Summer 2013

Sparse group sufficient dimension reduction and covariance cumulative slicing estimation

Bilin Zeng

SPARSE GROUP SUFFICIENT DIMENSION REDUCTION AND COVARIANCE CUMULATIVE SLICING ESTIMATION

by

BILIN ZENG

A DISSERTATION

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

in

MATHEMATICS

2013

Approved by:

Dr. Xuerong Meggie Wen, Advisor
Dr. V. A. Samaranayake
Dr. Robert Paige
Dr. Akim Adekpedjou
Dr. Gregory Gelles

DEDICATION

I would like to dedicate this Doctoral dissertation to my parents: Guoqi Zeng and Yanming Chen. There is no doubt that I wouldn't be able to complete this effort without their love and support.

ABSTRACT

This dissertation contains two main parts:

In Part One, for regression problems with grouped covariates, we adopt the idea of sparse group lasso (Friedman et al., 2010) to the framework of the sufficient dimension reduction. We propose a method called the *sparse group sufficient dimension reduction* (sgSDR) to conduct group and within group variable selections simultaneously without assuming a specific model structure on the regression function. Simulation studies show that our method is comparable to the sparse group lasso under the regular linear model setting, and outperforms sparse group lasso with higher true positive rates and substantially lower false positive rates when the regression function is nonlinear or (and) the error distributions are non-Gaussian. One immediate application of our method is to the gene pathway data analysis where genes naturally fall into groups (pathways). An analysis of a glioblastoma microarray data is included for illustration of our method.

In Part Two, for many-valued or continuous Y, the standard practice of replacing the response Y by a discrete version of Y usually results in the loss of power due to the ignorance of intra-slice information. Most of the existing slicing methods highly reply on the selection of the number of slices h. Zhu et al. (2010) proposed a method called the cumulative slicing estimation (CUME) which avoids the otherwise subjective selection of h. In this dissertation, we revisit CUME from a different perspective to gain more insights, and then refine its performance by incorporating the intra-slice covariances. The resulting new method, which we call the covariance cumulative slicing estimation (COCUM), is comparable to CUME when the predictors are normally distributed, and outperforms CUME when the predictors are non-Gaussian, especially in the existence of outliers. The asymptotic results of COCUM are also well proved.

ACKNOWLEDGMENTS

First, I would like to express my deepest gratitude to my Ph.D advisor Dr. Xuerong Meggie Wen for her endless support and patient guidance to develop this dissertation. I am so grateful to Dr. Wen for her enthusiasm exposing me to the interesting research topics of sufficient dimension reduction. I am also sincerely thankful to Dr. V. A. Samaranayake for directing me in course work and research in the first few years of the program; for his exceptional guidance on my job hunting; and for providing me the opportunity to work for SEQL Workshop. I appreciate Dr. Akim Adekpedjou for his great advice and strong support in my job hunting process. I also want to acknowledge Dr. Robert Paige and Dr. Gregory Gelles for their advice rendered during the past few years. In addition, I would like to give special thanks to Dr. Gayla Olbricht for her unique help on the section of gene data analysis in this dissertation. Moreover, I want to thank all faculty members and staff of the Department of Mathematics and Statistics, Missouri University of Science and Technology, for their help and advice in many ways. In addition, I would like to thank my office mates, Leslie Malott, Julius Heim, Elizabeth Stahlman-King, Brittany Whited; all of my Chinese friends, American friends and all international friends I had the honor to meet during the last five years. Last but not least, I would like to thank my parents, grandparents and Ding Chu for their counsel and support.

