model [6, 27], which assumes that the covariate effect may vary depending on the value of an underlying variable, such as time. The varying coefficient model is used in a variety of applications such as longitudinal data analysis. The varying coefficient model is given by:

$$Y = \mathbf{x}^\top \mathbf{a}(U) + \epsilon, \quad (2.1)$$

with  $E(\epsilon) = 0$ , and  $\text{Var}(\epsilon) = \sigma^2(U)$ . The predictor  $\mathbf{x} = (x_1, \dots, x_p)^\top$ , represents  $p$  features, and correspondingly,  $\mathbf{a}(U) = (a_1(U), \dots, a_p(U))^\top$  denotes the effect of each feature over the domain of the variable  $U$ .

For model (2.1), there are various proposals for estimation, for example local polynomial smoothing [13, 55, 28, 35, 20], polynomial splines [30, 31, 29], and smoothing splines [27, 28, 5]. In this paper, we will not only consider estimation for the varying coefficient model, but also we wish to identify which regions in the domain of  $U$  for which predictors have an effect and those regions where it may not. This is similar, although different than variable selection, as selection methods attempt to decide whether a variable is active or not while our interest focuses on identifying regions.

For variable selection in a traditional linear model, various shrinkage methods have been developed including least absolute shrinkage and selection operator (lasso) [50], Smoothly Clipped

Absolute Deviation (SCAD) [15], adaptive lasso [61] and excessively others. Although the lasso penalty gives sparse solutions, the estimates can be biased for large coefficients due to the linearity of the L1 penalty. To remedy this bias issue, [15] proposed the SCAD penalty and showed that the estimators enjoy the oracle property in the sense that not only it can select the correct submodel consistently, but also the asymptotic covariance of the estimator is the same as the asymptotic covariance matrix of the ordinary least squares (OLS) estimate as if the true subset model is known as a priori. To achieve the goal of grouped variable selection, [57] developed the group lasso penalty which penalized coefficients as a group in situations such as a factor in ANOVA. As with the lasso, the group lasso estimators do not enjoy the oracle property. As a remedy, [53] proposed the group SCAD penalty, which again selects the variables in a group manner.

For the varying coefficient model, existing works focus on identifying the nonzero coefficient functions, which achieves component selection for the varying coefficient functions. However, each coefficient function is either zero everywhere or else nonzero everywhere. For example, [54] considered the varying coefficient model under the framework of a B-spline basis and used the group SCAD to select the significant coefficient functions. [52] combined local constant regression and the group SCAD penalization together to select the components, while [37] directly applied the component selection and smoothing operator [38].

In this paper, we consider a different problem, detecting the nonzero regions for each component of the varying coefficient functions. Specifically, we would select the nonzero domain of each  $a_j(U)$ , which corresponds to the regions where the  $j$ th component of  $\mathbf{x}$  has an effect on  $Y$ . To this end, we incorporate local polynomial smoothing together with penalized regression. More specifically, we combine local linear smoothing and the group SCAD shrinkage method into one framework, which selects the nonzero regions for the coefficient functions and estimates them simultaneously. In this method, we deal with two tuning parameters, namely the bandwidth used in local polynomial smoothing and the shrinkage parameter used in the regularization method. We propose methods to select these two tuning parameters. Our theoretical results show that the resulting estimators have the same asymptotic bias and variance as the original local polynomial regression estimators.

The rest of the chapter is organized as follows. Section 2.2 reviews the local polynomial estimation for the varying coefficient model. Section 2.3 describes our methodology including the penalized estimation method and tuning procedure. Asymptotic properties are presented in Section 2.4. Section 2.5 evaluates the performance of our selection and estimation method. In Section 2.6, we apply our methods to the real data.

## 2.2 Local polynomial regression for the varying coefficient model

Suppose we have independent and identically distributed (iid) samples  $\{(U_i, \mathbf{x}_i^\top, Y_i)^\top, i = 1, \dots, n\}$  from the population  $(U, \mathbf{x}^\top, Y)^\top$  satisfying model (2.1). As  $\mathbf{a}(u)$  is a vector of unspecified functions, a smoothing method must be incorporated for estimation. In this article, we adopt the local linear approximation for this varying coefficient model [19]. For  $U$  in a small neighborhood of  $u$ , we can approximate the function  $a_j(U)$ ,  $1 \leq j \leq p$ , locally by linear regression

$$a_j(U) \approx a_j(u) + a'_j(u)(U - u).$$

For a fixed point  $u$ , denote  $a_j$  and  $b_j$  as  $a_j(u)$  and  $a'_j(u)$  respectively, and denote the estimates of  $a_j(u)$  and  $a'_j(u)$  as  $\hat{a}_j$  and  $\hat{b}_j$ , which give the function estimate and the derivative estimate respectively for the function  $a_j(\cdot)$  at the point  $u$ . For  $(\hat{a}_j, \hat{b}_j)$  ( $1 \leq j \leq p$ ), they can be estimated via local polynomial regression by solving the following optimization problem:

$$\min_{\mathbf{a}, \mathbf{b}} \sum_{i=1}^n \{Y_i - \mathbf{x}_i^\top \mathbf{a} - \mathbf{x}_i^\top \mathbf{b}(U_i - u)\}^2 (K_h(U_i - u)/K_h(0)), \quad (2.2)$$

where  $\mathbf{a} = (a_1, \dots, a_p)^\top$  and  $\mathbf{b} = (b_1, \dots, b_p)^\top$ ,  $K_h(t) = K(t/h)/h$ , and  $K(t)$  is a kernel function. The parameter  $h > 0$  is the bandwidth controlling the size of the local neighborhood. The model complexity is controlled by the bandwidth  $h$ , consequently choosing the bandwidth is essential in local polynomial regression. We will discuss how to select the bandwidth  $h$  in section 2.2.1.

The kernel function  $K$  is a nonnegative symmetric density function satisfying  $\int K(t)dt = 1$ . There are numerous choices for the kernel function, examples include Gaussian kernel ( $K(t) = \frac{1}{\sqrt{2\pi}} \exp(-t^2/2)$ ), Epanechnikov kernel ( $K(t) = 0.75(1 - t^2)_+$ ) and others. Typically, the estimates are not sensitive to the choice of the kernel function. In this paper, we will use the Epanechnikov kernel, which leads to computational efficiency due to its bounded support.

Notice here our loss function is slightly different from the loss function of the traditional local polynomial regression for varying coefficient model [19]. We have rescaled the original loss function by a term  $K_h(0)$ . For a fixed  $h$ , this change does not affect the estimates. However, this scaling is needed later to correctly balance the loss function and penalty term since  $K_h(U_i - u) = K(\frac{U_i - u}{h})/h$ , we include the term  $K_h(0)$  to eliminate the effect of  $h$  so that  $K_h(U_i - u)/K_h(0) = O(1)$ .

Denote  $\mathbf{a}_0 = (a_{10}, \dots, a_{p0})^\top$  and  $\mathbf{b}_0 = (b_{10}, \dots, b_{p0})^\top$  as the true value of the function values and true derivative values and  $\hat{\mathbf{a}} = (\hat{a}_{10}, \dots, \hat{a}_{p0})^\top$  and  $\hat{\mathbf{b}} = (\hat{b}_{10}, \dots, \hat{b}_{p0})^\top$  as the local polynomial regression estimates. Let  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)^\top$  to be a  $n \times p$  matrix. Further, denote  $\boldsymbol{\gamma}_j = (a_j, b_j)^\top$ ,  $1 \leq j \leq p$  and  $\boldsymbol{\gamma} = (\boldsymbol{\gamma}_1^\top, \dots, \boldsymbol{\gamma}_p^\top)^\top$  be a  $2p$  dimensional vector. Define  $\mathbf{U}_u =$

$\text{diag}(U_1 - u, \dots, U_n - u)$ , where  $\text{diag}(u_1, \dots, u_n)$  denotes the matrix with  $(u_1, \dots, u_n)$  on the diagonal and zeros elsewhere. Let  $\mathbf{x}_{(j)}$  be the  $j$ th column of  $\mathbf{X}$  and  $x_{ij}$  be the  $ij$ th element of  $\mathbf{X}$ . Denote  $\Gamma_{uj} = (\mathbf{x}_{(j)}, \mathbf{U}_u \mathbf{x}_{(j)})$  for  $1 \leq j \leq p$  and  $\Gamma_u = (\Gamma_{u1}, \dots, \Gamma_{up})$  to be a  $n \times p$  matrix. Define  $\mathbf{Y} = (Y_1, \dots, Y_n)^\top$  and  $W_u = \text{diag}(K_h(U_1 - u)/K_h(0), \dots, K_h(U_n - u)/K_h(0))$ . Using this notations, we can write (2.2) as

$$\min_{\boldsymbol{\gamma}} (\mathbf{Y} - \Gamma_u \boldsymbol{\gamma})^\top W_u (\mathbf{Y} - \Gamma_u \boldsymbol{\gamma}). \quad (2.3)$$

This notation then has a formulation of a weighted least square problem.

### 2.2.1 Choosing the bandwidth $h$

The standard approach to choose the bandwidth is based on the trade-off between the bias and variance. The simplest way is the rule of thumb, see [13] for details. It is fast in computation, however, it highly depends on the asymptotic expansion of the bias and variance, and may not work well in small samples. Moreover, the optimal bandwidth is based on several unknown quantities, for which good estimates may be difficult to obtain. To overcome these deficiencies, we adopt the mean square error (MSE) tuning method. For a detailed review, see [19] and [59]. This method uses information provided with finite samples and hence carries more information about the finite sample, which selects the bandwidth more accurately than other methods such as residual squares criterion (RSC) [12] and cross validation (CV).

The  $MSE(h)$  for a fixed smoothing bandwidth  $h$  is defined as

$$MSE(h) = E\{\mathbf{x}^\top \hat{\mathbf{a}}(U) - \mathbf{x}^\top \mathbf{a}(U)\}^2.$$

By direct calculation, we have

$$MSE(h) = E[\mathbf{B}^\top(U)\Omega(U)\mathbf{B}(U) + \text{tr}\{\Omega(U)V(U)\}], \quad (2.4)$$

where  $\mathbf{B}(U) = \text{Bias}(\hat{\mathbf{a}}(U))$ ,  $\Omega(U) = E(\mathbf{xx}^\top | U)$  and  $V(U) = \text{Cov}(\hat{\mathbf{a}}(U))$ . To estimate  $MSE(h)$ , we need to estimate  $\mathbf{B}(U)$ ,  $\Omega(U)$  and  $V(U)$ . Then we have

$$\widehat{MSE}(h) = n^{-1} \sum_{i=1}^n [\hat{\mathbf{B}}^\top(U_i) \hat{\Omega}(U_i) \hat{\mathbf{B}}(U_i) + \text{tr}\{\hat{\Omega}(U_i) \hat{V}(U_i)\}].$$

A grid search can be done to select the optimal bandwidth  $h$  which minimizes the  $\widehat{MSE}(h)$ .

The estimation of  $\mathbf{B}(U)$ ,  $\Omega(U)$  and  $V(U)$  was discussed in [20], and we give a brief review here. For each given  $u$ , we have  $\Omega(u) = E(\mathbf{xx}^\top | u)$ , and we can estimate this quantity by a

kernel smoother

$$\hat{\Omega}(u) = \frac{\sum_{i=1}^n \mathbf{x}_i \mathbf{x}_i^\top K_h(U_i - u)}{\sum_{i=1}^n K_h(U_i - u)}$$

Introduce the  $p \times 2p$  matrix  $M$ , where the  $(j, 2j - 1)$  ( $1 \leq j \leq n$ ) elements of  $M$  are 1, and the remaining elements are 0. [20] summarized the forms of the bias and variance. For the estimated bias, we have

$$\hat{\mathbf{B}}(u) = \hat{\text{bias}}(\hat{\mathbf{a}}(u)) = M(\Gamma_u^\top W_u \Gamma_u)^{-1} \Gamma_u^\top W_u \hat{\boldsymbol{\tau}},$$

where the  $i$ th element of  $\hat{\boldsymbol{\tau}}$  is

$$2^{-1} \mathbf{x}_i^\top \{ \hat{\mathbf{a}}^{(2)}(u)(U_i - u)^2 + 3^{-1} \hat{\mathbf{a}}^{(3)}(u)(U_i - u)^3 \}.$$

The preceding representation involves two unknown quantities  $\hat{\mathbf{a}}^{(2)}(u)$  and  $\hat{\mathbf{a}}^{(3)}(u)$ , which can be estimated by local cubic fitting with an appropriate pilot bandwidth  $h_*$ .

The estimated variance is given by

$$\hat{V}(u) = \hat{\text{cov}}(\hat{\mathbf{a}}(u)) = M(\Gamma_u^\top W_u \Gamma_u)^{-1} (\Gamma_u^\top W_u^2 \Gamma_u) (\Gamma_u^\top W_u \Gamma_u)^{-1} M^\top \hat{\sigma}^2(u).$$

The estimator  $\hat{\sigma}^2(u)$  can be obtained as a byproduct when we use local cubic fitting with a pilot bandwidth  $h_*$ . Denote  $\Gamma_{u_j}^* = (\mathbf{x}_{(j)}, \mathbf{U}_u \mathbf{x}_{(j)}, \mathbf{U}_u^2 \mathbf{x}_{(j)}, \mathbf{U}_u^3 \mathbf{x}_{(j)})$  and  $\Gamma_u^* = (\Gamma_{u_1}^*, \dots, \Gamma_{u_p}^*)$ . We have

$$\hat{\sigma}^2(u) = \frac{Y^\top \{ W_u^* - W_u^* \Gamma_u^* (\Gamma_u^{*\top} W_u^* \Gamma_u^*)^{-1} \Gamma_u^{*\top} W_u^* \} Y}{\text{tr} \{ W_u^* - (\Gamma_u^{*\top} W_u^* \Gamma_u^*)^{-1} (\Gamma_u^{*\top} W_u^{*2} \Gamma_u^*) \}}$$

where  $W_u^*$  is  $W_u$  with  $h$  replaced by  $h_*$ .

For the pilot bandwidth  $h_*$ , which is used for a pilot local cubic fitting, [12] introduced the RSC to select it. However, they only studied the univariate case, which applies to the varying coefficient model with one component. As we are considering the varying coefficient model with several components, their method is not applicable. Instead, we use five-fold cross validation to do the pilot fitting.

## 2.3 Penalized local polynomial regression estimation

In practice, it would be of interest to detect the nonzero region of each component of the vector  $\mathbf{a}$ . To achieve this goal, shrinkage methods can be applied. Notice that if  $a_j(u) = 0$  for  $u \in [c_1, c_2]$ , then  $a'_j(u) = 0$  for  $u \in [c_1, c_2]$ , which indicates that if the function estimates are zero

over certain regions, the derivative should also behave so. Consequently, we treat each  $(a_j, b_j)$  ( $1 \leq j \leq p$ ) as a separate group and do penalization together. To achieve the variable selection as well as estimation accuracy, a group SCAD penalty [53] is then added to (2.2) to get sparse solutions for  $\mathbf{a}$  and  $\mathbf{b}$ .

Recall that we need to solve the minimization problem (2.3), which can be rewritten as a least square problem with new data  $(W_u^{1/2}\mathbf{Y}, W_u^{1/2}\Gamma_u)$ :

$$\min_{\gamma} (W_u^{1/2}\mathbf{Y} - W_u^{1/2}\Gamma_u\gamma)^\top (W_u^{1/2}\mathbf{Y} - W_u^{1/2}\Gamma_u\gamma).$$

For a traditional linear model, the covariates would be scaled first before adding the penalty. A similar procedure would be applied by making each column of the covariate  $W_u^{1/2}\Gamma_u$  have the same variance. Denote  $s_j$  as the standard deviation for the pseudo covariates  $x_{ij}(K_h(U_i - u)/K_h(0))^{1/2}$  ( $1 \leq i \leq n$ ) and  $r_j$  as the standard deviation for the pseudo covariates  $x_{ij}(U_i - u)(K_h(U_i - u)/K_h(0))^{1/2}$  ( $1 \leq i \leq n$ ). In other words,  $(s_1, r_1, s_2, r_2, \dots, s_p, r_p)^\top$  are the standard deviations for each column of the pseudo covariate  $W_u^{1/2}\Gamma_u$ . We would ideally standardize the covariates and do the penalization. In fact, this is equivalent to keeping the covariates the same and change the penalty by reparameterization.

In typical situations, this rescaling is needed only for finite sample behavior. However, in this case, the convergence rates for function estimation,  $\hat{a}$ , and derivative estimation,  $\hat{b}$  are of different orders of magnitude. Hence this rescaling is also necessary asymptotically. We have shown in Lemma 1 in the Appendix that  $s_j = O_P(1)$  and  $r_j = O_P(h)$ , which properly adjusts the effect of the different rates of convergence of the function and derivative estimates.

For the local polynomial regression, it is no longer appropriate to use  $n$  as the sample size, as not all observations contribute equally to a given location. In fact, some will contribute nothing if the kernel has bounded support. Consequently, we define the effective sample size as  $m = \frac{\sum_{i=1}^n K_h(U_i - u)}{K_h(0)}$ . The penalized local polynomial regression estimates  $(\hat{\mathbf{a}}_\lambda^\top, \hat{\mathbf{b}}_\lambda^\top)^\top$  can be found by minimizing

$$\min_{\mathbf{a}, \mathbf{b}} \left[ \sum_{i=1}^n \{Y_i - \mathbf{x}_i^\top \mathbf{a} - \mathbf{x}_i^\top \mathbf{b}(U_i - u)\}^2 (K_h(U_i - u)/K_h(0)) + m \sum_{j=1}^p P_\lambda(\sqrt{s_j^2 a_j^2 + r_j^2 b_j^2}) \right], \quad (2.5)$$

where the SCAD penalty function [15] is a symmetric function at zero which satisfies  $P_\lambda(0) = 0$  and its first order derivative is defined as

$$P'_\lambda(t) = \lambda \{I(t \leq \lambda) + \frac{(a\lambda - t)_+}{(a-1)\lambda} I(t > \lambda)\} \text{ for some } a > 2 \text{ and } t > 0$$

For any point  $u$  in the domain of  $U$ , we can obtain the estimates of  $\mathbf{a}(u)$  using our penalized local polynomial regression method. To detect the nonzero regions for some of the varying

coefficients, a set of dense grid is chosen over the whole domain of  $U$ , say  $(u_1, \dots, u_N)$ . We can obtain the estimates of  $\mathbf{a}(u)$  on these grid points. If the estimate of certain component function is zero for a certain number of consecutive grid points, for instance, say estimate of  $a_j(u)$  is zero on grid points  $\{u_{l_1}, u_{l_1+1}, \dots, u_{l_2}\}$ . We would say the estimate of the function  $a_j(u)$  is zero over the domain  $[u_{l_1}, u_{l_2}]$ .

We will introduce some notations which would be used in the following sections. Denote  $\beta_j = (s_j a_j, r_j b_j)^\top$  and  $\beta = (\beta_1^\top, \dots, \beta_p^\top)^\top$ . Let  $\beta_0$  be the true value of  $\beta$ . Denote  $\hat{\beta} = (\hat{\beta}_1^\top, \dots, \hat{\beta}_p^\top)^\top$  to be the local polynomial estimates of  $\beta_0 = (\beta_{10}^\top, \dots, \beta_{p0}^\top)^\top$ , and  $\hat{\beta}_\lambda = (\hat{\beta}_{\lambda 1}^\top, \dots, \hat{\beta}_{\lambda p}^\top)^\top$  to be the penalized local polynomial estimates when the regularization parameter is  $\lambda$ .

### 2.3.1 Algorithms

We would discuss how to solve (2.5) in this subsection. For the SCAD penalization problem, [15] proposed the local quadratic approximation(LQA) algorithm to get the minimizer of the penalized loss function. With LQA, the optimization problem can be solved using a modified Newton-Raphson algorithm. The LQA estimator, however, cannot achieve sparse solutions. A threshold would then be used to shrink the small coefficients to zero. To remedy this issue, [63] proposed a new approximation method based on local linear approximation (LLA). The advantage of the LLA algorithm is that it inherits the computational efficiency of LASSO. Denote  $\beta_j^{(k)} = (s_j a_j^{(k)}, r_j b_j^{(k)})^\top$ . Given the estimate  $\{\hat{\beta}_j^{(k)}, j = 1, \dots, p\}$  at the  $k$ th iteration, we minimize

$$\min_{a,b} \left[ \sum_{i=1}^n \{Y_i - \mathbf{x}_i^\top \mathbf{a} - \mathbf{x}_i^\top \mathbf{b}(U_i - u)\}^2 (K_h(U_i - u)/K_h(0)) + m \sum_{j=1}^p w_j^{(k)} \|\beta_j\| \right],$$

to get  $\{\hat{\beta}_j^{(k+1)}, j = 1, \dots, p\}$ , where  $w_j^{(k)} = |P'_\lambda(\|\hat{\beta}_j^{(k)}\|)|$ . Repeat the iterations until convergence, and the limit would be the minimizer. The initial value of  $\hat{\beta}_j^{(0)}$  can be chosen as the unpenalized local polynomial estimates. We have found that one step estimates already perform very well, so it is not necessary to iterate further, see [63] for similar discussions. Consequently, the one step estimate is adopted because it can save the computational time.

### 2.3.2 Tuning of the regularization parameter

When tuning the regularization parameter  $\lambda$ , we adopt the Bayesian information criterion (BIC). Suppose  $\hat{\mathbf{a}}_\lambda$  and  $\hat{\mathbf{b}}_\lambda$  are the solutions of the optimization problem for a fixed  $\lambda$ . The BIC is given by

$$BIC(\lambda) = m \log\left(\frac{RSS(\lambda)}{m}\right) + \log m * df,$$

where  $RSS(\lambda) = \sum_{i=1}^n \{Y_i - \mathbf{x}_i^\top \hat{\mathbf{a}}_\lambda - \mathbf{x}_i^\top \hat{\mathbf{b}}_\lambda(U_i - u)\}^2 (K_h(U_i - u)/K_h(0))$ . The degrees of freedom (df) are given as  $\sum_{j=1}^p I(\|\hat{\boldsymbol{\beta}}_{\lambda_j}\| > 0) + \sum_{j=1}^p \frac{\|\hat{\boldsymbol{\beta}}_{\lambda_j}\|}{\|\hat{\boldsymbol{\beta}}_j\|} (d_j - 1)$ , where  $d_j = 2$  as we use local linear polynomial regression, see [57].

## 2.4 Asymptotic properties

In this subsection, we investigate the theoretical properties of our estimator. We begin with some notations. Without loss of generality, we assume the first  $2s$  components of  $\boldsymbol{\beta}$  are nonzero. Define  $\boldsymbol{\beta}_N = (\boldsymbol{\beta}_1^\top, \dots, \boldsymbol{\beta}_s^\top)^\top$  and  $\boldsymbol{\beta}_Z = (\boldsymbol{\beta}_{s+1}^\top, \dots, \boldsymbol{\beta}_p^\top)^\top$ . Denote  $\Gamma_{hu_j} = (\mathbf{x}_{(j)}/s_j, \mathbf{U}_u \mathbf{x}_{(j)}/r_j)$  for  $1 \leq j \leq p$  and  $\Gamma_{hu} = (\Gamma_{hu1}, \dots, \Gamma_{hup})$ . Recall that  $m = \frac{\sum_{i=1}^n K_h(U_i - u)}{K_h(0)}$ , which is the effective sample size. Our objective function can be written as

$$Q(\boldsymbol{\beta}) = (\mathbf{Y} - \Gamma_{hu} \boldsymbol{\beta})^\top W_u (\mathbf{Y} - \Gamma_{hu} \boldsymbol{\beta}) + m \sum_{j=1}^p P_{\lambda_n}(\|\boldsymbol{\beta}_j\|).$$

Denote  $b_n = (nh)^{-1/2}$ . We first state the following conditions:

Conditions (A)

(A1) The bandwidth satisfies  $nh \rightarrow \infty$  and  $n^{1/7}h \rightarrow 0$ .

(A2) Denote  $a_n = \max_{1 \leq j \leq s} P'_{\lambda_n}(\|\boldsymbol{\beta}_{j0}\|)$ , we have  $a_n^2 nh \rightarrow 0$ .

Condition (A1) is a condition on the bandwidth of the local polynomial regression, which indicates the bias will dominate the variance when estimating the functions and their derivatives. Condition (A2) is a condition on the penalty function and the strength of the true signals, which can be written as  $a_n = o(b_n)$ .

For the SCAD penalty function, we have  $\max_{1 \leq j \leq s} |P''_{\lambda_n}(\|\boldsymbol{\beta}_{j0}\|)| \rightarrow 0$  when  $n \rightarrow \infty$  and  $\liminf_{n \rightarrow \infty} \liminf_{\theta \rightarrow 0^+} P'_{\lambda_n}(\theta)/\lambda_n > 0$ . Moreover, as  $a_n \leq \lambda_n$ , by condition (A2), we have  $\lambda_n^2 nh \rightarrow 0$ , which indicates  $b_n = o(\lambda_n)$ . These results would be used in the proof of our paper, which we refer the readers to the Appendix A.

Under Conditions (A), if  $\lambda_n \rightarrow 0$  as  $n \rightarrow \infty$ , we would have the following theorems and corollary:

**Theorem 1**  $\hat{\boldsymbol{\beta}}_\lambda - \boldsymbol{\beta}_0 = O_P(b_n)$ .

Theorem 1 gives the consistency rate for our penalized estimates. By theorem 1 and Lemma 1 in the Appendix, we have  $\hat{a}_{\lambda_j} - a_{j0} = O_P(b_n)$  and  $\hat{b}_{\lambda_j} - b_{j0} = O_P(b_n/h)$  for any  $1 \leq j \leq p$ , which indicates that our penalized estimates have  $b_n$  consistency rate while the derivative estimates have  $b_n/h$  consistency.

**Theorem 2** *With probability tending to 1, for any  $\boldsymbol{\beta}_N$  satisfying  $\|\boldsymbol{\beta}_N - \boldsymbol{\beta}_{N0}\| = O_P(b_n)$  and*

any constant  $C$ ,

$$Q(\boldsymbol{\beta}_N^\top, \mathbf{0}) = \max_{\|\boldsymbol{\beta}_Z\| \leq Cb_n} Q(\boldsymbol{\beta}_N^\top, \boldsymbol{\beta}_Z^\top).$$

Theorem 2 indicates that we can capture the true zero components with probability going to 1.

Denote  $\Sigma_s$  the upper left corner  $2s \times 2s$  submatrix of  $\Sigma$ . Let  $T(\boldsymbol{\beta}_l)$  be a matrix function

$$\begin{aligned} T(\boldsymbol{\beta}_l) &= \frac{\partial [P'_{\lambda_n}(\|\boldsymbol{\beta}_l\|) \frac{\boldsymbol{\beta}_l}{\|\boldsymbol{\beta}_l\|}]}{\partial \boldsymbol{\beta}_l^\top} \\ &= P''_{\lambda_n}(\|\boldsymbol{\beta}_l\|) \frac{\boldsymbol{\beta}_l \boldsymbol{\beta}_l^\top}{\|\boldsymbol{\beta}_l\|^2} + P'_{\lambda_n}(\|\boldsymbol{\beta}_l\|) \frac{\boldsymbol{\beta}_l \boldsymbol{\beta}_l^\top}{\|\boldsymbol{\beta}_l\|^3} + \frac{P'_{\lambda_n}(\|\boldsymbol{\beta}_l\|)}{\|\boldsymbol{\beta}_l\|} I_2, \end{aligned}$$

where  $\boldsymbol{\beta}_l$  is a two dimensional vector

Denote  $H = \text{diag}(T(\boldsymbol{\beta}_{10}), \dots, T(\boldsymbol{\beta}_{s0}))$ , which is a  $2s \times 2s$  matrix, and  $\mathbf{d} = (P'_{\lambda_n}(\|\boldsymbol{\beta}_{10}\|) \frac{\boldsymbol{\beta}_{10}^\top}{\|\boldsymbol{\beta}_{10}\|}, \dots, P'_{\lambda_n}(\|\boldsymbol{\beta}_{s0}\|) \frac{\boldsymbol{\beta}_{s0}^\top}{\|\boldsymbol{\beta}_{s0}\|})^\top$  which is a  $2s$  dimensional vector.

**Theorem 3** Suppose  $\hat{\boldsymbol{\beta}}_N$  is the local polynomial estimator of the nonzero components, and  $\hat{\boldsymbol{\beta}}_{\lambda N}$  is the penalized local polynomial estimator for the nonzero components. We have

$$\hat{\boldsymbol{\beta}}_N - \boldsymbol{\beta}_{N0} = \Sigma_s^{-1} \left( \frac{m}{2n} H + \Sigma_s \right) [(\hat{\boldsymbol{\beta}}_{\lambda N} - \boldsymbol{\beta}_{N0}) + \left( \frac{m}{2n} H + \Sigma_s \right)^{-1} \frac{m}{2n} \mathbf{d}] + o_P(n^{-1}).$$

**Corollary 1** From theorem 3, we can get

$$\text{cov}^{-1/2}(\hat{\mathbf{a}}_N(u)) \{ \hat{\mathbf{a}}_{\lambda N}(u) - \mathbf{a}_{N0}(u) - \text{bias}(\hat{\mathbf{a}}_N(u)) \} \xrightarrow{d} N(0, I_s)$$

where  $\hat{\mathbf{a}}_{\lambda N}(u)$  denotes our penalized local polynomial estimates for the nonzero functions.

This corollary shows that when  $n \rightarrow \infty$ , the asymptotic distribution of  $\hat{\mathbf{a}}_N - \mathbf{a}_{N0}$  and  $\hat{\mathbf{a}}_{\lambda N} - \mathbf{a}_{N0}$  are approximately the same (the distribution of  $\hat{\mathbf{a}}_N - \mathbf{a}_{N0}$  is given in Lemma 3 in the Appendix), which indicates the oracle property. The distribution of  $\hat{\mathbf{a}}_{\lambda N} - \mathbf{a}_{N0}$  is asymptotic normal after adjusting the bias and variance.

## 2.5 Simulation example

Simulation studies are conducted to examine the performance of our penalized local polynomial regression approach and compare with the local polynomial regression method.

### 2.5.1 Example 1

We consider the univariate case here, where the data is simulated from the model  $Y = xa_1(u) + \epsilon$  with  $\epsilon \sim N(0, 1)$ . The true function  $a_1(u)$  is defined as

$$a_1(u) = \begin{cases} 50(u - 0.3)^3 & \text{if } 0 \leq u \leq 0.3 \\ 50(u - 0.7)^3 & \text{if } 0.7 \leq u \leq 1 \\ 0 & \text{otherwise,} \end{cases}$$

the plot of which is in panel (a) of Figure 2.1.

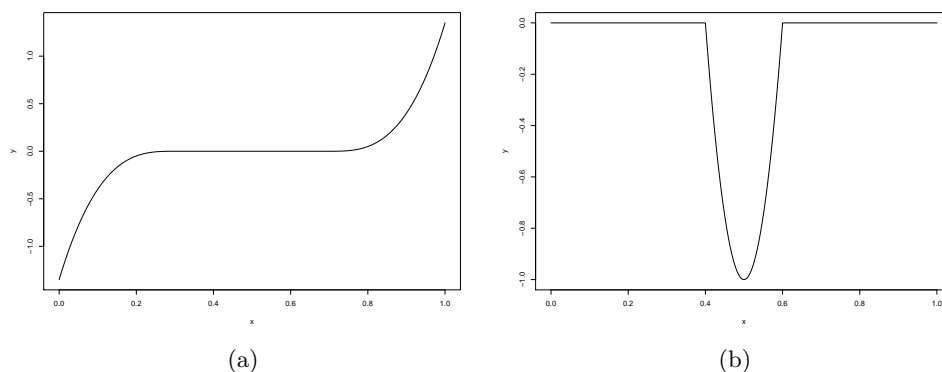


Figure 2.1: Plots of  $a_1(u)$  (left) and  $a_2(u)$  (right) for simulation example.

The covariates  $x_i$  are generated iid from  $N(0, 4)$ . The data points  $U_i$  are chosen as  $n$  equally spaced design points on  $[0, 1]$ . The sample sizes are varied to be  $n = 100, 200, 500$  in the simulations. For estimation, we fix 501 equally spaced grid points on  $[0, 1]$  and fit the penalized local polynomial regression on each point and get the estimates. We also run local polynomial regression on these grid points to make comparisons.

To examine the performance, we run 100 repetitions of Monte Carlo studies. As the zero region for the true function  $a_1(u)$  is  $[0.3, 0.7]$ , define the correct zero coverage (correctzero) as the proportion of region in  $[0.3, 0.7]$  that is estimated as zero. The mean of correctzero in 100 repetitions will be reported. Moreover, we report the mean square error of the penalized local polynomial regression, which is defined as  $\int_0^1 (\hat{a}_{1\lambda}(u) - a_1(u))^2 du$ . The mean square error of the original local polynomial regression is also reported, which is  $\int_0^1 (\hat{a}_1(u) - a_1(u))^2 du$ . These MSEs are calculated using numerical integrations based on the estimates on the 501 equally spaced grid points. The MSEs are used to identify how well we estimate the nonparametric

function. In addition, we generate an independent testing data set, which contains  $N = 501$  groups of independent triples  $(\tilde{U}_i, \tilde{x}_i, \tilde{Y}_i)$ . The time points  $U_i$  are chosen as 501 equally spaced points on  $[0, 1]$ . For  $x_i$  and  $\epsilon_i$  in the testing data set, they are randomly generated from the same distributions as the training data set. Define the estimation error (EE) for the entire model as

$$EE = \frac{\sum_{i=1}^N (\tilde{x}_i a_1(\tilde{U}_i) - \tilde{x}_i \tilde{a}_1(\tilde{U}_i))^2}{N}.$$

The estimation error of penalized local polynomial regression and local polynomial regression would be calculated, where we let  $\tilde{a}_1 = \hat{a}_{1\lambda}$  and  $\tilde{a}_1 = \hat{a}_1$  respectively. The mean of the these estimation errors will be reported, which are used to reflect the prediction error. The results for Example 1 are in Table 2.1.

Table 2.1: Simulation results for the univariate case using penalized local polynomial regression and original local polynomial regression when sample size varies from  $n = 100, 200, 500$ . The entries in the table denotes mean square error (MSE) of the estimated function, the correct zero coverage (correctzero) and the estimation error (EE) for the entire model.

Sample size	Method	MSE	correctzero	EE
$n = 100$	Penalized	0.0230(0.0023)	0.9855 (0.007)	0.077 (0.007)
	Original	0.0217(0.0013)	-	0.083 (0.005)
$n = 200$	Penalized	0.0088(0.0008)	0.9934 (0.004)	0.031 (0.003)
	Original	0.0110(0.0005)	-	0.043 (0.002)
$n = 500$	Penalized	0.0033(0.0002)	0.9943 (0.003)	0.012 (0.001)
	Original	0.0053(0.0002)	-	0.021 (0.001)

## 2.5.2 Example 2

Next, we consider the bivariate case, where the data is simulated from the model  $Y = x_1 a_1(u) + x_2 a_2(u) + \epsilon$  with  $\epsilon \sim N(0, 1)$ . The first function  $a_1(u)$  is the same function used in simulation 1. The second component function  $a_2(u)$  is defined as

$$a_2(u) = \begin{cases} 100((u - 0.5)^2 - 0.01) & \text{if } 0.4 \leq u \leq 0.6 \\ 0 & \text{otherwise,} \end{cases}$$

the plot of which is in panel (b) of Figure 2.1. It can be seen that  $a_2(u)$  is not differentiable at point 0.4 and 0.6.

For the design points  $U_i$  and the noise  $\epsilon_i$ , the settings are the same as simulation 1. For the covariates  $\mathbf{x}_i$ , we generate iid from  $N(\mathbf{0}, 4I_2)$ , where  $I_2$  is the  $2 \times 2$  identity matrix. The sample sizes are still set as  $n = 100, 200, 500$ . For estimation, the penalized local polynomial regression

is fitted on 501 equally spaced grid points on  $[0, 1]$ . An independent testing data set with size  $N = 501$  is generated. The time points  $\tilde{U}_i$  are still chosen as 501 equally spaced points on  $[0, 1]$ , and  $\tilde{x}_{i1}$ ,  $\tilde{x}_{i2}$  and  $\tilde{\epsilon}_i$  are generated the same as the training dataset. For the estimation error of the entire model, it is defined as

$$EE = \frac{\sum_{i=1}^N (\tilde{x}_{i1}a_1(\tilde{U}_i) + \tilde{x}_{i2}a_2(\tilde{U}_i) - \tilde{x}_{i1}\tilde{a}_1(\tilde{U}_i) - \tilde{x}_{i2}\tilde{a}_2(\tilde{U}_i))^2}{N}.$$

The mean of MSE and correctzero for both two functions  $a_1(\cdot)$  and  $a_2(\cdot)$  will be reported. The mean of the estimation errors for the entire model is also reported. All the results are summarized in Table 2.2.

Table 2.2: Simulation results for the multivariate case using penalized local polynomial regression and original local polynomial regression when sample size varies from  $n = 100, 200, 500$ . The entries in the table denotes mean square error of the estimated function  $a_1(\cdot)$  (MSE (function 1)) and  $a_2(\cdot)$  (MSE (function 2)), the correct zero coverage (correctzero) for  $a_1(\cdot)$  (correctzero(function 1)) and  $a_2(\cdot)$  (correctzero(function 2)), and the estimation error (EE) for the entire model.

Sample size	Method	MSE (function 1)	correctzero(function 1)	MSE(function 2)	correctzero(function 2)	EE
$n = 100$	Penalized	0.0266(0.0025)	0.9386 (0.012)	0.0482(0.0026)	0.7633 (0.013)	0.226 (0.012)
	Original	0.0240(0.0013)	–	0.0456(0.0015)	–	0.251 (0.008)
$n = 200$	Penalized	0.0125(0.0008)	0.9471 (0.013)	0.0241(0.0014)	0.8459 (0.009)	0.111 (0.005)
	Original	0.0117(0.0005)	–	0.0251(0.0011)	–	0.134 (0.004)
$n = 500$	Penalized	0.0052(0.0004)	0.9707 (0.008)	0.0105(0.0004)	0.8862 (0.004)	0.048 (0.002)
	Original	0.0056(0.0003)	–	0.0117(0.0003)	–	0.062 (0.002)

From these two simulations, we can see that our methods perform better than the original local polynomial regression in the sense that it gives smaller estimation error, which indicates better prediction. Meanwhile, we can estimate the zero regions of each component function quite well. When sample size increases, our method can capture the correct zero regions more accurately.

## 2.6 Real data application

We apply our method to the Boston housing data [26]. The data set is based on the 1970 US census. It consists of the median value of owner-occupied homes for 506 census tracts in Boston area. The data set was analyzed by [14] when they studied the semiparametric varying-coefficient partially linear models. Similarly as [14], we treat the median value of the houses as the response. We consider five predictors: CRIM (per capita crime rate by town), RM (average number of rooms per dwelling), TAX (full-value property-tax rate per \$10000), NOX (nitric

oxides concentration parts per 10 million) and LSTAT (percentage of lower income status of the population), which might explain the variation of the housing value. The covariates CRIM, RM, TAX, NOX are denoted as  $\mathbf{x}_{(2)}, \dots, \mathbf{x}_{(5)}$ , respectively. We set  $\mathbf{x}_{(1)} = \mathbf{1}$  to include the intercept term. We are interested in studying the association between the median value of the owner-occupied homes and these four covariates. As the distribution of LSTAT is asymmetric, similar as [14], the square root transformation is employed to make the resulting distribution symmetric. The histogram for the distribution of the  $\sqrt{\text{LSTAT}}$  is plotted in Figure 2.2. Specifically, we treat

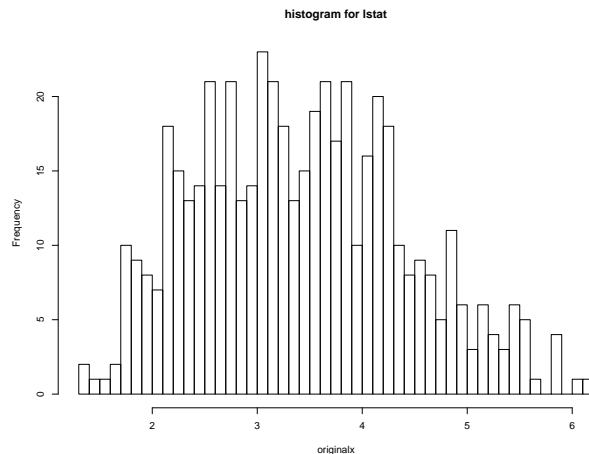


Figure 2.2: Histogram for  $\sqrt{\text{LSTAT}}$ .

$U = \sqrt{\text{LSTAT}}$  similarly as [14]. We construct the following model

$$\mathbf{Y} = \sum_{j=1}^5 a_j(U) \mathbf{x}_{(j)} + \epsilon.$$

When dealing with the data, we center the response first. We also center and scale the covariates  $\mathbf{x}_{(2)}, \dots, \mathbf{x}_{(5)}$ . As  $a_1(\cdot)$  is the intercept function, this term is not penalized. We only penalize the function  $a_2(\cdot), a_3(\cdot), a_4(\cdot), a_5(\cdot)$ . We solve

$$\begin{aligned} & \min_{\mathbf{a}, \mathbf{b}} \sum_{i=1}^n \{Y_i - \mathbf{x}_i^\top \mathbf{a} - \mathbf{x}_i^\top \mathbf{b}(U_i - u)\}^2 (K_h(U_i - u) / K_h(0)) \\ & + m \sum_{j=2}^5 P_\lambda(\sqrt{s_j^2 a_j^2 + r_j^2 b_j^2}), \end{aligned}$$

where  $\mathbf{a} = (a_1, a_2, a_3, a_4, a_5)^\top$  and  $\mathbf{b} = (b_1, b_2, b_3, b_4, b_5)^\top$ . The Epanechnikov kernel is employed, and the bandwidth ( $h = 1.23$ ) is selected by the MSE tuning method. As we do not penalize the intercept term, we change the definition of degree of freedom used in tuning  $\lambda$  by  $\sum_{j=2}^5 I(\|\hat{\boldsymbol{\beta}}_{\lambda_j}\| > 0) + \sum_{j \in S} \frac{\|\hat{\boldsymbol{\beta}}_{\lambda_j}\|}{\|\hat{\boldsymbol{\beta}}_j\|} (d_j - 1)$ . The estimated functions using our penalized local polynomial regression ( $a_1(\cdot), a_2(\cdot), a_3(\cdot), a_4(\cdot), a_5(\cdot)$ ) are plotted in panels (a)-(e) of Figure 2.3.