TABLE OF CONTENTS

Page

ABSTRACT ... iv

ACKNOWLEDGMENTS ... v

LIST OF ILLUSTRATIONS ... viii

LIST OF TABLES ... ix

SECTION

1. INTRODUCTION ... 1
    1.1. CURSE OF DIMENSIONALITY ... 1
    1.2. HIGH DIMENSIONAL DATA ANALYSIS ... 1
    1.3. SUFFICIENT DIMENSION REDUCTION (SDR) ... 3
    1.4. LITERATURE REVIEW ... 6
        1.4.1. Global Methods ... 6
        1.4.2. Local Methods ... 10
    1.5. EXTENSION ... 12
2. VARIABLE SELECTION WITHIN SDR ... 15
    2.1. VARIABLE SELECTION ... 15
        2.1.1. Test-Based Methods ... 15
        2.1.2. Shrinkage Methods ... 15
    2.2. MODEL FREE VARIABLE SELECTION ... 17
        2.2.1. Model Free Test-Based Methods ... 18
        2.2.2. Model Free Shrinkage Methods ... 20
3. SPARSE GROUP SUFFICIENT DIMENSION REDUCTION ... 23
    3.1. LITERATURE REVIEW ... 23
    3.2. SPARSE GROUP SUFFICIENT DIMENSION REDUCTION ... 27
        3.2.1. Methodology ... 27
        3.2.2. Selection of Tuning Parameters ... 30
            3.2.2.1. Cross validation ... 30
            3.2.2.2. Modified bayesian information criterion ... 32
        3.2.3. Simulation ... 34
            3.2.3.1. Model I ... 34
            3.2.3.2. Model II ... 35
            3.2.3.3. Model III ... 37
            3.2.3.4. Model IV ... 37
    3.3. GENE PATHWAY DATA ANALYSIS ... 40

3.3.1. Basics of Genetics 40
3.3.2. Microarray Technology 42
3.3.3. Gene Pathway Analysis Reviews 44
3.3.4. A Real Data Analysis 45
3.4. DISCUSSION 49
4. A NOTE ON CUMULATIVE MEAN ESTIMATION 50
4.1. INTRODUCTION 50
4.2. CUMULATIVE SLICING ESTIMATION (CUME) 51
4.3. THE ENSEMBLE ESTIMATORS' APPROACH 53
4.4. CUME REVISIT: THE ENSEMBLE ESTIMATORS' APPROACH 55
4.5. COVARIANCE INVERSE REGRESSION ESTIMATION (CIRE) 56
4.6. COVARIANCE CUMULATIVE SLICING ESTIMATION (COCUM) 58
4.6.1. The Method 58
4.6.2. The Asymptotic Properties 60
4.6.3. The Determination of d 68
4.7. SIMULATION STUDIES 70
4.7.1. Model I 71
4.7.2. Model II 72
4.7.3. Model III 73
4.7.4. Model IV 75
4.7.5. Model V 76
4.7.6. Model VI 77
4.7.7. Model VII 79
4.8. CONCLUSION 81