From Figure 2.3, we can see that the variable TAX has no effect on the response when  $U$  is between 2.6 to 4 and the variable NOX has no effect on the response when  $U$  is less than 2.6 or between 4.3 to 4.7.

We have also used the data to compare the prediction errors for both our penalized local polynomial regression and the original local polynomial regression. We randomly pick up 300 samples from the data as the training data and fit using both methods. After that, we use the remaining 206 data points as our test data, and we can get the prediction error given by  $\frac{1}{|S|} \sum_{i \in S} (y_i - \hat{y}_i)^2$ , where  $S$  denote the indices for the test data set. We repeat the above step for 100 times by choosing different random seeds, and calculate the mean prediction errors for both methods. We have found that the mean of our penalized prediction error 37.7(3.0) is smaller than that of the original local polynomial regression 39.5(3.1), which indicates that we achieve smaller prediction error by using our penalized local polynomial regression method.

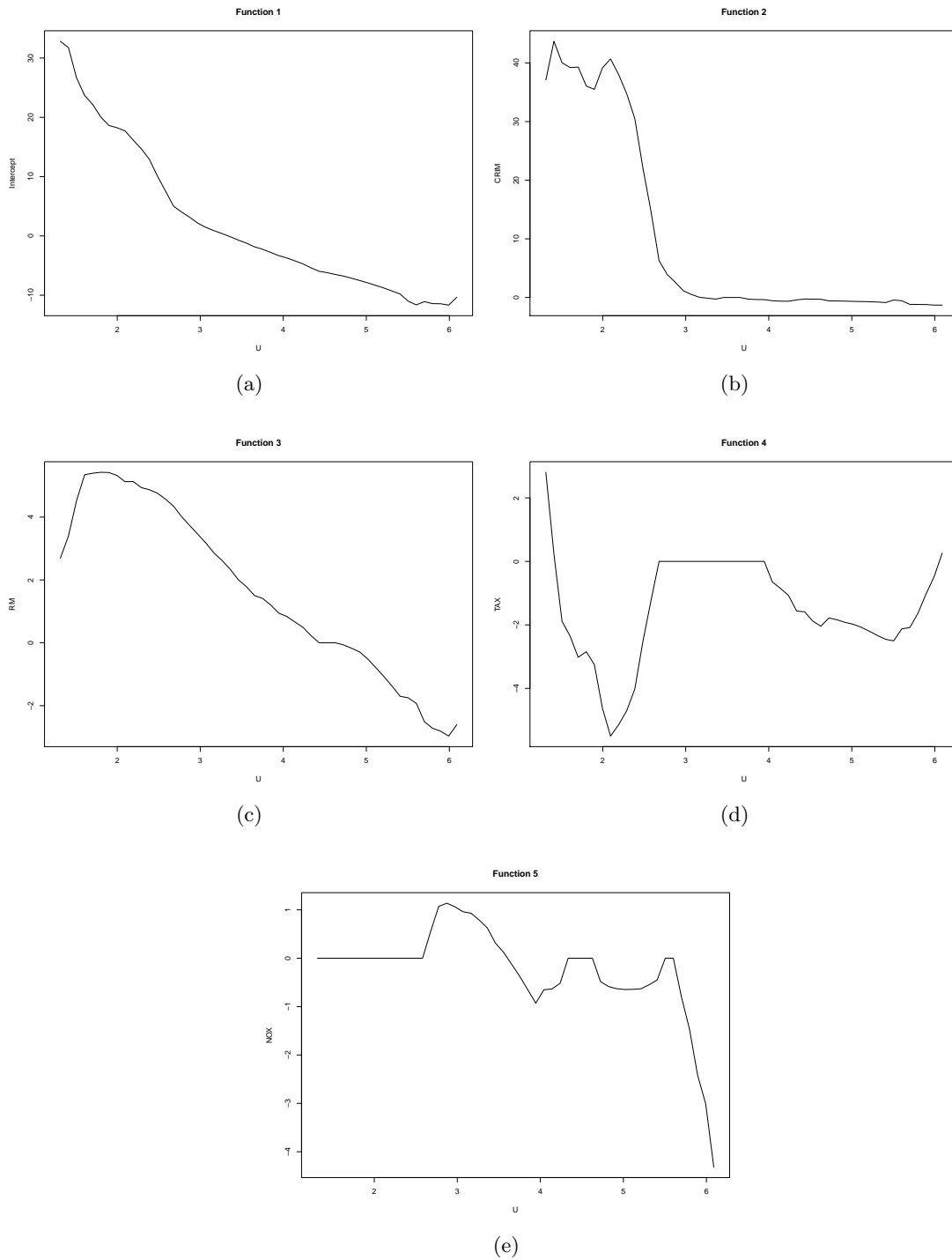


Figure 2.3: The penalized local polynomial estimates for the coefficient functions. Panels (a) (b) (c) (d) and (e) correspond to the estimates of the coefficient functions corresponding to the intercept, the variables CRIM, RM, TAX and NOX respectively.

## Chapter 3

# Fully efficient robust estimation, outlier detection and variable selection via penalized regression

### 3.1 Introduction

Outliers, which occur frequently in real data collection, are observations that deviate markedly from the rest of the observations. In the presence of outliers, likelihood-based inference can be unreliable, for example, the ordinary least squares (OLS) regression is very sensitive to outliers. To this end, outlier detection is critical in statistical learning because it can help to achieve robust statistical inference such as coefficient estimation, interval estimation and hypothesis testing. We consider the mean shift linear regression model  $y_i = \alpha + \mathbf{X}_i\boldsymbol{\beta} + \gamma_i + \epsilon_i$ , where  $\mathbf{X}_i$  is a  $p$  dimensional predictor,  $\boldsymbol{\beta}$  is a  $p$  dimensional parameter, and  $\gamma_i$  is a mean shift parameter which is nonzero when the  $i$ th observation is an outlier. This model was previously used by [22, 42, 48], and represents the general notion that the response can be arbitrary.

In this article, we are interested in variable selection as well as robust coefficient estimation together with the task of outlier detection based on this mean shift model. A popular method for variable selection is the penalized regression method such as LASSO [50], SCAD [16] and adaptive LASSO [61]. In fact, these penalized regression methods cannot only be used for variable selection but outlier detection as well. For example, [42] used an L1 regression while [48] imposed a nonconvex penalty function on  $\gamma_i$ 's to avoid the trivial estimate  $\hat{\gamma}_i = y_i$  and  $\hat{\boldsymbol{\beta}} = 0$  and achieve a sparse solution in terms of the shift parameter. If the estimate of  $\gamma_i$  is nonzero, the  $i$ th observation is identified as an outlier. However, as we will show, the breakdown point of the approach proposed by [48] is at most  $1/(p+1)$  while for [42], it is  $1/n$ . Hence, both methods are not robust to even a small number of outliers.

In the literature, asymptotic efficiency and breakdown point are two criteria to evaluate a robust regression technique. They represent the typical trade-off in efficiency for robustness. It is ideal to achieve full asymptotic efficiency compared to OLS while maintaining high breakdown point of  $1/2$ . Typical robust regression methods do not enjoy these two properties simultaneously. The OLS, which is fully efficient, only has a breakdown point of  $1/n$ , and hence even a single outlier can render the estimate arbitrarily bad. The M-estimates [33] also have a breakdown point of  $1/n$  while Mallow Generalized M-estimates [40] can have a breakdown point of only  $1/(p+1)$  [41, 10]. Moreover, neither of these two methods enjoys the full efficiency. There are several methods which enjoy high breakdown point of  $1/2$ , such as the Least Median of Squares (LMS) estimates [25, 46], the Least Trimmed Squares (LTS) estimates [46], S-estimates [44], MM-estimates [56] and the Schweppe one-step Generalized M-estimates [7], however these methods are not fully efficient. There have been some methods introduced achieving both properties, for example the Robust and Efficient Weighted Least Squares Estimators (REWLS) [23] and the generalized empirical likelihood method [2].

The proposed method achieves both full efficiency and high breakdown, while also performing variable selection simultaneously. Specifically, our method is robust to outliers and enjoys a high breakdown point that can be as high as  $1/2$ . Moreover, when there are no outliers, we show that applying our method is asymptotically equivalent to the penalized least squares for selection. In fact, if the regularization parameter is chosen appropriately, our estimator can enjoy full asymptotic efficiency under the gaussian distribution. Besides these properties, we also investigate the outlier detection consistency of our method, and show that under some regularity conditions, this method will correctly detect the outliers with probability tending to 1. In addition to these theoretical properties, we propose an efficient algorithm for our method, where the total number of unknown parameters,  $n+p$ , is larger than the sample size. The extended bayesian information criteria (EBIC) [4] is adopted to select the tuning parameters which control the outlier detection and variable selection respectively. Our method can also be extended to the high dimensional setting, where the dimension of the covariate  $p$  is diverging and much larger than the sample size. We show that the method still enjoys high breakdown even under the high dimension scenario.

## 3.2 Methodology

Denote  $\mathbf{y} = (y_1, \dots, y_n)^\top$ ,  $\mathbf{X} = (\mathbf{X}_1^\top, \dots, \mathbf{X}_n^\top)^\top$ ,  $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_n)^\top$  and  $\boldsymbol{\epsilon} = (\epsilon_1, \dots, \epsilon_n)^\top$ . Our model can be written as

$$\mathbf{y} = \alpha \mathbf{1} + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\gamma} + \boldsymbol{\epsilon},$$

where  $\alpha$  is the intercept,  $\mathbf{1}$  is a  $n \times 1$  vector with every element 1 and the error term  $\epsilon_i$ 's are i.i.d. with  $E(\epsilon_i) = 0$ . These mean shift parameters  $\gamma_i$ 's serve as indicators of the outliers in the regression of  $y_i|\mathbf{X}_i$ . If the  $i$ th subject is an outlier,  $\gamma_i \neq 0$ . Note that outliers may still occur in the covariate space, i.e. high leverage points, while having  $\gamma_i = 0$ , but we will show that these leverage points will not result in the breakdown of the estimator. We are interested in both outlier detection and variable selection for this model. To achieve these two goals, it is natural to devise a selection method via shrinkage. We impose penalties on  $\gamma_i$ 's to encourage them to shrink to zero and identify observations with nonzero  $\gamma_i$ 's as outliers. Meanwhile, we add penalties on the coefficient  $\beta$  to achieve variable selection. Specifically, we solve the following minimization problem

$$\min_{\alpha, \beta, \gamma} Q_n(\alpha, \beta, \gamma) = \min_{\alpha, \beta, \gamma} \|\mathbf{y} - \alpha\mathbf{1} - \mathbf{X}\beta - \boldsymbol{\gamma}\|_2^2 + \lambda_n \sum_{j=1}^p |\beta_j| + \mu_n \sum_{i=1}^n \frac{|\gamma_i|}{|\tilde{\gamma}_i|}, \quad (3.1)$$

where  $\tilde{\gamma}_i$ 's are obtained from the residuals of an initial robust regression fit. Here  $\lambda_n$  and  $\mu_n$  are different regularization parameters controlling the variable selection and outlier detection respectively.

### 3.2.1 Robust Initial Estimator

It is interesting to note that we impose an adaptive penalty on  $\boldsymbol{\gamma}$  which relies on the weight depending on an initial robust fit. The weight plays a similar role as the weight used in the adaptive LASSO problem [61], but it is based on the residuals rather than the parameter estimates. Various methods can be used for this initial step, for example LTS [46], LMS [25, 46], and MM-estimators [56], among others. The breakdown point of our method is at least as high as the breakdown point of the initial fit. In this article, we use the LTS method to obtain the initial robust estimates. We shall show that this will carry over to the high breakdown point of our estimator. Meanwhile, full efficiency and outlier detection consistency can be achieved by using this initial estimator. Theoretical properties of our estimators will be discussed in detail in Section 3. Denote  $r_i = |y_i - \mathbf{X}_i\beta|$ , the LTS method solves  $\min_{\beta} \sum_{i=1}^h r_{(i)}^2$ , where  $r_{(i)}$ 's are the order statistics of  $r_i$  with  $r_{(1)} \leq r_{(2)} \leq \dots \leq r_{(n)}$ . The number of included residuals  $h$ , is chosen to determine the breakdown point of the estimator. In particular, the breakdown point can be shown to be  $(n - h + 1)/n$ . In our simulation study and real data application, we use the truncation number  $h = \lfloor 3n/4 \rfloor$ , where  $\lfloor x \rfloor$  denotes largest integer less than  $x$ . For implementation, the R function "ltsReg" is adopted to obtain the initial estimates  $\tilde{\beta}$ . After we get the initial estimates  $\tilde{\beta}$ , the initial residuals are defined as  $\tilde{\gamma}_i = y_i - \mathbf{X}_i\tilde{\beta}$ .

### 3.2.2 Algorithm

The optimization problem (4.2) is an L1 penalized problem and it can be easily transformed to a quadratic programming problem. A more efficient way is to use the Least Angle Regression algorithm (LARS) [11]. Define  $\rho_n = \frac{\mu_n}{\lambda_n}$ , the optimization problem (4.2) becomes

$$\min_{\alpha, \beta, \gamma} Q_n(\alpha, \beta, \gamma) = \min_{\alpha, \beta, \gamma} \|\mathbf{y} - \alpha \mathbf{1} - \mathbf{X}\beta - \gamma\|_2^2 + \lambda_n \left\{ \sum_{j=1}^p |\beta_j| + \rho_n \sum_{i=1}^n \frac{|\gamma_i|}{|\tilde{\gamma}_i|} \right\}.$$

For a fixed  $\rho_n$ , we can do reparameterization  $\gamma_i^* = \frac{\rho_n \gamma_i}{|\tilde{\gamma}_i|}$  and the problem becomes

$$\min_{\alpha, \beta, \gamma} Q_n(\alpha, \beta, \gamma) = \min_{\alpha, \beta, \gamma} \|\mathbf{y} - \alpha \mathbf{1} - \mathbf{X}\beta - \mathbf{B}\gamma^*\|_2^2 + \lambda_n \left\{ \sum_{j=1}^p |\beta_j| + \sum_{i=1}^n |\gamma_i^*| \right\}, \quad (3.2)$$

with  $\mathbf{B} = \text{diag}(\frac{|\tilde{\gamma}_1|}{\rho_n}, \dots, \frac{|\tilde{\gamma}_n|}{\rho_n})$  and  $\gamma^* = (\gamma_1^*, \dots, \gamma_n^*)^\top$ . Problem (3.2) is a typical LASSO problem, and can be solved easily by R package “lars”, which indeed gives the whole solution path of (2) as a function of  $\lambda_n$ .

### 3.2.3 Tuning parameter selection

The optimization problem (4.2) involves tuning for two parameters  $\lambda_n$  and  $\mu_n$ , which is equivalent to tuning  $\lambda_n$  and  $\rho_n$  together. Since the number of parameters is  $n + p$  and larger than the sample size, we use the extended BIC (EBIC)[4] due to its nice asymptotic properties for high dimensional problems. Suppose  $\hat{\beta}$  and  $\hat{\gamma}$  are the estimates when the tuning parameters are set as  $\lambda_n$  and  $\rho_n$ . Let  $e_i^2 = (y_i - \hat{\alpha} - \mathbf{X}_i^* \hat{\beta} - \hat{\gamma}_i)^2$ , and define the residual sum of squares as  $RSS = \sum_{i=1}^n e_i^2$ . The EBIC is defined as

$$EBIC = n \log(RSS/n) + k \{ \log n + c \log(n + p) \},$$

where  $k$  is the degree of freedom (df) defined as the number of nonzero components of  $(\beta^\top, \gamma^\top)^\top$  and  $c$  is a constant that must be specified. In our case, we have  $p+n$  parameters which has order  $O(n)$ . By Theorem 1 of [4], when  $c > 1$  the EBIC can select the tuning parameter consistently if the number of parameters is on the order of  $n$ . Towards this end, we set  $c = 1 + \varepsilon$  with  $\varepsilon$  being a very small positive number to meet the requirement of their theoretical results. Based on our preliminary numerical experience, we have found that the results are not sensitive to the choice of small  $\varepsilon$ . Consequently, we set  $c = 1.01$  for convenience. We set two dimensional grids on  $\rho_n$  and  $\lambda_n$  to find the combination that minimizes the EBIC. Specifically, we first choose a set of dense grid of  $\rho_n$ , and for each  $\rho_n$ , as we mentioned in last subsection, we use LARS algorithm to obtain the solution paths of the problem (3.2). We pick the grid of  $\lambda_n$  on each point that the df

changes. For high dimensional problems with the number of parameters exceeding the sample size, we will get a perfect fit if the df is large enough, which would make the EBIC very small as RSS goes to zero. This results in the wrong selection of  $\lambda_n$  because it tends to select the  $\lambda_n$  that gives a perfect fit. Consequently, we only search over the  $\lambda_n$  which leads to  $k \leq \lfloor 0.5n \rfloor$  because we assume that the number of outliers is less than half of the sample size.

### 3.3 Theoretical results

In this section, we discuss some theoretical properties. We investigate some asymptotic results for the estimators in the first two subsections, and we consider the high breakdown point in the last subsection. Without loss of generality, we only show the results for the case when there is no intercept. The results as well as the proofs for the case with an intercept follows in a similar manner.

#### 3.3.1 Asymptotic theory when there are no outliers

We first consider the case when there are no outliers. We will show in this subsection that our methods can select important predictors consistently with probability tending to 1.

Since our method relies on the initial fit obtained from the least trimmed squares, more specifically, the residuals obtained from the least trimmed squares, we need some asymptotic results for the residuals, which are discussed in Lemma 1 and 2 in the Appendix. We start with the conditions that we need for our theorem and corollary.

Conditions (A)

(A1) For any  $1 \leq j \leq p$ ,  $\mathbf{X}_{ij}$ 's follow an independent and identically distributed (iid) distribution with  $E(\mathbf{X}_{ij}^2) < \infty$ .

(A2) The error  $\epsilon_i$ 's are iid with  $E(\epsilon_i^{2k}) < \infty$  for some  $k > 0$ .

(A2') The error  $\epsilon_i$ 's follow an iid subgaussian distribution.

Condition (A1) is a mild condition imposing boundness on the second moment of the covariates. Condition (A2) assumes that the random error has a finite  $2k$ th moment, which guarantees the polynomial tail bound. Condition (A2') is a stronger condition than (A2), which indicates that the error has an exponential tail bound.

We first discuss our main results for outlier detection consistency and variable selection consistency when no outlier exists. Without loss of generality, we assume that the first  $q$  components of  $\beta_0$  are nonzero, denoted by  $\beta_0(1)$ , and the remaining  $p - q$  components are zero, denoted by  $\beta_0(2) = \mathbf{0}$ . We first do some reparameterizations and let  $\eta = \gamma/\sqrt{n}$ . Define  $\theta = (\beta(1)^\top, \beta(2)^\top, \eta^\top)^\top = (\theta(1)^\top, \theta(2)^\top, \theta(3)^\top)^\top$  with  $\theta(1) = \beta(1)$ ,  $\theta(2) = \beta(2)$  and  $\theta(3) = \eta$ . Denote  $\mathbf{X}_{a,b}$  to be a submatrix consisting of the  $a$ th to  $b$ th column of the matrix  $\mathbf{X}$ .

The design matrix is defined as

$$\mathbf{A} = \begin{pmatrix} \mathbf{X}_{1,q} & \mathbf{X}_{q+1,p} & \sqrt{n}I_n \end{pmatrix}$$

and

$$C = \frac{1}{n} \mathbf{A}^\top \mathbf{A} = \begin{pmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{pmatrix}$$

with  $C_{11} = \frac{1}{n} \mathbf{X}_{1,q}^\top \mathbf{X}_{1,q}$ ,  $C_{21} = \frac{1}{n} \mathbf{X}_{q+1,p}^\top \mathbf{X}_{1,q}$  and  $C_{31} = \frac{1}{\sqrt{n}} \mathbf{X}_{1,q}$ .

Our method is equivalent to solving

$$\min_{\boldsymbol{\theta}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\boldsymbol{\theta}\|_2^2 + \lambda_n \sum_{j=1}^p |\boldsymbol{\theta}_j| + \sqrt{n}\mu_n \sum_{j=p+1}^{p+n} |\boldsymbol{\theta}_j| / |\tilde{\gamma}_{j-p}|$$

Suppose  $\mathbf{a}_1$  and  $\mathbf{a}_2$  are two column vectors with same dimension, denote  $\mathbf{a}_1 \leq \mathbf{a}_2$  if the inequality holds elementwise. Now we are ready to introduce the following conditions:

Condition (B)

(B1) The strong irrepresentable condition: there exists a positive constant vector  $\boldsymbol{\eta}$  such that  $|C_{21}C_{11}^{-1}\text{sign}(\boldsymbol{\theta}_0(1))| \leq \mathbf{1} - \boldsymbol{\eta}$ .

There exists  $0 < d \leq 1$  and  $M_1, M_2, M_3 > 0$  so that

(B2)  $\frac{1}{n} X_j^\top X_j \leq M_1$  for any  $1 \leq j \leq p$ , where  $X_j$  denotes the  $j$ th column of  $\mathbf{X}$ .

(B3)  $\boldsymbol{\alpha}^\top C_{11} \boldsymbol{\alpha} \geq M_2$  for any  $\|\boldsymbol{\alpha}\| = 1$ .

(B4)  $n^{\frac{1-d}{2}} \min_{j=1, \dots, q} |\beta_{j0}| \geq M_3$ .

Condition (B1) is introduced by [60] to guarantee the selection consistency for LASSO. Condition (B2) can be achieved by normalizing the covariates. Condition (B3) is trivial and only requires the smallest eigenvalue of the matrix  $C_{11}$  is nonzero. Condition (B4) quantifies the smallest signal of the coefficient  $\beta_{j0}$ , and we could identify the signal on the order of  $O(n^{\frac{d-1}{2}})$  for some  $0 < d \leq 1$ .

Define  $\mathbf{a}_1 =_s \mathbf{a}_2$  if the signs of these two vectors are the same elementwise. We have the following theorem:

**Theorem 1** *Under conditions (A1)(A2)(B), for  $\lambda_n = o(n^{(d+1)/2})$  and  $\lambda_n/\sqrt{n} \rightarrow \infty$ , and  $\mu_n n^{-1/2k-d/2-1/2} \rightarrow \infty$ , we have  $P(\hat{\boldsymbol{\theta}} =_s \boldsymbol{\theta}_0) \rightarrow 1$  as  $n \rightarrow \infty$ .*

Theorem 1 indicates that we can select the important predictors consistently when no outliers exist.

The condition can be relaxed if we assume the random error  $\epsilon_i$ 's have subgaussian distribution. We have the following corollary:

**Corollary 1** *Under conditions (A1)(A2')(B), for  $\lambda_n = Cn^{\frac{1+d_1}{2}}$  with a constant  $C$ ,  $0 < d_1 < d$  and  $\mu_n n^{-1/2-d_1/2}(\log n)^{-1/2} \rightarrow \infty$ , we have  $P(\hat{\boldsymbol{\theta}} =_s \boldsymbol{\theta}_0) \rightarrow 1$  as  $n \rightarrow \infty$ .*

**Remark:** In fact, if we do not impose any penalty on  $\boldsymbol{\beta}$ , i.e.  $\lambda_n = 0$ , which is the case of [48] by choosing the L-1 penalty, we have  $P(\hat{\boldsymbol{\gamma}} = \mathbf{0}) \rightarrow 1$ , which is equivalent to doing the least square regression. Thus, their estimator has full efficiency compared to the OLS. Even if we have the penalty on  $\boldsymbol{\beta}$  as in our case, we can impose some conditions on  $\lambda_n$  such that the LASSO estimator has full efficiency compared to the OLS, though these conditions do not guarantee the selection consistency of LASSO estimator. The theoretical justification of the arguments here can be revealed in similar manner.

### 3.3.2 High breakdown point

Let the  $n \times (p+1)$  matrix  $\mathbf{Z} = (\mathbf{X}, \mathbf{y})$  denote the sample, and  $\tilde{\mathbf{Z}}_m$  denotes the contaminated sample by replacing the  $m$  data points by arbitrary values. The finite sample breakdown point for the regression  $\hat{\boldsymbol{\beta}}$  is defined as

$$BP(\hat{\boldsymbol{\beta}}, \mathbf{Z}) = \min \left\{ \frac{m}{n} : \sup_{\tilde{\mathbf{Z}}_m} \|\hat{\boldsymbol{\beta}}(\tilde{\mathbf{Z}}_m)\|_2 = \infty \right\},$$

where  $\hat{\boldsymbol{\beta}}(\tilde{\mathbf{Z}}_m)$  denotes the estimate of the regression parameter using the contaminated sample  $\tilde{\mathbf{Z}}_m$ .

We assume the general position condition, which is standard in high breakdown point proofs. Suppose  $G$  is the set containing all good points  $(\mathbf{X}_i, y_i)$ , for any  $p+1$  vector  $\mathbf{v} \neq \mathbf{0}$ ,  $\{(\mathbf{X}_i, y_i) : (\mathbf{X}_i, y_i) \in G, \text{ and } \mathbf{X}_i \mathbf{v} = 0\}$  contains at most  $p-1$  points.

We first discuss the breakdown point proposed by [48]. Their method tries to solve

$$\min_{\boldsymbol{\beta}, \boldsymbol{\gamma}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \boldsymbol{\gamma}\|_2^2 + \sum_{i=1}^n P_{\mu_n}(|\gamma_i|), \quad (3.3)$$

where  $P_{\mu_n}(\cdot)$  is a nonconvex penalty. They have shown that their estimator is equivalent to M-estimator where the score function of the estimator is determined by the penalty function. If the LASSO penalty is used, it is equivalent to using Huber loss function [32], which has breakdown point of  $1/n$ , while the SCAD penalty is equivalent to using Hampel's loss function [24] that redescends to zero, which has breakdown point of at most  $1/(p+1)$  [41, 10]. These two examples show that the method proposed by [48] cannot guarantee high breakdown.

In contrast, our method would have high breakdown point of at least  $(n - h + 1)/n$ , which is shown by the following theorem

**Theorem 2** *Suppose we use the least trimmed square with truncation number  $h$  as the initial estimator, under the general position condition, the breakdown point of our estimator satisfies that  $BP(\hat{\boldsymbol{\beta}}, \mathbf{Z}) \geq \min\{(n - h + 1)/n, \lfloor (n - p)/2 \rfloor / n\}$ .*

**Remark:** It is well known that the LTS with truncation number  $h$  has a breakdown point of  $\min\{(n - h + 1)/n, \lfloor (n - p)/2 \rfloor / n\}$ , see [45] for example. This theorem provides a lower bound of the breakdown point of the proposed method, which performs at least as well as the LTS initial estimator in terms of high breakdown point. However, we often choose  $h < n/2$  so that the breakdown point can not exceed 0.5 since we aim for model to fit the majority of the data. In fact, if we do not impose any penalties on  $\boldsymbol{\beta}$ , the estimator of our method would be a regression equivalent estimator in terms of  $\boldsymbol{\beta}$ , which has an upper bound of the breakdown point of  $\lfloor (n - p)/2 \rfloor + 1$ .

This theorem reveals the importance of including adaptive weights while penalizing the mean shift parameter. Our estimator enjoys high breakdown by using the residuals of some robust initial fit with high breakdown such as the LTS method.

### 3.3.3 Outlier detection consistency

In this subsection, we consider the case when there are outliers in the conditional distribution of  $y|\mathbf{X}$ , and show that we can identify these outliers consistently. We shall assume that the fraction of outliers in the data will remain nonzero as more data are collected, otherwise we are in the trivial case. Hence we have that  $s_n = O(n)$ . We will prove that our method can identify the outliers consistently under this scenario, i.e. identify the outlier as well as the normal observations with probability tending to 1. The same reparameterization  $\boldsymbol{\eta} = \boldsymbol{\gamma}/\sqrt{n}$  is done here. Without loss of generality, we assume the first  $s_n$  components  $\boldsymbol{\eta}_0(1)$  are outliers while the remaining  $n - s_n$  components  $\boldsymbol{\eta}_0(2) = \mathbf{0}$  corresponding to the normal data points. With a little abuse of the notations from last subsection, we define  $\boldsymbol{\theta} = (\boldsymbol{\eta}(1)^\top, \boldsymbol{\beta}^\top, \boldsymbol{\eta}(2)^\top)^\top = (\boldsymbol{\theta}(1)^\top, \boldsymbol{\theta}(2)^\top, \boldsymbol{\theta}(3)^\top)^\top$ . Denote  $\mathbf{X}_{a,b}$  as the  $a$ th to  $b$ th row of the matrix  $\mathbf{X}$ . The design matrix is defined as

$$\mathbf{A} = \begin{pmatrix} A_1 & A_2 \end{pmatrix},$$

where  $A_1 = ((\sqrt{n}I_{s_n}, \mathbf{0}_{s_n \times (n-s_n)})^\top, \mathbf{X})$ ,  $A_2 = (\mathbf{0}_{(n-s_n) \times s_n}, \sqrt{n}I_{n-s_n})^\top$ . Denote

$$C = \frac{1}{n} \mathbf{A}^\top \mathbf{A} = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$$

with

$$C_{11} = \begin{pmatrix} I_{s_n} & \frac{1}{\sqrt{n}} \mathbf{X}_{1:s_n} \\ \frac{1}{\sqrt{n}} \mathbf{X}_{1:s_n}^\top & \frac{1}{n} \mathbf{X}^\top \mathbf{X} \end{pmatrix},$$

$$C_{21} = (\mathbf{0}_{(n-s_n) \times s_n}, \frac{1}{\sqrt{n}} \mathbf{X}_{s_n+1:n}) \text{ and } C_{22} = I_{n-s_n}.$$

The estimator is the solution to

$$\min_{\boldsymbol{\theta}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\boldsymbol{\theta}\|_2^2 + \sqrt{n}\mu_n \sum_{j=1}^{s_n} |\boldsymbol{\theta}_j|/|\tilde{\gamma}_j| + \lambda_n \sum_{j=s_n+1}^{s_n+p} |\boldsymbol{\theta}_j| + \sqrt{n}\mu_n \sum_{j=p+s_n+1}^{p+n} |\boldsymbol{\theta}_j|/|\tilde{\gamma}_{j-p}|$$

Noticing that  $s_n = O(n)$ , our problem is actually a weighted L-1 regression with the number of nonzero components on the order of  $O(n)$ , so our problem is unlike the traditional high dimensional penalized problem for example [60], where they dealt with the case when the number of nonzero components is on the order of  $O(n^a)$  with  $a < 1$ .

Denote  $\pi_n = \min_{i=1, \dots, s_n} |\gamma_{i0}|$ . In addition to some of the conditions we state in the last subsection, we need the following conditions:

Condition (C)

(C1)  $\pi_n n^{-1/2-1/2k} \rightarrow \infty$  as  $n \rightarrow \infty$ .

(C1')  $\pi_n n^{-1/2} (\log n)^{-1/4} \rightarrow \infty$  as  $n \rightarrow \infty$ .

(C2) The number of outliers  $s_n < n - h$ ,  $s_n = O(n)$ .

(C3) There exists  $M_4 > 0$  so that  $\boldsymbol{\alpha}^\top C_{11} \boldsymbol{\alpha}^\top \geq M_4$  for any  $\|\boldsymbol{\alpha}\| = 1$ .

Condition (C1) and (C1') requires the minimum signal of the outliers diverges with the sample size. We show this assumption reasonable and necessary by the following simple case with  $y_i = \gamma_i + \epsilon_i$ , where  $\epsilon_i \sim N(0, 1)$  and  $\gamma_i = d > 0$  for  $1 \leq i \leq s_n$  and  $\gamma_i = 0$  for  $s_n + 1 \leq i \leq n$ . Hence,  $\pi_n = d$  for this case. Since the support of the distribution is the entire real line, there cannot be a fixed  $d$  that will define an ‘‘outlier’’ in this distribution. Hence it must be assumed that  $\pi_n$  diverges sufficiently fast in order to distinguish the ‘‘outlier’’ from a random variable from the true distribution.

We state our main theorem of outlier detection consistency as follows

**Theorem 3** *Under conditions (A1)(A2)(B2)(C1)(C2)(C3), for  $\mu_n = o(\frac{\pi_n^2}{\sqrt{n}})$ ,  $\mu_n n^{-1/k} \rightarrow \infty$ ,  $\lambda_n = o(\sqrt{n}\pi_n)$  and  $\lambda_n = o(\mu_n n^{1/2-1/2k})$ , we have  $P(\hat{\gamma} =_s \gamma_0) \rightarrow 1$  as  $n \rightarrow \infty$ .*

The conditions (C1) can be relaxed to (C1') if we assume the error  $\epsilon_i$  follows an iid sub-gaussian distribution, i.e. condition (A2'). We have the following corollary:

**Corollary 2** *Under conditions (A1)(A2')(B2)(C1')(C2)(C3), for  $\mu_n = o(\frac{\pi_n^2}{\sqrt{n}})$ ,  $\sqrt{\log n}/\mu_n = O(1)$ ,  $\lambda_n = o(\sqrt{n}\pi_n)$  and  $\lambda_n = o(\mu_n \sqrt{n/\log n})$ , we have  $P(\hat{\gamma} =_s \gamma_0) \rightarrow 1$  as  $n \rightarrow \infty$ .*

### 3.4 Simulation Studies

In this section, we demonstrate our method using simulation examples. The covariate  $\mathbf{X}_i$  is generated iid from multivariate normal with covariance matrix  $\Sigma$ , where the  $jk$ th element of the matrix  $\Sigma_{jk} = 0.5^{|j-k|}$ . The true coefficient is set as  $\boldsymbol{\beta}_0 = (4, 2, 1, 0.5, 0.2, 0, \dots, 0)^\top$  with  $q = 5$  nonzero components and the remaining  $(p - q)$  elements being zero. The random error is simulated from  $\epsilon_i \sim N(0, 0.25)$ . The data is generated from  $y_i = \mathbf{X}_i \boldsymbol{\beta}_0 + \epsilon_i$ . Then we contaminate the first  $cn$  observations by setting  $\mathbf{X}_i^* = \mathbf{X}_i + L$  and  $y_i^* = y_i + V$  for  $1 \leq i \leq cn$  with parameters  $L$  and  $V$  given later. In other words, the first  $cn$  observations are outliers and the remaining ones are the normal points.

We investigate the numerical performance of our method by using the following measures:

1. **M** the masking probability (fraction of undetected true outliers)
2. **S** the swamping probability (fraction of good points labeled as outliers)
3. **JD** the joint outlier detection rate (fraction of simulations with 0 masking)
4. **FZR** the false zero rate (fraction of a coefficient that is nonzero estimated as zero)

$$FZR(\hat{\boldsymbol{\beta}}) = \frac{|\{j \in \{1, \dots, p\} : \hat{\beta}_j = 0 \wedge \beta_j \neq 0\}|}{|\{j \in \{1, \dots, p\} : \beta_j \neq 0\}|}$$

5. **FPR** the false positive rate (fraction of a coefficient that is zero estimated as nonzero)

$$FPR(\hat{\boldsymbol{\beta}}) = \frac{|\{j \in \{1, \dots, p\} : \hat{\beta}_j \neq 0 \wedge \beta_j = 0\}|}{|\{j \in \{1, \dots, p\} : \beta_j = 0\}|}$$

6. **SR** the correct selection rate (fraction of identifying both nonzeros and zeros of  $\boldsymbol{\beta}$ )
7. **CR** the correct coverage rate (fraction of identifying nonzeros of  $\boldsymbol{\beta}$ )
8. **MSE** the mean square error (MSE) of the parameters

$$MSE = (\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}_0)^\top E(\mathbf{X}_f^\top \mathbf{X}_f) (\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}_0),$$

where  $\alpha$  is the estimated intercept,  $\boldsymbol{\eta} = (\alpha, \boldsymbol{\beta}^\top)^\top$ , and  $\mathbf{X}_f = (1, \mathbf{X})$  with  $\mathbf{X}$  being the uncontaminated covariates.

For better performance in terms of outlier detection,  $M$  and  $S$  should be as small as possible while  $JD$  should be as large as possible. For sparse estimator of  $\boldsymbol{\beta}$ ,  $FZR$  and  $FPR$  should be as small as possible and  $SR$  and  $CR$  need to be as large as possible. With respect to the estimation accuracy of  $\boldsymbol{\beta}$ ,  $MSE$  should be as small as possible.

We compare our method with the SLTS method, LAD-Lasso [53] and the REWLS [23] methods. The SLTS method gives a binary weight for each observation. If  $w_i = 1$ , we identify the  $i$ th observation as a normal point and if  $w_i = 0$ , we regard it as an outlier. The truncation number is chosen the same as our initial LTS fit, that is  $h = \lfloor 0.75n \rfloor$ . For the LAD-Lasso method, as it cannot be used for outlier detection, we only report the  $MSE$ ,  $FZR$ ,  $FPR$ ,  $SR$  and  $CR$ . For the REWLS method, we also use an initial LTS fit, where we also use the truncation number  $h = \lfloor 0.75n \rfloor$ . Since the REWLS method does not perform variable selection, we only report the  $M$ ,  $S$ ,  $JD$  and  $MSE$ . In fact, for all methods mentioned above that relies on the truncation number  $h$ , our preliminary simulations indicates that the results remain almost the same when using different truncation number  $h$  as long as  $n - h$  is larger than the number of outliers and  $h \geq 0.5n$ .

We use different  $(n, p, V, L, c)$  combinations. For each combination, we run 100 repetitions of Monte Carlo studies. Results are summarized in Table 3.1, which reports the average over 100 repetitions in terms of the above performance criteria. The standard errors of these quantities are given in the parentheses. Here Regularized, SLTS, LL and REWLS denote our proposed method, the sparse least trimmed squares, the LAD-lasso method and the REWLS method respectively. From Table 3.1, we can see our method, SLTS and REWLS perform quite well in terms of outlier detection. We also observe that when there is no contamination, our method is more efficient in estimating  $\beta$  than the SLTS method and LAD-Lasso, which is actually expected, see section 3.3.1 for the theoretical justification. In fact, our estimator is also more efficient than the REWLS in the finite sample scenario. Although REWLS is also fully efficient asymptotically, the finite-sample efficiency of that method can be relative low, which is noted by [23] for example. When the contamination rate is 0.2, our method performs slightly worse than the SLTS and REWLS methods in terms of outlier detection. Moreover, our method has a much higher selection accuracy than the two comparison methods for the parameter  $\beta$  no matter we have contamination or not, although the correct coverage rate of the three methods are doing comparably well.

### 3.5 Real data application

We apply our method to the Deschutes Basin Data Set from the Regional Environmental Monitoring and Assessment Program, U.S. Environmental Protection Agency. This data set was previously analyzed by [43], where they compared the performance of several variable selection methods on this real data set. The data consists of biochemical oxygen demand (BOD) from 26 samples and various predictors including alkalinity (ALK), stream flow (SF), dissolved oxygen (DIO), total phosphorus (PHOS), pH (PH), total organic carbon (TOC), Turbidity (TURB) and water temperature (TEMP).

Table 3.1: Simulation results for our method compared with the SLTS, LL and REWLS methods.

(n,p,V,L,c)	method	M	S	JD	FZR	FPR	SR	CR	MSE
(100,15,4,0,0.1)	Regularized	0(0)	0.01(0.001)	1	0.01(0.005)	0.04(0.006)	0.66	0.94	0.2(0.006)
	SLTS	0(0)	0.04(0.002)	1	0.01(0.005)	0.34(0.023)	0.09	0.92	0.25(0.007)
	LL	*	*	*	0.04(0.008)	0.4(0.029)	0.1	0.82	0.22(0.006)
	REWLS	0(0)	0.01(0.001)	1	*	*	*	*	0.21(0.004)
(100,15,4,0,0.2)	Regularized	0.02(0.014)	0.02(0.002)	0.98	0.02(0.007)	0.05(0.009)	0.6	0.89	0.24(0.014)
	SLTS	0(0)	0.01(0.001)	1	0.03(0.008)	0.38(0.024)	0.11	0.98	0.21(0.005)
	LL	*	*	*	0.01(0.004)	0.74(0.023)	0	0.95	0.3(0.008)
	REWLS	0.001(0.001)	0(0)	0.98	*	*	*	*	0.22(0.004)
(100,15,4,4,0.1)	Regularized	0(0)	0.01(0.001)	1	0.03(0.007)	0.04(0.006)	0.54	0.86	0.22(0.006)
	SLTS	0(0)	0.04(0.002)	1	0.03(0.007)	0.37(0.024)	0.13	0.96	0.24(0.007)
	LL	*	*	*	0.02(0.008)	0.94(0.011)	0	0.9	2.47(0.015)
	REWLS	0(0)	0.01(0.001)	1	*	*	*	*	0.21(0.004)
(100,15,4,4,0.2)	Regularized	0(0)	0.02(0.002)	1	0.03(0.008)	0.07(0.01)	0.46	0.83	0.25(0.007)
	SLTS	0(0)	0.01(0.001)	1	0.04(0.008)	0.35(0.025)	0.14	0.98	0.21(0.006)
	LL	*	*	*	0.03(0.009)	0.94(0.013)	0	0.86	2.65(0.016)
	REWLS	0(0)	0(0)	1	*	*	*	*	0.22(0.004)
(100,15,0,0,0)	Regularized	*	0(0)	*	0.01(0.004)	0.03(0.005)	0.69	0.95	0.17(0.005)
	SLTS	*	0.09(0.003)	*	0.01(0.004)	0.36(0.023)	0.1	0.94	0.25(0.007)
	LL	*	*	*	0.04(0.008)	0.3(0.029)	0.21	0.78	0.2(0.005)
	REWLS	*	0.02(0.002)	*	*	*	*	*	0.22(0.005)
(200,15,4,0,0.1)	Regularized	0(0)	0.01(0.001)	1	0(0.003)	0.04(0.006)	0.67	0.98	0.15(0.004)
	SLTS	0(0)	0.02(0.001)	1	0(0.002)	0.27(0.02)	0.14	1	0.17(0.005)
	LL	*	*	*	0.02(0.006)	0.18(0.02)	0.33	0.89	0.17(0.005)
	REWLS	0(0)	0(0)	1	*	*	*	*	0.15(0.003)
(200,15,4,0,0.2)	Regularized	0.03(0.017)	0.02(0.001)	0.97	0(0.002)	0.05(0.006)	0.59	0.99	0.19(0.013)
	SLTS	0(0)	0(0)	1	0.01(0.004)	0.32(0.02)	0.08	1	0.15(0.004)
	LL	*	*	*	0.01(0.003)	0.61(0.021)	0.01	0.97	0.24(0.005)
	REWLS	0(0)	0(0)	1	*	*	*	*	0.16(0.003)
(200,15,4,4,0.1)	Regularized	0(0)	0(0)	1	0(0.002)	0.05(0.008)	0.64	0.99	0.17(0.004)
	SLTS	0(0)	0.02(0.001)	1	0(0)	0.28(0.019)	0.12	0.99	0.16(0.005)
	LL	*	*	*	0.04(0.011)	0.95(0.01)	0	0.85	2.49(0.012)
	REWLS	0(0)	0(0)	1	*	*	*	*	0.15(0.003)
(200,15,4,4,0.2)	Regularized	0(0)	0.01(0.001)	1	0.01(0.005)	0.09(0.01)	0.37	0.94	0.22(0.005)
	SLTS	0(0)	0(0)	1	0.01(0.005)	0.35(0.019)	0.06	1	0.15(0.004)
	LL	*	*	*	0.06(0.011)	0.94(0.01)	0	0.78	2.66(0.012)
	REWLS	0(0)	0(0)	1	*	*	*	*	0.16(0.003)
(200,15,0,0,0)	Regularized	*	0(0)	*	0(0)	0.02(0.005)	0.8	1	0.12(0.003)
	SLTS	*	0.05(0.002)	*	0(0)	0.24(0.017)	0.16	0.99	0.19(0.006)
	LL	*	*	*	0.02(0.006)	0.06(0.014)	0.69	0.9	0.14(0.004)
	REWLS	*	0.01(0.001)	*	*	*	*	*	0.14(0.003)

We use the same model as [43], which is given by

$$\begin{aligned} \log(BOD) = & \beta_0 + \beta_1 ALK + \beta_2 SF + \beta_3 DIO + \beta_4 PHOS \\ & + \beta_5 PH + \beta_6 \log(TOC) + \beta_7 \log(TURB) + \beta_8 TEMP + \epsilon \end{aligned}$$

We standardize all of the predictors before applying our method. We use the LTS with 25% of truncation as the initial estimates. Our method select predictors with indices (2, 3, 7). We have also detected 5 data points with subject indices (7, 8, 20, 23, 26) as the outliers, and all of the five points have a positive  $\gamma$  estimates. When using the LTS method with 25% of truncation, it selects all of the predictors. It also detects 6 data points as outliers with subject indices (5, 7, 8, 20, 23, 26). When using the LAD-Lasso, it does not eliminate any variables. For the REWLS method, we use the truncation 25% for the initial LTS fit. As it does not perform variable selection, we only report the outlier detection result. The REWLS method identifies 2 data points as outliers with indices (7, 8). We can see that the LTS and LAD-Lasso methods both do not select any variables. For our method, we pick up three important predictors and achieve variable selection. In fact, [43] compared seven variable selection methods on this data set. For each method, they reported the average number of explanatory variables when fitting to different randomly selected training samples. The largest average number of model size of the seven methods is 3.4, which is fitted using stepwise variable selection using AIC as criterion. This shows that the LTS and LAD-Lasso methods may over select the variables for this data set. In fact, in our simulation studies, we have seen the same phenomenon that LTS and LAD-Lasso methods tend to over select as they have a much higher false positive rate than our method. From the selection result of our method, we can see that the stream flow, dissolved oxygen and turbidity have positive effects on the biochemical oxygen demand. We also find that the outliers detected by our method have all been detected by the LTS method. For the REWLS method, it identifies only two outliers, which are both detected by our method.

### 3.6 Extension to the high dimensional case

In this section, we study the problem when the number of parameters  $p$  diverges with  $n$ , denoted by  $p_n$  in this section. In fact, we can easily extend our outlier detection and variable selection procedures to the  $p_n \gg n$  case. Under this scenario, especially when  $p_n > h$ , the least trimmed squares cannot be used as an initial fit because it leads to overfitting. To overcome this problem, we use the SLTS [1] as the initial fit and get the fitted residuals  $\tilde{\gamma}$ . Specifically, instead of solving

the least trimmed squares, they tried to solve the following minimization problem

$$\min_{H, \beta} Q(H, \beta) = \min_{H, \beta} \left\{ \sum_{i \in H} (y_i - \mathbf{X}_i \beta)^2 + h \lambda \sum_{j=1}^{p_n} |\beta_j| \right\},$$

with  $H \subseteq \{1, \dots, n\}$  with  $|H| = h$ . For a fixed subsample  $H$ , suppose  $\hat{\beta}_H = \operatorname{argmin}_{\beta} Q(H, \beta)$  and

$$H_{opt} = \operatorname{argmin}_{H \subseteq \{1, \dots, n\}, |H|=h} Q(H, \hat{\beta}_H),$$

the SLTS estimator is given by  $\hat{\beta}_{H_{opt}}$ . For the tuning parameter  $\lambda$ , we could still use the root trimmed mean squared prediction error criterion [1].

After we get the initial fit, we apply similar reparameterization and solve (3.2) using “lars” package. For the tuning method, we could still use the EBIC but only need to change the constant  $c$  accordingly depending on the dimension  $p_n$ . Suppose  $p_n + n = O(n^\kappa)$ , [4] shows that we require  $c > 2 - \frac{1}{\kappa}$ . We may choose  $c$  to be slightly larger than the lower bound, for example  $c = 2.01 - \frac{1}{\kappa}$ .

Under the high dimensional case, our method still enjoys the high breakdown point, which is stated in the following corollary.

**Corollary 3** *Suppose we use the sparse least trimmed squares method with truncation number  $h$ , under the general position condition, the breakdown point of our estimator  $BP(\hat{\beta}, \mathbf{Z}) \geq (n - h + 1)/n$ .*

Since [1] showed that the SLTS has breakdown point  $(n - h + 1)/n$ , we can see that our method would perform at least as well as the SLTS initial estimator with respect to high breakdown point.

## Chapter 4

# Discussions and Future work

In this thesis, we study two types of variable selection problems via penalized regression. In Chapter 1, we introduce the domain selection for the varying coefficient model using penalized local polynomial regression. Our method can identify the zero regions for each component and do estimation simultaneously. We further prove that our estimator enjoys the oracle property in the sense that they have the same asymptotic distribution as the local polynomial estimates as if the true sparsity is known. In Chapter 2, we propose a fully efficient robust estimation, outlier detection and variable selection method for the linear regression. By introducing a mean shift model, our method can identify the outliers via regularization with nonzero mean shift parameter estimates. Adaptive weights are used in shrinkage estimation of the mean shift parameter, which makes our estimator robust in terms of a high breakdown point as high as  $1/2$ . We also impose L-1 penalties on other coefficients to achieve variable selection. Our estimator enjoys fully efficiency and the outlier detection consistency when no leverage points exist.

We point out several major directions that are worth further investigation.

### 4.1 Extension of the Domain selection for the varying coefficient model

A potential extension of our method is to detect the constant regions for each coefficient function, which can be achieved by penalizing the derivative estimates. Instead of solving (2.5), we may try to solve

$$\min_{\mathbf{a}, \mathbf{b}} \left[ \sum_{i=1}^n \{Y_i - \mathbf{x}_i^\top \mathbf{a} - \mathbf{x}_i^\top \mathbf{b}(U_i - u)\}^2 (K_h(U_i - u)/K_h(0)) + m \sum_{j=1}^p P_\lambda(b_j) \right],$$

where  $P_\lambda(\cdot)$  is a penalty function such as SCAD used in the thesis. By shrinking the  $b_j$ 's to zero, we would be able to detect the constant regions. One issue we may look into further is what order of polynomial to use. We may need to use higher order of local kernel smoothing such as local quadratic or cubic fitting and change the loss function correspondingly.

Another interesting problem is to use different smoothing method for the nonparametric coefficient functions. For example, we may use B-splines, and represent the nonparametric function  $a_j(U) = \sum_{k=1}^s a_{jk} B_k(U)$  for  $1 \leq j \leq p$ , where  $\{B_k(U), 1 \leq k \leq s\}$  are a set of B-spline basis. By imposing a group SCAD penalty on the group of coefficient  $\{a_{jk}, 1 \leq k \leq s\}$ , we may achieve the domain selection. However, there are still some potential issues such as what type of loss function to use and how to select the B-spline basis as well as how many B-spline basis to use. We direct the readers to [34], where they use B-spline method to detect the zero region of the coefficient function for the functional linear model. Similar ideas may work for the varying coefficient model.

Our method only focus on the case where the number of predictors is fixed at  $p$ . It is interesting to see how our method performs when  $p$  diverges with the sample size  $n$ . The consistency rate and the asymptotic distribution of the estimator may also change correspondingly. A more interesting problem is how to do domain selection when  $p$  is on the exponential order of  $n$ . A popular method nowadays is the sure independence screening [17] technique. In fact, [18] proposed the nonparametric screening method for the varying coefficient model and it can serve as an initial step to screen out most of the predictors that are not related to the response. We can perform our method on those selected predictors to achieve domain selection.

## 4.2 Extension of the fully efficient robust estimation, outlier detection and variable selection method

Our asymptotic theories, i.e. full efficiency and outlier detection consistency, rely on the assumption that there are no leverage points. This raises an interesting problem that whether these properties still hold when the leverage points exist. We may also think about how to extend the asymptotic theories to the diverging  $p$  case. How fast can  $p$  diverge to guarantee these properties?

Another interesting problem is how to extend our method to the family of the generalized linear model or even the nonparametric settings. We are currently exploring the nonparametric outlier detection method. The traditional univariate nonparametric regression assumes that

$$y_i = f(x_i) + \epsilon_i,$$

where  $f(\cdot)$  is an unknown nonparametric function and  $\epsilon_i$ 's are iid random error with  $E(\epsilon_i) = 0$ .

Since we do not have multiple predictors for the above nonparametric model, we only focus on the outlier detection. Similar as Chapter 2, we consider the mean shift model

$$y_i = f(x_i) + \gamma_i + \epsilon_i,$$

where  $\gamma_i \neq 0$  for those outliers.

As our model involves the nonparametric function  $f(\cdot)$ , we need to apply a smoothing technique. The smoothing spline is adopted since it gives a global estimate of  $f(\cdot)$ . Without loss of generality, we assume the domain of  $x_i$  is in the bounded domain  $[0, 1]$ . Similar as Chapter 2, we add a penalty function to shrink most of the  $\gamma_i$  to zero. Denote  $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_n)^\top$ , and we solve the following minimization problem

$$\min_{f, \boldsymbol{\gamma}} \sum_{i=1}^n (y_i - f(x_i) - \gamma_i)^2 + \sum_{i=1}^n P_{\kappa_i}(\gamma_i) + \lambda \int_0^1 [f''(x)]^2 dx, \quad (4.1)$$

where  $P_{\kappa_i}(\cdot)$  is a penalty function with parameter  $\kappa_i$ . The popular penalty functions include Lasso, SCAD and adaptive Lasso, among others.

Denote  $k_1(s) = s - 0.5$ ,  $k_2(s) = \frac{1}{2}(k_1^2(s) - \frac{1}{12})$  and  $k_4(s) = \frac{1}{24}(k_1^4(s) - \frac{k_1^2(s)}{2} + \frac{7}{240})$ . Define  $K(s, t) = k_2(s)k_2(t) - k_4(s - t)$ . By the representation theorem [36], the solution of  $f(\cdot)$  in the above optimization problem can be represented as

$$f(x) = d_1 + d_2 x + \sum_{i=1}^n \alpha_i K(x_i, x).$$

Let  $\mathbf{1}$  be a column of 1,  $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_n)^\top$ ,  $\mathbf{X} = (x_1, \dots, x_n)^\top$  and  $\mathbf{K}$  be a  $n \times n$  matrix with  $ij$ th element  $K(x_i, x_j)$ . We can write (4.1) as

$$\min_{\boldsymbol{\alpha}, \boldsymbol{\gamma}} \|\mathbf{y} - d_1 \mathbf{1} - d_2 \mathbf{X} - \mathbf{K} \boldsymbol{\alpha} - \boldsymbol{\gamma}\|_2^2 + \sum_{i=1}^n P_{\kappa_i}(\gamma_i) + \lambda \boldsymbol{\alpha}^\top \mathbf{K} \boldsymbol{\alpha}. \quad (4.2)$$

Denote  $N = (\mathbf{1}, \mathbf{X}, \mathbf{K})$  and  $\boldsymbol{\theta} = (d_1, d_2, \boldsymbol{\alpha}^\top)^\top$ . We also define

$$L = \begin{pmatrix} \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times n} \\ \mathbf{0}_{n \times 2} & \mathbf{K} \end{pmatrix}$$

The above problem can be solved iteratively using the profiling idea. For a fixed  $\boldsymbol{\gamma}$ , we obtain

$$\hat{\boldsymbol{\theta}} = (N^\top N + \lambda L)^{-1} N^\top (\mathbf{y} - \boldsymbol{\gamma}) \quad (4.3)$$

Denote  $H = N(N^\top N + \lambda L)^{-1}N^\top$ . By plugging the  $\hat{\boldsymbol{\theta}}$  into the (4.2), we solve

$$\min_{\boldsymbol{\gamma}} [(\mathbf{y} - \boldsymbol{\gamma})^\top (I - H)(\mathbf{y} - \boldsymbol{\gamma}) + \sum_{i=1}^n P_{\kappa_i}(\gamma_i)]. \quad (4.4)$$

The optimization problem now becomes solving  $\boldsymbol{\gamma}$  only. If we write  $(\mathbf{y} - \boldsymbol{\gamma})^\top (I - H)(\mathbf{y} - \boldsymbol{\gamma}) = \|(I - H)^{1/2}\mathbf{y} + \{I - (I - H)^{1/2}\}\boldsymbol{\gamma} - \boldsymbol{\gamma}\|_2^2$ , by [47], we can update  $\boldsymbol{\gamma}$  via

$$\boldsymbol{\gamma}^{(j+1)} = \Theta(\{I - (I - H)^{1/2}\}\boldsymbol{\gamma}^{(j)} + (I - H)^{1/2}\mathbf{y}; \boldsymbol{\kappa}), \quad (4.5)$$

where  $\boldsymbol{\kappa} = (\kappa_1, \dots, \kappa_n)^\top$ ,  $\Theta(\cdot)$  is a threshold function determined by the penalty function, and the threshold function applies to the vectors elementwise. For the threshold function, we give two examples. If we use the Lasso penalty  $P_\kappa(\gamma) = \kappa|\gamma|$ , the threshold function is defined as

$$\Theta(x, \kappa) = \begin{cases} 0, & |x| \leq \kappa; \\ x - \text{sgn}(x)\kappa, & |x| > \kappa, \end{cases} \quad (4.6)$$

If we use the SCAD penalty function  $P_\kappa(\gamma)$ , the threshold function is defined as

$$\Theta(x, \kappa) = \begin{cases} 0, & |x| \leq \kappa; \\ \text{sgn}(x)(|x| - \kappa), & \kappa < |x| \leq 2\kappa; \\ \{(a - 1)x - \text{sgn}(x)a\kappa\}/(a - 2), & 2\kappa < |x| \leq a\kappa; \\ x, & |x| > a\kappa, \end{cases} \quad (4.7)$$

where  $a = 3.7$ .

Using the iterative formula (4.5), we solve for  $\boldsymbol{\gamma}^{(j)}$  until convergence, and the limit would be  $\hat{\boldsymbol{\gamma}}$ . For the initial point  $\boldsymbol{\gamma}^{(0)}$ , we choose it as  $(I - H)^{1/2}\mathbf{y}$ . After we get the solution for  $\hat{\boldsymbol{\gamma}}$ , we plug it into (4.3) and get  $\hat{\boldsymbol{\theta}}$ .

Notice that we have  $n$  different  $\kappa_i$ 's, it would bring difficulties in tuning these parameters. To simplify, we use the same  $\kappa$  for all  $n$  observations, i.e.  $\kappa_i$ 's are equal for all  $1 \leq i \leq n$ . In this way, we only need to tune  $\lambda$  and  $\kappa$  together. To tune them, we use the following procedure:

1. We first set grids of  $\lambda$  and  $\kappa$  respectively. For a fixed  $\lambda$ , we calculate the smoothing matrix  $H$ . We use  $H_\lambda$  to denote it when necessary as it depends on  $\lambda$ .
2. Now we tune  $\kappa$  on a set of grid points. For each  $\kappa$ , we use (4.5) to solve for  $\hat{\boldsymbol{\gamma}}(\lambda, \kappa)$ . We define

$$RSS(\lambda, \kappa) = \|(I - H_\lambda)(\mathbf{y} - \hat{\boldsymbol{\gamma}}(\lambda, \kappa))\|_2^2$$

and choose  $\kappa$  which minimizes the following BIC criterion

$$BIC(\lambda, \kappa) = m \log(RSS(\lambda, \kappa)/m) + k \log(m),$$

where  $k$  denotes the number of nonzero in  $\hat{\gamma}(\lambda, \kappa)$  and  $m$  is the effective sample size which is defined as  $m = \text{tr}\{(I - H_\lambda)\}$ . Let  $\kappa_{opt}(\lambda)$  be the optimal parameter we choose for a fixed  $\lambda$ . Denote  $\hat{\gamma}(\lambda, \kappa_{opt}(\lambda))$  be the corresponding solution for  $\kappa_{opt}(\lambda)$ .

3. We select the  $\lambda$  which minimizes the GCV

$$GCV(\lambda) = \frac{RSS(\lambda, \kappa_{opt}(\lambda))}{(n - \text{tr}(H_\lambda))^2}$$

Note, for step 2, we restrict  $k < n/2$ , i.e. assuming the true number of outlier is less than half of the sample size, to avoid overfitting. When tuning  $\kappa$ , we only consider those  $\kappa$  whose corresponding solution  $\hat{\gamma}(\lambda, \kappa)$  has less than  $n/2$  nonzero components.

We have run some preliminary simulation using SCAD penalty and find that the procedure works pretty well in outlier detection as well as the nonparametric function estimation. However, further simulation results are needed to compare with other nonparametric outlier detection methods. It is also interesting to develop some asymptotic theories on relative efficiency compared to nonparametric smoothing spline estimation when no outlier exists. The outlier detection consistency and high breakdown point of this approach are also worth investigating. A natural question can be raised that whether the approach can be improved so that it can be more robust in terms of higher breakdown point.

Another direction is to consider the nonparametric multidimensional regression or the semi-parametric multidimensional regression. To unify the notation, we consider the following partially linear model

$$y_i = f(\mathbf{X}_i) + \mathbf{Z}_i^\top \boldsymbol{\beta} + \epsilon_i, \quad (4.8)$$

where  $f(\cdot)$  is an unknown nonparametric function,  $\mathbf{X}_i = (X_{i1}, \dots, X_{ip})^\top$  are  $p$  dimensional covariates,  $\mathbf{Z}_i = (Z_{i1}, \dots, Z_{iq})^\top$  are  $q$ -dimensional covariates and  $\epsilon_i$  is the random error. This model is often used in genetics to jointly model the genetic pathway effect nonparametrically and the clinical effect parametrically, see [39] for example. If we do not have  $\mathbf{Z}_i$ , this becomes the nonparametric multidimensional regression problem.

The mean shift model can also be used under this scenario to detect the outliers in the data. Basically, we consider

$$y_i = f(\mathbf{X}_i) + \mathbf{Z}_i^\top \boldsymbol{\beta} + \gamma_i + \epsilon_i, \quad (4.9)$$

and try to find out those nonzero  $\gamma_i$ 's.

In genetics, it is often the case that the number of genes  $p$  is larger than the sample size, which makes it difficult to do statistical inference using nonparametric regression. To solve this problem, [39] used the kernel machine smoothing method for the nonparametric function  $f(\cdot)$ . Basically, they assume the function  $f(\cdot)$  resides in a functional space  $H_K$  generated by a positive definite kernel function  $K(\cdot, \cdot)$ . Under the kernel machine framework, estimation would be more accurate for multi-dimensional data. From Mercer's Theorem [8], there is a one-to-one correspondence between a positive definite kernel function and a function space  $H_K$  under some regularity conditions. We call  $H_K$  the Reproducing Kernel Hilbert Space (RKHS) generated by the kernel  $K$ . We can expand the function  $f(\cdot)$  on the basis functions in  $H_K$ , where the basis functions can be represented using the kernel function. By representer theorem [36], the solution for the nonparametric function  $f(\cdot)$  can be written as

$$f(\mathbf{X}) = \sum_{i=1}^n \theta_i K(\mathbf{X}, \mathbf{X}_i), \quad (4.10)$$

where  $\boldsymbol{\theta} = (\theta_1, \dots, \theta_n)^\top$  are unknown parameters. There are several popular kernels such as the  $d$ th degree polynomial kernel  $K(\mathbf{x}_1, \mathbf{x}_2) = (\mathbf{x}_1^\top \mathbf{x}_2 + 1)^d$  and the gaussian kernel  $K(\mathbf{x}_1, \mathbf{x}_2) = \exp(-\|\mathbf{x}_1 - \mathbf{x}_2\|^2/\rho)$ , where  $d$  and  $\rho$  are the tuning parameters. Under the kernel machine framework, one often starts with certain kernel function, which implicitly determines the functional space  $H_K$ .

We would borrow the similar idea in outlier detection and consider solving the following optimization problem

$$\min_{f, \boldsymbol{\gamma}} \sum_{i=1}^n (y_i - f(\mathbf{X}_i) - \mathbf{z}_i^\top \boldsymbol{\beta} - \gamma_i)^2 + \sum_{i=1}^n P_{\kappa_i}(\gamma_i) + \lambda \|f\|_{H_K}^2.$$

Let  $\mathbf{y} = (y_1, \dots, y_n)^\top$ ,  $\boldsymbol{\theta} = (\theta_1, \dots, \theta_n)^\top$ ,  $\mathbf{Z} = \mathbf{Z}_1, \dots, \mathbf{Z}_n^\top$ ,  $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_n)^\top$  and  $\mathbf{K}$  be a  $n \times n$  matrix with  $ij$ th element  $K(\mathbf{X}_i, \mathbf{X}_j)$ . Plug (4.10) into the objective function, we obtain

$$\min_{\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\gamma}} \|\mathbf{y} - \mathbf{Z}\boldsymbol{\beta} - \mathbf{K}\boldsymbol{\theta} - \boldsymbol{\gamma}\|_2^2 + \sum_{i=1}^n P_{\kappa_i}(\gamma_i) + \lambda \boldsymbol{\theta}^\top \mathbf{K} \boldsymbol{\theta}.$$

We would use the similar profiling technique when solving (4.2) to obtain the estimates of  $\boldsymbol{\beta}$ ,  $\boldsymbol{\theta}$  and  $\boldsymbol{\gamma}$ . However, further studies should be conducted on how to tune parameters under this scenario since the dimension of  $\mathbf{X}_i$  can be much higher than the sample size. We are also curious how this method works numerically and what asymptotic theories are behind this method.

## REFERENCES

- [1] Andreas Alfons, Christophe Croux, and Sarah Gelper. Sparse least trimmed squares regression for analyzing high-dimensional large data sets. *Annals of Applied Statistics*, page to appear, 2013.
- [2] Howard Bondell and Leonard Stefanski. Efficient robust regression via two-stage generalized empirical likelihood. *Journal of the American Statistical Association*, page to appear, 2013.
- [3] Leo Breiman. Better subset regression using the nonnegative garrote. *Technometrics*, 37:373–384, 1995.
- [4] Jiahua Chen and Zehua Chen. Extended BIC for small- $n$ -large- $P$  sparse GLM. *Statistica Sinica*, 22(2):555–574, 2012.
- [5] Chin-Tsang Chiang, John A. Rice, and Colin O. Wu. Smoothing spline estimation for varying coefficient models with repeatedly measured dependent variables. *Journal of the American Statistical Association*, 96(454):605–619, 2001.
- [6] W.S. Cleveland, E. Grosse, and W.M. Shyu. Local regression models. In J.M. Chambers and T.J. Hastie, editors, *Statistical Models in S*. 1991.
- [7] Clint W. Coakley and Thomas P. Hettmansperger. A bounded influence, high breakdown, efficient regression estimator. *Journal of the American Statistical Association*, 88(423):872–880, 1993.
- [8] Nello Cristianini and John Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 1 edition, 2000.
- [9] Christophe Croux, Peter J. Rousseeuw, and Ola Hössjer. Generalized  $S$ -estimators. *J. Amer. Statist. Assoc.*, 89(428):1271–1281, 1994.
- [10] David Donoho and Peter J. Huber. The notion of breakdown point. In *A Festschrift for Erich L. Lehmann*, Wadsworth Statist./Probab. Ser., pages 157–184. Wadsworth, Belmont, CA, 1983.
- [11] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *The Annals of Statistics*, 32(2):407–499, 2004. With discussion, and a rejoinder by the authors.
- [12] J. Fan and I. Gijbels. Data-driven Bandwidth Selection in Local Polynomial Fitting: Variable Bandwidth and Spatial Adaptation. *Journal of the Royal Statistical Society, Series B*, 57:371–394, 1995.
- [13] Jianqing Fan and Irene Gijbels. *Local Polynomial Modelling and Its Applications: Monographs on Statistics and Applied Probability 66 (Chapman & Hall/CRC Monographs on Statistics & Applied Probability)*. Chapman & Hall, 1 edition, March 1996.

- [14] Jianqing Fan and Tao Huang. Profile likelihood inferences on semiparametric varying-coefficient partially linear models. *Bernoulli*, 11(6):1031–1057, December 2005.
- [15] Jianqing Fan and Runze Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96:1348–1360, December 2001.
- [16] Jianqing Fan and Runze Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96:1348–1360, December 2001.
- [17] Jianqing Fan and Jinchi Lv. Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society. Series B.*, 70(5):849–911, 2008.
- [18] Jianqing Fan, Yunbei Ma, and Wei Dai. Nonparametric independence screening in sparse ultra-high dimensional varying coefficient models. *manuscript*, 2013.
- [19] Jianqing Fan and Wenyang Zhang. Statistical estimation in varying coefficient models. *The Annals of Statistics*, 27(5):1491–1518, 1999.
- [20] Jianqing Fan and Wenyang Zhang. Statistical methods with varying coefficient models. *Statistics and Its Interface*, 1, 2008.
- [21] Wenjiang J. Fu. Penalized regressions: The bridge versus the lasso. *Journal of Computational and Graphical Statistics*, 7(3):397–416, 1998.
- [22] I Gannaz. Robust estimation and wavelet thresholding in partial linear models. Technical Report math.ST/0612066, Dec 2006.
- [23] Daniel Gervini and Víctor J. Yohai. A class of robust and fully efficient regression estimators. *The Annals of Statistics*, 30(2):583–616, 2002.
- [24] Frank R. Hampel. The influence curve and its role in robust estimation. *J. Amer. Statist. Assoc.*, 69:383–393, 1974.
- [25] Frank R. Hampel. Beyond location parameters: robust concepts and methods. In *Proceedings of the 40th Session of the International Statistical Institute (Warsaw, 1975)*, Vol. 1. *Invited papers*, volume 46, pages 375–382, 383–391 (1976), 1975. With discussion.
- [26] David Harrison and Daniel L. Rubinfeld. Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5(1):81–102, March 1978.
- [27] Trevor Hastie and Robert Tibshirani. Varying-coefficient models. *Journal of the Royal Statistical Society. Series B*, 1993.
- [28] D. R. Hoover, J. A. Rice, C. O. Wu, and L. P Yang. Nonparametric smoothing estimates of time-varying coefficient models with longitudinal data. *Biometrika*, 85:809–822, 1998.
- [29] Jianhua Z. Huang and Haipeng Shen. Functional coefficient regression models for non-linear time series: a polynomial spline approach. *Scandinavian Journal of Statistics*, 31(4):515–534, 2004.

- [30] Jianhua Z. Huang, Colin O. Wu, and Lan Zhou. Varying-coefficient models and basis function approximations for the analysis of repeated measurements. *Biometrika*, 89(1):111–128, 2002.
- [31] Jianhua Z. Huang, Colin O. Wu, and Lan Zhou. Polynomial spline estimation and inference for varying coefficient models with longitudinal data. *Statistica Sinica*, 14(3):763–788, 2004.
- [32] Peter J. Huber. Robust estimation of a location parameter. *Annals of Mathematical Statistics*, 35:73–101, 1964.
- [33] Peter J. Huber. *Robust statistics*. John Wiley & Sons Inc., New York, 1981. Wiley Series in Probability and Mathematical Statistics.
- [34] Gareth M. James, Jing Wang, and Ji Zhu. Functional linear regression that’s interpretable. *The Annals of Statistics*, 37(5A):2083–2108, 2009.
- [35] Göran Kauermann and Gerhard Tutz. On model diagnostics using varying coefficient models. *Biometrika*, 86(1):119–128, 1999.
- [36] George Kimeldorf and Grace Wahba. Some results on Tchebycheffian spline functions. *J. Math. Anal. Appl.*, 33:82–95, 1971.
- [37] Chenlei Leng. A simple approach for varying-coefficient model selection. *Journal of Statistical Planning and Inference*, 139(7):2138–2146, 2009.
- [38] Yi Lin and Hao Helen Zhang. Component selection and smoothing in multivariate non-parametric regression. *The Annals of Statistics*, 34(5):2272–2297, 2006.
- [39] Dawei Liu, Xihong Lin, and Debashis Ghosh. Semiparametric regression of multidimensional genetic pathway data: least-squares kernel machines and linear mixed models. *Biometrics*, 63(4):1079–1088, 1311, 2007.
- [40] C.L. Mallows. On some topics in robustness. *unpublished memorandum, Bell Tel. Laboratories, Murray Hill*, 1975.
- [41] Ricardo Maronna, Oscar Bustos, and Víctor Yohai. Bias- and efficiency-robustness of general  $M$ -estimators for regression with random carriers. In *Smoothing techniques for curve estimation (Proc. Workshop, Heidelberg, 1979)*, volume 757 of *Lecture Notes in Math.*, pages 91–116. Springer, Berlin, 1979.
- [42] Lauren McCann and Roy E. Welsch. Robust variable selection using least angle regression and elemental set sampling. *Computational Statistics & Data Analysis*, 52(1):249–257, 2007.
- [43] Paul A. Murtaugh. Performance of several variable-selection methods applied to real ecological data. *Ecology Letters*, 12(10):1061–1068, October 2009.
- [44] P. Rousseeuw and V. Yohai. Robust regression by means of S-estimators. In *Robust and nonlinear time series analysis (Heidelberg, 1983)*, volume 26 of *Lecture Notes in Statist.*, pages 256–272. Springer, New York, 1984.

- [45] P. J. Rousseeuw and A. M. Leroy. *Robust regression and outlier detection*. John Wiley & Sons, Inc., New York, NY, USA, 1987.
- [46] Peter J. Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, 79(388):871–880, 1984.
- [47] Yiyuan She. Thresholding-based iterative selection procedures for model selection and shrinkage. *Electronic Journal of Statistics*, 3:384–415, 2009.
- [48] Yiyuan She and Art B. Owen. Outlier detection using nonconvex penalized regression. *Journal of the American Statistical Association*, 106(494):626–639, 2011.
- [49] Andrew F. Siegel. Robust regression using repeated medians. *Biometrika*, 69(1):242–244, April 1982.
- [50] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B*, 58(1):267–288, 1996.
- [51] Hansheng Wang, Runze Li, and Chih-Ling Tsai. Tuning parameter selectors for the smoothly clipped absolute deviation method. *Biometrika*, 93:553–568, 2007.
- [52] Hansheng Wang and Yingcun Xia. Shrinkage estimation of the varying coefficient model. *Journal of the American Statistical Association*, 104(486):747–757, 2009.
- [53] Lifeng Wang, Guang Chen, and Hongzhe Li. Group scad regression analysis for microarray time course gene expression data. *Bioinformatics*, 23:1486–1494, June 2007.
- [54] Lifeng Wang, Hongzhe Li, and Jianhua Huang. Variable selection for nonparametric varying-coefficient models for analysis of repeated measurements. *Journal of American Statistical Association*, 27:1556–1569, 2008.
- [55] Colin O. Wu, Chin-Tsang Chiang, and Donald R. Hoover. Asymptotic confidence regions for kernel smoothing of a varying-coefficient model with longitudinal data. *Journal of the American Statistical Association*, 93(444):1388–1402, 1998.
- [56] Víctor J. Yohai. High breakdown-point and high efficiency robust estimates for regression. *The Annals of Statistics*, 15(2):642–656, 1987.
- [57] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society. Series B*, 68(1):49–67, 2006.
- [58] Cunhui Zhang. Penalized linear unbiased selection. *To appear in Annals of Statistics*, 2009.
- [59] Wenyang Zhang and Sik-Yum Lee. Variable bandwidth selection in varying-coefficient models. *Journal of Multivariate Analysis*, 2000.
- [60] Peng Zhao and Bin Yu. On model selection consistency of Lasso. *Journal of Machine Learning Research*, 7:2541–2563, 2006.
- [61] Hui Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.

- [62] Hui Zou and Trevor Hastie. Regularization and variable selection via the Elastic Net. *Journal of the Royal Statistical Society B*, 67:301–320, 2005.
- [63] Hui Zou and Runze Li. One-step sparse estimates in nonconcave penalized likelihood models. *Annals of Statistics*, 36:1509–1533, 2008.

## APPENDICES

# Appendix A

## Proof in Chapter 2

We next prove our main results in Chapter 2. We begin with some auxiliary lemmas and their proofs in Section A.1 and A.2. Then we continue with proofs of our main theorems and corollaries in Section A.3.

### A.1 Lemmas

We will introduce the following Lemmas that would be used in proving the theorems.

**Lemma 1**  $s_j = O_P(1)$ ,  $r_j = O_P(h)$  and  $m = O_P(nh)$ .

Denote  $\Xi = \frac{1}{n}\Gamma_u^\top W_u \Gamma_u$ , which is a  $2p \times 2p$  matrix. We will divide  $\Xi$  into  $p \times p$  submatrices.

$$\begin{pmatrix} A_{11} & \dots & A_{1p} \\ \dots & \dots & \dots \\ A_{p1} & \dots & A_{pp} \end{pmatrix}$$

where the  $2 \times 2$  matrix  $A_{kl}$  ( $1 \leq k, l \leq p$ ) is as follows:

$$\begin{pmatrix} \frac{1}{n} \sum_{i=1}^n x_{ik} x_{il} K_h(U_i - u) / K_h(0) & \frac{1}{n} \sum_{i=1}^n x_{ik} x_{il} (U_i - u) K_h(U_i - u) / K_h(0) \\ \frac{1}{n} \sum_{i=1}^n x_{ik} x_{il} (U_i - u) K_h(U_i - u) / K_h(0) & \frac{1}{n} \sum_{i=1}^n x_{ik} x_{il} (U_i - u)^2 K_h(U_i - u) / K_h(0) \end{pmatrix}$$

We will define the new submatrix  $B_{kl}$  as:

$$\begin{pmatrix} \frac{1}{n} \sum_{i=1}^n x_{ik} x_{il} K_h(U_i - u) / K_h(0) s_j^2 & \frac{1}{n} \sum_{i=1}^n x_{ik} x_{il} (U_i - u) K_h(U_i - u) / K_h(0) s_j r_j \\ \frac{1}{n} \sum_{i=1}^n x_{ik} x_{il} (U_i - u) K_h(U_i - u) / K_h(0) s_j r_j & \frac{1}{n} \sum_{i=1}^n x_{ik} x_{il} (U_i - u)^2 K_h(U_i - u) / K_h(0) r_j^2 \end{pmatrix}$$

**Lemma 2** Every element of  $B_{kl}$  is on the order of  $O_P(h)$ .

Define  $\Sigma$  to be the following matrix

$$\begin{pmatrix} B_{11} & \dots & B_{1p} \\ \dots & \dots & \dots \\ B_{p1} & \dots & B_{pp} \end{pmatrix}$$

where every element in  $\Sigma$  is on the order of  $O_P(h)$ .

The following Lemma is taken directly from Theorem 1 of [20].

**Lemma 3** *Under the conditions in [59], we have*

$$cov^{-1/2}(\hat{\mathbf{a}}(u))\{\hat{\mathbf{a}}(u) - \mathbf{a}(u) - bias(\hat{\mathbf{a}}(u))\} \xrightarrow{d} N(0, I_p)$$

with

$$\begin{aligned} Bias(\hat{\mathbf{a}}(u)) &= 2^{-1}\mu_2\mathbf{a}''(u)h^2 \\ Cov(\hat{\mathbf{a}}(u)) &= \{nhf(u)E(XX^\top|U=u)\}^{-1}\nu_0\sigma^2(u) \end{aligned}$$

where  $\mu_2 = \int u^2K(u)du$  and  $\nu_0 = \int K^2(u)du$ ,  $f(u)$  is the density of  $u$ .

**Lemma 4** *From Lemma 3 and [13], we have*

$$\begin{aligned} bias(\hat{\mathbf{a}}(u)) &= O_P(h^2) \\ cov(\hat{\mathbf{a}}(u)) &= O_P((nh)^{-1}) \\ bias(\hat{\mathbf{b}}(u)) &= O_P(h^2) \\ cov(\hat{\mathbf{b}}(u)) &= O_P((nh^3)^{-1}) \end{aligned}$$

Remark: Under condition (A1), We will have  $\hat{\mathbf{a}}(u) - \mathbf{a}(u) = O_P(b_n)$  and  $\hat{\mathbf{b}}(u) - \mathbf{b}(u) = O_P(b_n/h)$ . Consequently, we can get  $\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_0 = O_P(b_n)$ , where  $\hat{\boldsymbol{\beta}}$  is the local polynomial regression estimator and  $\boldsymbol{\beta}_0$  is the true value.

## A.2 Proof of Lemmas

Proof of Lemma 1:

For the following proof, we will denote  $C$  as a constant term. We denote  $f(t)$  be the density

function of the random variable  $U_i$ . We can see

$$\begin{aligned}
E[x_{ij}(U_i - u)\{K_h(U_i - u)/K_h(0)\}^{1/2}] &= Ch^{1/2}E\{E(x_{ij}(U_i - u)(K_h(U_i - u))^{1/2}|U_i)\} \\
&= Ch^{1/2}E\{E(x_{ij}|U_i) \int (t - u)(K_h(t - u))^{1/2}f(t)dt\} \\
&= ChE\{E(x_{ij}|U_i) \int vK(v)f(u + hv)dv\} \\
&= O_P(h)
\end{aligned}$$

$$\begin{aligned}
E([x_{ij}(U_i - u)\{K_h(U_i - u)/K_h(0)\}^{1/2}]^2) &= ChE[E\{x_{ij}^2(U_i - u)^2(K_h(U_i - u))|U_i\}] \\
&= ChE\{E(x_{ij}^2|U_i) \int (t - u)^2(K_h(t - u))f(t)dt\} \\
&= CE\{E(x_{ij}^2|U_i) \int h^2v^2K(v)f(u + hv)dv\} \\
&= O_P(h^2).
\end{aligned}$$

As a result, we can get  $Var\{x_{ij}(U_i - u)(K_h(U_i - u)/K_h(0))^{1/2}\} = O_P(h^2)$ . Consequently,  $r_j = O_P(E[x_{ij}(U_i - u)\{K_h(U_i - u)/K_h(0)\}^{1/2}]) + O_P(\sqrt{Var[x_{ij}(U_i - u)\{K_h(U_i - u)/K_h(0)\}^{1/2}]}) = O_P(h)$ . Similarly, we will have  $s_j = O_P(1)$ .

For the effective sample size  $m$ , we have

$$\begin{aligned}
E(m) &= E\left(\frac{\sum_{i=1}^n K_h(U_i - u)}{K_h(0)}\right) \\
&= Cnh \int K_h(t - u)f(t)dt \\
&= Cnh \int K(v)f(u + hv)dv \\
&= O(nh)
\end{aligned}$$

Similarly, we have  $Var(m) = O(nh)$ . As  $nh = o(1)$ , we can see that  $m = O_P(E(m)) + O_P(\sqrt{Var(m)}) = O_P(nh)$ .

Proof of Lemma 2:

We can see

$$\sum_{i=1}^n x_{ik}x_{il}K_h(U_i - u) = O_P\left(E\left(\sum_{i=1}^n x_{ik}x_{il}K_h(U_i - u)\right)\right) + O_P\left(\sqrt{Var\left(\sum_{i=1}^n x_{ik}x_{il}K_h(U_i - u)\right)}\right).$$

For the mean, we will get

$$\begin{aligned}
E\left(\sum_{i=1}^n x_{ik}x_{il}K_h(U_i - u)\right) &= nE[x_{ik}x_{il}K_h(U_i - u)] \\
&= nE[E(x_{ik}x_{il}K_h(U_i - u))|U_i] \\
&= nE[x_{ik}x_{il} \int K_h(t - u)f(t)dt] \\
&= nE[x_{ik}x_{il} \int K(v)f(u + hv)dv] \\
&= O_P(n)
\end{aligned}$$

For the variance, we can use similar technique and get

$$\text{Var}\left(\sum_{i=1}^n x_{ik}x_{il}K_h(U_i - u)\right) = O_P(n/h).$$

As  $s_j = O_P(1)$  and  $r_j = O_P(h)$ , we will get  $\frac{1}{n} \sum_{i=1}^n x_{ik}x_{il}K_h(U_i - u)/s_j^2 = O_P(h)$ . Similarly, we obtain  $\frac{1}{n} \sum_{i=1}^n x_{ik}x_{il}(U_i - u)K_h(U_i - u)/K_h(0)s_jr_j = O_P(h)$  and  $\frac{1}{n} \sum_{i=1}^n x_{ik}x_{il}(U_i - u)K_h(U_i - u)/K_h(0)r_j^2 = O_P(h)$ .

For Lemma 3, see [59] for detailed proof.

For Lemma 4, we can use the similar technique in [13].

### A.3 Proof of Theorems and Corollary

Proof of Theorem 1:

We write our tuning parameter  $\lambda$  as  $\lambda_n$  because the tuning parameter depends on the sample size. Let  $\alpha_n = b_n + a_n$ . Denote  $L(\boldsymbol{\beta}) = (\mathbf{Y} - \Gamma_{hu}\boldsymbol{\beta})^\top W_u(\mathbf{Y} - \Gamma_{hu}\boldsymbol{\beta})$ . We want to show that for any  $\epsilon > 0$ , there exists a large constant C such that

$$P\left\{\inf_{\|\mathbf{v}\|=C} Q(\boldsymbol{\beta}_0 + \alpha_n\mathbf{v}) > Q(\boldsymbol{\beta}_0)\right\} \geq 1 - \epsilon, \tag{A.1}$$

which implies  $\hat{\boldsymbol{\beta}}_\lambda - \boldsymbol{\beta}_0 = O_P(\alpha_n)$ .

As  $P_{\lambda_n}(0) = 0$ , we will have

$$\begin{aligned}
& Q(\boldsymbol{\beta}_0 + \alpha_n \mathbf{v}) - Q(\boldsymbol{\beta}_0) \\
& \geq L(\boldsymbol{\beta}_0 + \alpha_n \mathbf{v}) - L(\boldsymbol{\beta}_0) + m \sum_{j=1}^s [P_{\lambda_n}(\|\boldsymbol{\beta}_{j0} + \alpha_n \mathbf{v}_j\|) - P_{\lambda_n}(\|\boldsymbol{\beta}_{j0}\|)] \\
& = \alpha_n^2 \mathbf{v}^\top \Gamma_{hu}^\top W_u \Gamma_{hu} \mathbf{v} - 2\alpha_n \mathbf{v}^\top \Gamma_{hu}^\top W_u (\mathbf{Y} - \Gamma_{hu} \boldsymbol{\beta}_0) \\
& \quad + m \sum_{j=1}^s [P_{\lambda_n}(\|\boldsymbol{\beta}_{j0} + \alpha_n \mathbf{v}_j\|) - P_{\lambda_n}(\|\boldsymbol{\beta}_{j0}\|)] \\
& = n\alpha_n^2 \mathbf{v}^\top \Sigma \mathbf{v} - 2\alpha_n \mathbf{v}^\top \Gamma_{hu}^\top W_u (\mathbf{Y} - \Gamma_{hu} \hat{\boldsymbol{\beta}}) \\
& \quad - 2n\alpha_n \mathbf{v}^\top \Sigma (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_0) + m \sum_{j=1}^s [P_{\lambda_n}(\|\boldsymbol{\beta}_{j0} + \alpha_n \mathbf{v}_j\|) - P_{\lambda_n}(\|\boldsymbol{\beta}_{j0}\|)] \\
& = A_1 + A_2 + A_3 + A_4
\end{aligned}$$

By Lemma 2, we have  $A_1 = O_P(nh\alpha_n^2)$ . As  $\hat{\boldsymbol{\beta}}$  is the minimizer of  $L(\boldsymbol{\beta})$ , we will have  $\frac{\partial L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}|_{\boldsymbol{\beta}=\hat{\boldsymbol{\beta}}} = 0$ , which indicates  $A_2 = 0$ . By Lemma 4,  $\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_0\| = O_P(b_n)$ , so we will have  $A_3 = O_P(nh\alpha_n^2)$ , so by choosing a sufficient large  $C$ ,  $A_1$  dominates  $A_3$ . For the term  $A_4$ , we have

$$\begin{aligned}
A_4 & = m \sum_{j=1}^s [P_{\lambda_n}(\|\boldsymbol{\beta}_{j0} + \alpha_n \mathbf{v}_j\|) - P_{\lambda_n}(\|\boldsymbol{\beta}_{j0}\|)] \\
& = m \sum_{j=1}^s P'_{\lambda_n}(\|\boldsymbol{\beta}_{j0}\|)(\|\boldsymbol{\beta}_{j0} + \alpha_n \mathbf{v}_j\| - \|\boldsymbol{\beta}_{j0}\|) \\
& \quad + m \sum_{j=1}^s P''_{\lambda_n}(\|\boldsymbol{\beta}_{j0}\|)(\|\boldsymbol{\beta}_{j0} + \alpha_n \mathbf{v}_j\| - \|\boldsymbol{\beta}_{j0}\|)^2(1 + o(1)) \\
& \leq m \sum_{j=1}^s P'_{\lambda_n}(\|\boldsymbol{\beta}_{j0}\|)\alpha_n \|\mathbf{v}_j\| + m \sum_{j=1}^s \max_{1 \leq j \leq s} |P''_{\lambda_n}(\|\boldsymbol{\beta}_{j0}\|)|\alpha_n^2 \|\mathbf{v}_j\|^2(1 + o(1)) \\
& \leq m \sum_{j=1}^s a_n \alpha_n \|\boldsymbol{\beta}_{j0}\| \|\mathbf{v}\| + ms\alpha_n^2 \max_{1 \leq j \leq s} |P''_{\lambda_n}(\|\boldsymbol{\beta}_{j0}\|)| \|\mathbf{v}\|^2(1 + o(1)) \\
& = A_5 + A_6.
\end{aligned}$$

By Lemma 1, we have  $A_5 = O_P(m\alpha_n^2) = O_P(nh\alpha_n^2)$  which is dominated by  $A_1$ . We will also have  $A_6 = o_P(nh\alpha_n^2)$ , which is also dominated by  $A_1$ . As a result, (A.1) holds, which indicates  $\hat{\boldsymbol{\beta}}_\lambda - \boldsymbol{\beta}_0 = O_P(\alpha_n)$ . Further, under condition (A2), we will have  $\hat{\boldsymbol{\beta}}_\lambda - \boldsymbol{\beta}_0 = O_P(b_n)$ .

Proof of Theorem 2:

We need to prove

$$P(\|\hat{\boldsymbol{\beta}}_{\lambda_j}\| = 0) \rightarrow 1 \quad (\text{A.2})$$

for any  $s + 1 \leq j \leq p$ . If  $\hat{\boldsymbol{\beta}}_{\lambda_j} \neq 0$ , it should be the solution of the following equation

$$\begin{aligned} \mathbf{0} &= \frac{\partial Q(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}_j} \Big|_{\boldsymbol{\beta}=\hat{\boldsymbol{\beta}}_{\lambda_j}} \\ &= -2\Gamma_{huj}^\top W_u (\mathbf{Y} - \sum_{j=1}^p \Gamma_{huj} \hat{\boldsymbol{\beta}}_{\lambda_j}) + mP'_{\lambda_n}(\|\hat{\boldsymbol{\beta}}_{\lambda_j}\|) \frac{\hat{\boldsymbol{\beta}}_{\lambda_j}}{\|\hat{\boldsymbol{\beta}}_{\lambda_j}\|} \\ &= B_1 + B_2. \end{aligned}$$

We can see

$$\begin{aligned} B_1 &= -2\Gamma_{huj}^\top W_u (\mathbf{Y} - \sum_{j=1}^p \Gamma_{huj} \hat{\boldsymbol{\beta}}_j) - 2\Gamma_{huj}^\top W_u \sum_{j=1}^p (\Gamma_{huj} \hat{\boldsymbol{\beta}}_{\lambda_j} - \Gamma_{huj} \hat{\boldsymbol{\beta}}_j) \\ &= 0 - 2\Gamma_{huj}^\top W_u \Gamma_u (\hat{\boldsymbol{\beta}}_{\lambda_j} - \hat{\boldsymbol{\beta}}_j) \\ &= -2n\Sigma_j O_P(b_n) \\ &= O_P(nhb_n), \end{aligned}$$

where  $\Sigma_j$  is the  $(2j-1)$ th and  $2j$ th row of the matrix  $\Sigma$ . For  $B_2$ , under Condition (A2), we have  $\|B_2\| = mP'_{\lambda_n}(\|\boldsymbol{\beta}_{\lambda_j}\|) = O_P(nh\lambda_n)$ . Under Condition (A2), we will have  $P(\|B_2\| > \|B_1\|) \rightarrow 1$ , which indicates with probability tending to one, (A.2) does not hold. Thus  $\hat{\boldsymbol{\beta}}_{\lambda_j}$  must locate at the place where  $Q(\boldsymbol{\beta})$  is not differentiable. Since the only place  $Q(\boldsymbol{\beta})$  is not differentiable for  $\boldsymbol{\beta}_j$  is the origin, we will have  $P(\|\hat{\boldsymbol{\beta}}_{\lambda_j}\| = 0) \rightarrow 1$  for any  $s + 1 \leq j \leq p$ .

Proof of Theorem 3:

Let  $L(\boldsymbol{\beta}) = (\mathbf{Y} - \Gamma_{hu}\boldsymbol{\beta})^\top W_u (\mathbf{Y} - \Gamma_{hu}\boldsymbol{\beta})$ . We will have

$$\frac{\partial Q(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}_j} \Big|_{\boldsymbol{\beta}=(\hat{\boldsymbol{\beta}}_N^\top, \mathbf{0}^\top)^\top} = 0,$$

for  $1 \leq j \leq s$ . Note that  $\hat{\beta}_{\lambda N}$  is a consistent estimator,

$$\begin{aligned}
\frac{\partial Q(\beta)}{\partial \beta_j} \Big|_{\beta=(\hat{\beta}_N^\top, \mathbf{0}^\top)^\top} &= \frac{\partial L(\beta)}{\partial \beta_j} \Big|_{\beta=(\hat{\beta}_{\lambda N}^\top, \mathbf{0}^\top)^\top} + mP'_{\lambda_n}(\|\hat{\beta}_{\lambda_j}\|) \frac{\hat{\beta}_{\lambda_j}}{\|\hat{\beta}_{\lambda_j}\|} \\
&= \frac{\partial L(\hat{\beta}_N)}{\partial \beta_j} + \sum_{l=1}^s \left\{ \frac{\partial^2 L(\hat{\beta}_N)}{\partial \beta_j \partial \beta_l} + o_P(1) \right\} (\hat{\beta}_{\lambda_l} - \hat{\beta}_l) \\
&\quad + m \{ P'_{\lambda_n}(\|\beta_{l0}\|) \frac{\beta_{l0}}{\|\beta_{l0}\|} + (T(\beta_{l0}) + o_P(1)) (\hat{\beta}_{\lambda_l} - \beta_{l0}) \} \\
&= 0 + \sum_{l=1}^s (2n\Sigma_{jl} + o_P(1)) (\hat{\beta}_{\lambda_l} - \beta_{l0} + \beta_{l0} - \hat{\beta}_l) \\
&\quad + m \{ P'_{\lambda_n}(\|\beta_{l0}\|) \frac{\beta_{l0}}{\|\beta_{l0}\|} + (T(\beta_{l0}) + o_P(1)) (\hat{\beta}_{\lambda_l} - \beta_{l0}) \}
\end{aligned}$$

where  $\Sigma_{jl}$  denotes the  $(2j-1, 2l-1)$ ,  $(2j-1, 2l)$ ,  $(2j, 2l-1)$ ,  $(2j, 2l)$  elements of the matrix  $\Sigma$ .

We will have

$$(2n\Sigma_s + o_P(1))(\hat{\beta}_N - \beta_{N0}) = (mH + 2n\Sigma_s + o_P(1))(\hat{\beta}_{\lambda N} - \beta_{N0}) + m\mathbf{d}.$$

We can write

$$\begin{aligned}
\hat{\beta}_N - \beta_{N0} &= \Sigma_s^{-1} \left( \frac{m}{2n} H + \Sigma_s \right) (\hat{\beta}_{\lambda N} - \beta_{N0}) + \frac{m}{2n} \Sigma_s^{-1} \mathbf{d} + o_P(n^{-1}) \\
&= \Sigma_s^{-1} \left( \frac{m}{2n} H + \Sigma_s \right) [(\hat{\beta}_{\lambda N} - \beta_{N0}) + \left( \frac{m}{2n} H + \Sigma_s \right)^{-1} \frac{m}{2n} \mathbf{d}] + o_P(n^{-1})
\end{aligned}$$

Proof of Corollary 1:

From Theorem 3, we can get

$$\begin{aligned}
& cov^{-1/2}(\hat{\beta}_N) \{ \hat{\beta}_N - \beta_{N0} - bias(\hat{\beta}_N) \} \\
&= cov^{-1/2}(\hat{\beta}_N) \{ \Sigma_s^{-1} \left( \frac{m}{2n} H + \Sigma_s \right) (\hat{\beta}_{\lambda N} - \beta_{N0}) + \left( \frac{m}{2n} H + \Sigma_s \right)^{-1} \frac{m}{2n} \mathbf{d} \} \\
&\quad + o_P(n^{-1}) - bias(\hat{\beta}_N) \\
&= cov^{-1/2}(\hat{\beta}_N) \Sigma_s^{-1} \left( \frac{m}{2n} H + \Sigma_s \right) (\hat{\beta}_{\lambda N} - \beta_{N0}) + cov^{-1/2}(\hat{\beta}_N) \Sigma_s^{-1} \frac{m}{2n} \mathbf{d} \\
&\quad - cov^{-1/2}(\hat{\beta}_N) bias(\hat{\beta}_N) - cov^{-1/2}(\hat{\beta}_N) o_P(n^{-1}) \\
&= T_1 + T_2 + T_3 + T_4
\end{aligned}$$

Under Condition (A2), as  $\Sigma_s = O_P(h)$  and  $\frac{m}{2n} H = o(h)$ , the first term  $T_1$  is approximately  $cov^{-1/2}(\hat{\beta}_N)(\hat{\beta}_{\lambda N} - \beta_{N0})$ . We can easily see that the second term  $T_2$  is negligible compared to the first term. Similarly, as  $b_n = o(1/n)$ , the fourth term  $T_4$  is negligible compared to the first

term. As a result, we can get

$$\begin{aligned} & cov^{-1/2}(\hat{\boldsymbol{\beta}}_N)\{\hat{\boldsymbol{\beta}}_{\lambda N} - \boldsymbol{\beta}_{N0} - bias(\hat{\boldsymbol{\beta}}_N)\} \\ = & cov^{-1/2}(\hat{\boldsymbol{\beta}}_N)\{\hat{\boldsymbol{\beta}}_N - \boldsymbol{\beta}_{N0} - bias(\hat{\boldsymbol{\beta}}_N)\} + o_P(1) \end{aligned} \quad (\text{A.3})$$

From Lemma 3, we can get

$$cov^{-1/2}(\hat{\mathbf{a}}_N(u))\{\hat{\mathbf{a}}_N(u) - \mathbf{a}_{N0}(u) - bias(\hat{\mathbf{a}}_N(u))\} \xrightarrow{d} N(0, I_s)$$

where  $\mathbf{a}_N(u)$  denotes the components that are nonzero. As a result, we can get from (A.3) that

$$cov^{-1/2}(\hat{\mathbf{a}}_N(u))\{\hat{\mathbf{a}}_{\lambda N}(u) - \mathbf{a}_{N0}(u) - bias(\hat{\mathbf{a}}_N(u))\} \xrightarrow{d} N(0, I_s)$$

where  $\hat{\mathbf{a}}_{\lambda N}(u)$  denotes our penalized local polynomial estimates for the function value. This shows the oracle property.

# Appendix B

## Proof in Chapter 3

We next prove our main results in Chapter 3. We begin with some auxiliary lemmas and their proofs in Section B.1 and continue with proofs of our main theorems and corollaries in Section B.2.

### B.1 Lemmas and Proofs

Denote  $\kappa_n = \max_{1 \leq i \leq n} |\tilde{\gamma}_i|$ . We have the following lemmas:

**Lemma 1** *Under condition (A1)(A2), we have  $\kappa_n = O_P(n^{\frac{1}{2k}})$ .*

Proof of Lemma 1:

As the least trimmed estimator is root-n consistent [45], we have  $\|\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0\| = O_P(n^{-1/2})$ . By condition (A1),  $\max_{1 \leq i \leq n} \{\mathbf{X}_i(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0)\} \leq \max_{1 \leq i \leq n} \|\mathbf{X}_i\| O_P(n^{-1/2}) = O_P(n^{-1/2} n^{1/2k}) = O_P(1)$ . Following condition (A2), one obtains  $\max_{1 \leq i \leq n} |\epsilon_i| = O_P(n^{\frac{1}{2k}})$ . Notice that  $\tilde{\gamma}_i = \epsilon_i + \mathbf{X}_i(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0)$ , we obtain  $\kappa_n = O_P(n^{\frac{1}{2k}})$ .

**Lemma 2** *Under condition (A1)(A2'), we have  $\kappa_n = o_P(\sqrt{\log n})$ .*

Proof of Lemma 2:

Similar as Lemma 1, we have  $\max_{1 \leq i \leq n} \{\mathbf{X}_i(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0)\} = O_P(1)$ . By condition (A2'), one obtains  $\max_{1 \leq i \leq n} |\epsilon_i| = o_P(\sqrt{\log n})$ . Notice that  $\tilde{\gamma}_i = \epsilon_i + \mathbf{X}_i(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0)$ , we have  $\kappa_n = o_P(\sqrt{\log n})$ .

The Lemma 3 and 4 applies for the case when outliers exist. With slightly abuse of the notation, denote  $\delta_n = \min_{i=1, \dots, s_n} |\tilde{\gamma}_i|$  and  $\kappa_n = \max_{i=s_n+1, \dots, n} |\tilde{\gamma}_i|$ .

**Lemma 3** *Under Condition (A1)(A2)(C1)(C2)(C3), we have  $\delta_n = O_P(\pi_n)$  and  $\kappa_n = O_P(n^{1/2k})$ .*

Proof of Lemma 3:

As the least trimmed estimator is root-n consistent [45], we have  $\|\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0\| = O_P(n^{-1/2})$ , which

indicates  $\mathbf{X}_i(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0) = O_P(1)$  by condition (A1). Since the first  $s_n$  data points are the outliers and noticing that  $\tilde{\gamma}_i = \mathbf{X}_i(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0) + \epsilon_i$  for  $s_n + 1 \leq i \leq n$ , by condition (A2), one obtains  $\max_{i=s_n+1, \dots, n} |\tilde{\gamma}_i| = O_P(n^{1/2k})$ . As  $\tilde{\gamma}_i = \mathbf{X}_i(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0) + \epsilon_i + \gamma_{i0}$  and  $\pi_n n^{-1/2k} \rightarrow \infty$  by condition (C1), one obtains  $\min_{i=1, \dots, s_n} |\tilde{\gamma}_i| = O_P(\pi_n)$ .

**Lemma 4** *Under Condition (A1)(A2')(C1')(C2)(C3), we have  $\delta_n = O_P(\pi_n)$  and  $\kappa_n = o_P(\sqrt{\log n})$ .*

Proof of Lemma 4:

The proof of Lemma 4 is similar as that of Lemma 3, and we have  $\mathbf{X}_i(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0) = O_P(1)$ . By condition (A2'), one obtains  $\max_{i=s_n+1, \dots, n} |\tilde{\gamma}_i| = o_P(\sqrt{\log n})$ . As  $\tilde{\gamma}_i = \mathbf{X}_i(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0) + \epsilon_i + \gamma_{i0}$  and  $\pi_n (\log n)^{-1/2} \rightarrow \infty$ , by condition (C1'), one obtains  $\min_{i=1, \dots, s_n} |\tilde{\gamma}_i| = O_P(\pi_n)$ .

## B.2 Proof of Theorems and Corollaries

Proof of Theorem 1:

For any  $k$  dimensional vectors  $\mathbf{a}$  and  $\mathbf{b}$ , define  $\frac{\mathbf{a}}{\mathbf{b}} = (\mathbf{a}_1/\mathbf{b}_1, \dots, \mathbf{a}_k/\mathbf{b}_k)^\top$ . Let  $\hat{\mathbf{u}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0 = (\mathbf{u}(1)^\top, \mathbf{u}(2)^\top, \mathbf{u}(3)^\top)^\top$ . Also define  $V_n(\mathbf{u}) = \sum_{i=1}^n [(\epsilon_i - \mathbf{A}\mathbf{u})^2 - \epsilon_i^2] + \lambda_n \|\mathbf{u}(1) + \boldsymbol{\beta}_0(1)\|_1 + \lambda_n \|\mathbf{u}(2)\|_1 + \sqrt{n} \mu_n \frac{\|\mathbf{u}(3)\|_1}{|\tilde{\gamma}|}$ . The first summation in  $V_n(\mathbf{u})$  can be simplified as  $-2W(\sqrt{n}\mathbf{u}) + (\sqrt{n}\mathbf{u})^\top C(\sqrt{n}\mathbf{u})$ , where  $W = (W(1)^\top, W(2)^\top, W(3)^\top)^\top = (\mathbf{X}_{1,q}^\top \boldsymbol{\epsilon}/\sqrt{n}, X_{q+1,p}^\top \boldsymbol{\epsilon}/\sqrt{n}, \boldsymbol{\epsilon})$ . Then by definition we have:

$$\{\text{sign}(\hat{\boldsymbol{\theta}}_j) = \text{sign}(\boldsymbol{\theta}_{j0}), \text{ for } j = 1, \dots, q\} \in \{\text{sign}(\boldsymbol{\theta}_0(1))\hat{\mathbf{u}}(1) > -|\boldsymbol{\theta}_0(1)|\}$$

By uniqueness of Lasso solutions, if there exists  $\hat{\mathbf{u}}$  the following holds

$$\begin{aligned} C_{11}(\sqrt{n}\hat{\mathbf{u}}(1)) - W(1) &= -\frac{\lambda_n}{2\sqrt{n}} \text{sign}(\boldsymbol{\theta}_0(1)) \\ |\hat{\mathbf{u}}(1)| &< |\boldsymbol{\beta}_0(1)| \\ -\frac{\lambda_n}{2\sqrt{n}} \mathbf{1} &\leq C_{21}(\sqrt{n}\hat{\mathbf{u}}(1)) - W(2) \leq \frac{\lambda_n}{2\sqrt{n}} \mathbf{1} \\ -\frac{\mu_n}{2} \frac{\mathbf{1}}{|\tilde{\gamma}|} &\leq C_{31}(\sqrt{n}\hat{\mathbf{u}}(1)) - W(3) \leq \frac{\mu_n}{2} \frac{\mathbf{1}}{|\tilde{\gamma}|}, \end{aligned}$$

then  $\text{sign}(\hat{\boldsymbol{\theta}}(1)) = \text{sign}(\boldsymbol{\theta}_0(1))$ ,  $\text{sign}(\hat{\boldsymbol{\theta}}(2)) = \text{sign}(\boldsymbol{\theta}_0(2)) = \mathbf{0}$ ,  $\text{sign}(\hat{\boldsymbol{\theta}}(3)) = \text{sign}(\boldsymbol{\theta}_0(3)) = \mathbf{0}$ . Substitute  $\hat{\mathbf{u}}(1)$  and bound the absolute values, the existence of such  $\hat{\mathbf{u}}$  is implied by

$$\begin{aligned} A_n & : |C_{11}^{-1}W(1)| < \sqrt{n}(|\boldsymbol{\theta}_0(1)| - \frac{\lambda_n}{2n}|C_{11}^{-1}\text{sign}(\boldsymbol{\theta}_0(1))|) \\ B_n & : |C_{21}C_{11}^{-1}W(1) - W(2)| \leq \frac{\lambda_n}{2\sqrt{n}}(\mathbf{1} - |C_{21}C_{11}^{-1}\text{sign}(\boldsymbol{\theta}_0(1))|) \\ C_n & : |C_{31}C_{11}^{-1}W(1) - W(3)| \leq \frac{\mu_n}{2} \frac{\mathbf{1}}{|\tilde{\gamma}|} - \frac{\lambda_n}{2\sqrt{n}}|C_{31}C_{11}^{-1}\text{sign}(\boldsymbol{\theta}_0(1))| \end{aligned}$$

As a result,  $P(\hat{\boldsymbol{\theta}} =_s \boldsymbol{\theta}_0) \geq P(A_n \cap B_n \cap C_n)$ . Meanwhile

$$\begin{aligned} 1 - P(A_n \cap B_n \cap C_n) & \geq \sum_{j=1}^q P(|z_j| \geq \sqrt{n}(|\boldsymbol{\theta}_0(1)| - \frac{\lambda_n}{2n}a_j)) + \sum_{j=1}^{p-q} P(|\zeta_j| \geq \frac{\lambda_n}{2\sqrt{n}}b_j) \\ & \quad + \sum_{j=1}^n P(|\xi_j| \geq \frac{\mu_n}{2|\tilde{\gamma}_j|} - \frac{\lambda_n}{2\sqrt{n}}c_j), \end{aligned}$$

where  $\mathbf{z} = C_{11}^{-1}W(1)$ ,  $\boldsymbol{\zeta} = (\zeta_1, \dots, \zeta_{p-q})^\top = C_{21}C_{11}^{-1}W(1) - W(2)$ ,  $\boldsymbol{\xi} = (\xi_1, \dots, \xi_n)^\top = C_{31}C_{11}^{-1}W(1) - W(3)$ ,  $\mathbf{a} = (a_1, \dots, a_q)^\top = C_{11}^{-1}\text{sign}(\boldsymbol{\theta}_0(1))$ ,

$\mathbf{b} = (b_1, \dots, b_{p-q})^\top = \mathbf{1} - |C_{21}C_{11}^{-1}\text{sign}(\boldsymbol{\theta}_0(1))|$  and  $\mathbf{c} = (c_1, \dots, c_n)^\top = |C_{31}C_{11}^{-1}\text{sign}(\boldsymbol{\theta}_0(1))|$ . Now if we write  $\mathbf{z} = H_A^\top \boldsymbol{\epsilon}$  where  $H_A^\top = (h_1^a, \dots, h_q^a)^\top = C_{11}^{-1}(n^{-1/2}\mathbf{X}_{1,q}^\top)$ , then  $H_A^\top H_A = C_{11}^{-1}$ . Under the condition (B3), one has  $z_j = (h_j^a)^\top \boldsymbol{\epsilon}$  with  $\|h_j^a\|^2 \leq \frac{1}{M_2}$  for any  $j = 1, \dots, q$ . Similarly if we write  $\boldsymbol{\zeta} = H_B^\top \boldsymbol{\epsilon}$  where  $H_B^\top = (h_1^b, \dots, h_{p-q}^b)^\top = C_{21}C_{11}^{-1}(n^{-1/2}\mathbf{X}_{1,q}^\top - n^{-1/2}\mathbf{X}_{q+1,p}^\top)$ , then

$$H_B^\top H_B = \frac{1}{n} \mathbf{X}_{q+1,p}^\top \{I_n - \mathbf{X}_{1,q}(\mathbf{X}_{1,q}^\top \mathbf{X}_{1,q})^{-1} \mathbf{X}_{1,q}^\top\} \mathbf{X}_{q+1,p}.$$

Since  $I_n - \mathbf{X}_{1,q}(\mathbf{X}_{1,q}^\top \mathbf{X}_{1,q})^{-1} \mathbf{X}_{1,q}^\top$  has eigenvalues between 0 and 1, under condition (B2), one obtains  $\zeta_j = (h_j^b)^\top \boldsymbol{\epsilon}$  with  $\|h_j^b\|^2 \leq M_1$  for any  $1 \leq j \leq p-q$ . Similarly, we can write  $\boldsymbol{\xi} = H_C^\top \boldsymbol{\epsilon}$ , where  $H_C^\top = (h_1^c, \dots, h_n^c)^\top = C_{31}C_{11}^{-1}(n^{-1/2}\mathbf{X}_{1,q}^\top - I_n)$ , then  $H_C^\top H_C = I_n - \mathbf{X}_{1,q}(\mathbf{X}_{1,q}^\top \mathbf{X}_{1,q})^{-1} \mathbf{X}_{1,q}^\top$ , which indicates  $\xi_j = (h_j^c)^\top \boldsymbol{\epsilon}$  with  $\|h_j^c\|^2 \leq 1$  for any  $1 \leq j \leq n$ . Also notice that

$$|\frac{\lambda_n}{n}a_j| = \frac{\lambda_n}{n} |C_{11}^{-1}\text{sign}(\boldsymbol{\theta}_0(1))| \leq \frac{\lambda_n}{nM_2}$$

Under condition (A2), one obtains  $E(\zeta_i^{2k}) < \infty$  because given constant  $n$ -dimensional vector  $\boldsymbol{\alpha}$ ,  $E(\boldsymbol{\alpha}^\top \boldsymbol{\epsilon}^{2k}) \leq (2k-1)!! \|\boldsymbol{\alpha}\|^2 E(\epsilon_i^{2k})$ . For random variables  $Z$  with bounded  $2k$ 'th moments, the tail probability is bounded by  $P(Z > t) = O(t^{-2k})$ . Under condition (B4), for  $\lambda_n = o(n^{(d+1)/2})$ , we obtain

$$\sum_{j=1}^q P(|z_j| \geq \sqrt{n}(|\boldsymbol{\theta}_0(1)| - \frac{\lambda_n}{2n} a_j)) = O(n^{-kd}) = o(1)$$

$$\sum_{j=1}^{p-q} P(|\zeta_j| \geq \frac{\lambda_n}{2\sqrt{n}} b_j) = (p-q)O(\frac{n^k}{\lambda_n^{2k}}) = O(\frac{n^k}{\lambda_n^{2k}}) = o(1).$$

Notice that  $|\frac{\lambda_n}{2\sqrt{n}} c_j| \leq \frac{\lambda_n}{2M_2}$  and  $\mu_n \lambda_n^{-1} \kappa_n^{-1} \rightarrow \infty$ , which is indicated by  $\mu_n n^{-1/2k-d/2-1/2} \rightarrow \infty$ , we have

$$\sum_{i=1}^n P(|\xi_j| \geq \frac{\mu_n}{2|\tilde{\gamma}_i|} - \frac{\lambda_n}{2\sqrt{n}} c_i) = nO(\frac{\kappa_n^{2k}}{\mu_n^{2k}}) = o(1).$$

This completes the Theorem 1.

Proof of Corollary 1:

We will follow the same framework as the proof of Theorem 1 except changing the tail bound of the above three inequalities. Under condition (A1)(A2'), if there exists  $\lambda_n = Cn^{\frac{1+d_1}{2}}$  with  $0 < d_1 < d$  and  $\frac{\mu_n}{\sqrt{\log n \kappa_n}} \rightarrow \infty$ , we have

$$\sum_{j=1}^q P(|z_j| \geq \sqrt{n}(|\boldsymbol{\theta}_0(1)| - \frac{\lambda_n}{2n} a_j)) = o(e^{-n^d}) = o(1),$$

$$\sum_{j=1}^{p-q} P(|\zeta_j| \geq \frac{\lambda_n}{2\sqrt{n}} b_j) = o(e^{-n^{d_1}}) = o(1),$$

Notice that  $\mu_n \lambda_n^{-1} \kappa_n^{-1} \rightarrow \infty$ , which is implied by  $\mu_n n^{-1/2-d_1/2} (\log n)^{-1/2} \rightarrow \infty$ , one obtains

$$\sum_{j=1}^n P(|\xi_j| \geq \frac{\mu_n}{2|\tilde{\gamma}_j|} - \frac{\lambda_n}{2\sqrt{n}} c_j) = nO(e^{-\mu_n^2/\kappa_n^2}) = o(1),$$

which completes the proof of Corollary 1.

Proof of Theorem 2:

Suppose we contaminate  $m \leq n - h$  observations, and assume  $\tilde{\mathbf{y}}$  and  $\tilde{\mathbf{X}}$  are the contaminated data. Define  $\tilde{\boldsymbol{\beta}}$  as the solution using LTS when subset size is set as  $h$  and  $\tilde{\gamma}_i = \tilde{y}_i - \tilde{X}_i \tilde{\boldsymbol{\beta}}$ . Since the LTS with truncation number  $h$  have a breakdown point  $\min\{(n-h+1)/n, \lfloor (n-p)/2 \rfloor / n\}$ , we have  $\|\tilde{\boldsymbol{\beta}}\|_2 \leq M$  for some  $M > 0$ . As  $L_1$  norm and the Euclidean norm are topologically

equivalent, for any  $p$  dimensional vector  $\boldsymbol{\beta}$ , we have  $c_1\|\boldsymbol{\beta}\|_2 \leq \|\boldsymbol{\beta}\|_1 \leq c_2\|\boldsymbol{\beta}\|_2$ . Consequently, we obtain  $Q_n(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\gamma}}) = \lambda_n\|\tilde{\boldsymbol{\beta}}\|_1 + n\mu_n \leq c_2\lambda_n M + n\mu_n = M_1$ .

Recall that we solve the following minimization problem

$$\min_{\boldsymbol{\beta}, \boldsymbol{\gamma}} Q_n(\boldsymbol{\beta}, \boldsymbol{\gamma}) = \min_{\boldsymbol{\beta}, \boldsymbol{\gamma}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \boldsymbol{\gamma}\|_2^2 + \lambda_n \sum_{j=1}^p |\beta_j| + \mu_n \sum_{i=1}^n \frac{|\gamma_i|}{|\tilde{\gamma}_i|},$$

and suppose the minimizer of  $Q_n(\boldsymbol{\beta}, \boldsymbol{\gamma})$  is  $\hat{\boldsymbol{\beta}}$  and  $\hat{\boldsymbol{\gamma}}$ .

For any  $\|\boldsymbol{\beta}\|_2 \geq \frac{M_1+1}{\lambda_n c_1}$ , we have  $Q_n(\boldsymbol{\beta}, \boldsymbol{\gamma}) \geq \lambda_n\|\boldsymbol{\beta}\|_1 \geq c_1\lambda_n\|\boldsymbol{\beta}\|_2 \geq M_1 + 1 > Q_n(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\gamma}}) \geq Q_n(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}})$ . As a result,  $\|\hat{\boldsymbol{\beta}}\|_2 \leq \frac{M_1+1}{\lambda_n c_1}$ , which indicates the breakdown point is at least  $\min\{(n-h+1)/n, \lfloor (n-p)/2 \rfloor / n\}$ .

Proof of Corollary 3:

If we replace the initial LTS estimator and the residual by the corresponding SLTS estimator and residual, we can prove this corollary similarly as the proof of Theorem 2 because the SLTS estimator has a breakdown point of  $(n-h+1)/n$  [1].

Proof of Theorem 3:

Let  $\hat{\mathbf{u}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0 = (\mathbf{u}(1)^\top, \mathbf{u}(2)^\top, \mathbf{u}(3)^\top)^\top$  and  $\mathbf{u}^*(1) = (\mathbf{u}(1)^\top, \mathbf{u}(2)^\top)^\top$ . Define  $V_n(\mathbf{u}) = \sum_{i=1}^n [(\epsilon_i - \mathbf{A}\mathbf{u})^2 - \epsilon_i^2] + \sqrt{n}\mu_n\|\mathbf{u}(1) + \boldsymbol{\theta}_0(1)\|_1/|\tilde{\gamma}(1)| + \lambda_n\|\mathbf{u}(2) + \boldsymbol{\theta}_0(2)\|_1 + \sqrt{n}\mu_n\|\mathbf{u}(3)\|_1/|\tilde{\gamma}(2)|$ . The first summation in  $V_n(\mathbf{u})$  can be simplified as  $-2W(\sqrt{n}\mathbf{u}) + (\sqrt{n}\mathbf{u})^\top C(\sqrt{n}\mathbf{u})$ , where  $W = (W(1)^\top, W(2)^\top, W(3)^\top)^\top$ , with  $W(1) = \boldsymbol{\epsilon}_{1:s_n}$ ,  $W(2) = \mathbf{X}^\top \boldsymbol{\epsilon} / \sqrt{n}$  and  $W(3) = \boldsymbol{\epsilon}_{s_n+1:n}$ . Denote  $W^*(1) = (W(1)^\top, W(2)^\top)^\top$ . Then by definition we have:

$$\{\text{sign}(\hat{\boldsymbol{\theta}}_j) = \text{sign}(\boldsymbol{\theta}_{j0}), \text{ for } j = 1, \dots, s_n\} \in \{\text{sign}(\boldsymbol{\theta}_0(1))\hat{\mathbf{u}}(1) > -|\boldsymbol{\theta}_0(1)|\}$$

By uniqueness of Lasso solutions, if there exists  $\hat{\mathbf{u}}$  the following holds

$$\begin{aligned} \{C_{11}(\sqrt{n}\hat{\mathbf{u}}^*(1))\}_{1:s_n} - W(1) &= -\frac{\mu_n \text{sign}(\boldsymbol{\theta}_0(1))}{2 |\tilde{\gamma}(1)|} \\ |\hat{\mathbf{u}}(1)| &< |\boldsymbol{\theta}_0(1)| \\ -\frac{\lambda_n}{2\sqrt{n}} \mathbf{1} &\leq \{C_{11}(\sqrt{n}\hat{\mathbf{u}}^*(1))\}_{(s_n+1):(s_n+p)} - W(2) \leq \frac{\lambda_n}{2\sqrt{n}} \mathbf{1}. \\ -\frac{\mu_n}{2} \frac{\mathbf{1}}{|\tilde{\gamma}(2)|} &\leq C_{21}(\sqrt{n}\hat{\mathbf{u}}^*(1)) - W(3) \leq \frac{\mu_n}{2} \frac{\mathbf{1}}{|\tilde{\gamma}(2)|} \end{aligned}$$

Denote  $\boldsymbol{\varpi} = (\frac{\mu_n}{2} \{\frac{\text{sign}(\boldsymbol{\theta}_0(1))}{|\tilde{\gamma}(1)|}\}^\top, \frac{\lambda_n}{2\sqrt{n}} \mathbf{1}^\top)^\top$ . Substitute  $\hat{\mathbf{u}}^*(1)$  and bound the absolute values, the existence of such  $\hat{\mathbf{u}}$  is implied by

$$\begin{aligned} A_n &: |(C_{11})^{-1}W^*(1)|_{1:s_n} < \sqrt{n}|\boldsymbol{\theta}_0(1)| - \{(C_{11})^{-1}\boldsymbol{\varpi}\}_{1:s_n} \\ B_n &: |C_{21}C_{11}^{-1}W^*(1) - W(3)| \leq \frac{\mu_n}{2} \frac{\mathbf{1}}{|\tilde{\gamma}(2)|} - C_{21}C_{11}^{-1}\boldsymbol{\varpi} \end{aligned}$$

As a result,  $P(\hat{\gamma} =_s \gamma_0) \geq P(A_n \cap B_n)$ .

Meanwhile

$$1 - P(A_n \cap B_n) \geq \sum_{j=1}^{s_n} P(|z_j| \geq \sqrt{n}\{|\boldsymbol{\theta}_0(1)|\}_j - a_j) + \sum_{j=1}^{n-s_n} P(|\zeta_j| \geq b_j),$$

where  $\mathbf{z} = C_{11}^{-1}W^*(1) = (z_1, \dots, z_{s_n+p})^\top$ ,  $\boldsymbol{\zeta} = (\zeta_1, \dots, \zeta_{n-s_n})^\top = C_{21}C_{11}^{-1}W^*(1) - W(3)$ ,  $\mathbf{a} = (a_1, \dots, a_{s_n+p})^\top = C_{11}^{-1}\boldsymbol{\varpi}$ ,  $\mathbf{b} = (b_1, \dots, b_{n-s_n})^\top = \frac{\mu_n}{2} \frac{\mathbf{1}}{|\tilde{\gamma}(2)|} - C_{21}C_{11}^{-1}\boldsymbol{\varpi}$ . Now if we write  $\mathbf{z} = H_A^\top \boldsymbol{\epsilon}$ , then  $H_A^\top H_A = C_{11}^{-1}$ . Under the condition (C3), we have  $z_j = (h_j^a)'\boldsymbol{\epsilon}$  with  $\|h_j^a\|^2 \leq \frac{1}{M_4}$  for any  $j = 1, \dots, s_n$ . Similarly if we write  $\boldsymbol{\zeta} = H_B^\top \boldsymbol{\epsilon}$ , we have

$$H_B^\top H_B = \frac{1}{n} A_2^\top \{I - A_1(A_1^\top A_1)^{-1} A_1^\top\} A_2.$$

Since  $I - A_1(A_1^\top A_1)^{-1} A_1^\top$  is a projection matrix with eigenvalues 0 and 1, noticing that  $A_2^\top A_2 = C_{22} = nI_{n-s_n}$ , one obtains  $\zeta_j = (h_j^b)'\boldsymbol{\epsilon}$  with  $\|h_j^b\|^2 \leq 1$  for any  $j = 1, \dots, s_n + p$ . Also note that

$$|C_{11}^{-1}\boldsymbol{\varpi}| \leq \frac{1}{M_4} \|\boldsymbol{\varpi}\| \leq \frac{1}{2M_4} \left( \frac{\mu_n \sqrt{n}}{\delta_n} + \frac{\lambda_n}{\sqrt{n}} \right),$$

and by condition (B2), we have

$$|C_{21}C_{11}^{-1}\boldsymbol{\varpi}| \leq \frac{M_1}{M_4} \|\boldsymbol{\varpi}\| \leq \frac{M_1}{2M_4} \left( \frac{\mu_n \sqrt{n}}{\delta_n} + \frac{\lambda_n}{\sqrt{n}} \right),$$

When  $\mu_n = o(\frac{\pi_n \delta_n}{\sqrt{n}})$ ,  $\mu_n \kappa_n^{-1} n^{-1/2k} \rightarrow \infty$ ,  $\lambda_n = o(\sqrt{n} \pi_n)$ ,  $\lambda_n = o(\mu_n \kappa_n n^{1/2})$  and  $\sqrt{n} \kappa_n = o(\delta_n)$ , we have

$$\sum_{j=1}^{s_n} P(|z_j| \geq \sqrt{n}\{|\boldsymbol{\theta}_0(1)|\}_j - a_j) = s_n o(n^{-1}) = o(1)$$

$$\sum_{j=1}^{n-s_n} P(|\zeta_j| \geq b_j) = (n - s_n) o(\kappa_n^{2k} / \mu_n^{2k}) = o(1).$$

The conditions on  $\mu_n$  and  $\lambda_n$  can be implied by  $\mu_n = o(\frac{\pi_n^2}{\sqrt{n}})$ ,  $\mu_n n^{-1/k} \rightarrow \infty$ ,  $\lambda_n = o(\sqrt{n} \pi_n)$  and  $\lambda_n = o(\mu_n n^{1/2-1/2k})$ . Under condition (C1), we have  $\sqrt{n} \kappa_n = o(\delta_n)$ , which finishes the proof.

Proof of Corollary 2:

We will follow the same framework as the proof of Theorem 2 except changing the tail bound of the above two inequalities. When  $\mu_n = o(\frac{\pi_n \delta_n}{\sqrt{n}})$ ,  $\frac{\mu_n}{\kappa_n} \rightarrow \infty$ ,  $\lambda_n = o(\sqrt{n} \pi_n)$ ,  $\lambda_n = o(\mu_n \kappa_n n^{1/2})$

and  $\sqrt{n}\kappa_n = o(\delta_n)$ , we have

$$\sum_{j=1}^{s_n} P(|z_j| \geq \sqrt{n}\{\|\boldsymbol{\theta}_0(1)\}\}_j - a_j) = s_n o(e^{-\pi_n^2}) = o(1)$$

$$\sum_{j=1}^{n-s_n} P(|\zeta_j| \geq b_j) = (n - s_n) o(e^{-\frac{\mu_n^2}{\kappa_n^2}}) = o(1)$$

The conditions on  $\mu_n$  and  $\lambda_n$  can be implied by  $\mu_n = o(\frac{\pi_n^2}{\sqrt{n}})$ ,  $\sqrt{\log n}/\mu_n = O(1)$ ,  $\lambda_n = o(\sqrt{n}\pi_n)$  and  $\lambda_n = o(\mu_n \sqrt{n/\log n})$ . Under condition (C1'),  $\sqrt{n}\kappa_n = o(\delta_n)$ , which finishes the proof.