APPENDICES

A. MATLAB ALGORITHM: sgSDR 84
B. MATLAB ALGORITHM: COCUM 100
BIBLIOGRAPHY 105
VITA 115

LIST OF ILLUSTRATIONS

Figure Page

3.1 Structure of the DNA Molecule 41

3.2 Pairing Base 41

3.3 Chromosomes: Human 42

3.4 Gene 42

3.5 Affymetrix Gene Chip Microarray 43

3.6 Hybridization of mRNA Sample to the Array: 1 44

3.7 Hybridization of mRNA Sample to the Array: 2 44

3.8 Glioblastoma 46

4.1 The graph of y: Model I - Model III 82

4.2 The graph of y: Model IV - Model VII 83

LIST OF TABLES

Table Page

3.1 sgSDR: Linear Model I With Gaussian Error 35

3.2 sgSDR: Linear Model II With Cauchy Error and Correlated Predictors 36

3.3 sgSDR: Linear Model III With Cauchy Error 38

3.4 sgSDR: Nonlinear Model IV with Gaussian and Cauchy Error 39

3.5 Pathway Selections for Glioblastoma Data 48

4.1 COCUM: Model I 72

4.2 COCUM: Model I Dimension Estimate 72

4.3 COCUM: Model II 73

4.4 COCUM: Model II Dimension Estimate 74

4.5 COCUM: Model III 74

4.6 COCUM: Model III Dimension Estimate 75

4.7 COCUM: Model IV 76

4.8 COCUM: Model IV Dimension Estimate 76

4.9 COCUM: Model V 77

4.10 COCUM: Model V Dimension Estimate 78

4.11 COCUM: Model VI 78

4.12 COCUM: Model VI Dimension Estimate 79

4.13 COCUM: Model VII 80

4.14 COCUM: Model VII Dimension Estimate 80

1. INTRODUCTION

1.1. CURSE OF DIMENSIONALITY

With the fast development of technology and information society, high dimensional data has become an issue that can arise in every scientific field, for example, the analysis of genetic data and some of the economic models. However, it is commonly known that it is difficult to analyze and organize the high dimensional data due to the curse of dimensionality (Bellman, 1961). The basic idea of curse of dimensionality is when the dimensionality increases, the volume of the space increases so fast that the available data become sparse. In other words, the sparsity increases exponentially given a fixed amount of sample data points. Intuitively, much larger data sets are needed to achieve the same accuracy even under moderate dimension case. However, this is impractical in reality due to limited sample sizes. This becomes even more difficult when the number of predictors p exceeds the number of observations n. A typical example of this phenomena can arise in genetic association studies while one faces thousands and millions of genes with only tens or hundreds of sample size. From Fisher's point of view (Fisher, 1924), large sample regression methods are appropriate only when the sample size n is much larger than the number of predictors p, preferably more than one thousand. So dimension reduction was an issue even during Fisher's era and before. Nowadays, how to deal with the high dimensional data has become a popular topic for statisticians.

1.2. HIGH DIMENSIONAL DATA ANALYSIS

Several methodologies have been developed to address the issue of curse of dimensionality. There are essentially two approaches: Function Approximation and Dimension Reduction. The former assumes that the regression function is a sum of univariate smooth functions (Xia, Tong, Li and Zhu, 2002) which includes additive model approach (Hastie and Tibshirani, 1986) and the projection pursuit regression (Friedman and Stuetzle, 1981). However, most of the function approximation methods seem to have singled out the approximation of the regression function as their objective (Li, 1991). As indicated by Li

(1991), dimension reduction in statistics has a wider scope than functional approximation. Hence, dimension reduction methods became more popular after 1990. As it is defined in the dictionary, dimension reduction is the process of reducing the number of random variables under consideration, and can be divided into feature selection and feature extraction. Feature selection aims to find out a subset of the original predictors, and many of the variable selection methods are designed to achieve this goal. On the other hand, feature extraction reduces the dimension of space by looking for the linear combination of the original variables to obtain the most important information. Dimension reduction has been used for data visualization and there are many different types of dimension reduction methods.

Next we will categorize some of the classical dimension reduction methods from two different perspectives. According to Zhu (2011), one way of classifying dimension reduction methods depends on whether the training data is labeled (supervised) or not (unsupervised). A major difference between the supervised dimension reduction and unsupervised dimension reduction is the former makes use of predictor information which is related to the response variable while the latter leaves out the information from the response variables. Unsupervised dimension reduction includes the traditional Principle Component Analysis (PCA) and Singular Value Decomposition (SVD), while the supervised dimension reduction includes Ordinary Least Square (OLS) and Partial Least Square (PLS).

The other way of classifying dimension reduction methods is based on the geometric structure point of views (Burges, 2009), which includes the projective methods and manifold learning methods. The idea of projective methods is to project the high dimensional data into a lower dimensional subspace that can capture the information from the data. This is also the main idea of most *sufficient dimension reduction* methods which we will discuss latter in this article. Meanwhile, Principle Component Analysis (PCA) (Pearson, 1901), Kernel PCA (Schlkopf et al., 1998), Probabilistic PCA (Tipping and Bishop, 1999A; Tipping and Bishop, 1999B), Canonical Correlation analysis (Hotelling, 1936), Oriented PCA (Diamantaras and Kung, 1996) all belong to this category. In Manifold learning methods, there are several methods such as Landmark MDS (Silva and

Tenenbaum, 2002), Locally Linear Embedding (LLE) (Roweis and Saul, 2000), Laplacian Eignmaps (Belkin and Niyogi, 2003) and Spectral Clustering (Shi and Malik, 2000; Meila and Shi, 2000; Ng et al., 2002). Among many dimension reduction methods, sufficient dimension reduction has been widely used during recent years. As we stated before, we will work within the framework of sufficient dimension reduction in this dissertation.

1.3. SUFFICIENT DIMENSION REDUCTION (SDR)

The phrase "sufficient dimension reduction" with its modern meaning was introduced in 1990's (Li, 1991; Cook, 1998a; Cook and Yin, 2001) in the context of regression graphics. A common sufficient dimension reduction objective is to reduce the dimension of predictors **X** without loss of information on the regression and without requiring a pre-specified parametric model. For a typical regression problem with a univariate random response Y and a p-dimensional random vector **X**,

$$Y = g(\boldsymbol{\beta}^T \mathbf{X}, \epsilon) \tag{1.1}$$

where g is an unknown link function, sufficient dimension reduction (SDR: Li, 1991; Cook and Weisberg, 1991; Cook, 1998) aims to reduce the dimension of **X** without loss of information on the regression and without requiring a pre-specified parametric model. The terminology "sufficient" is similar to Fisher's classical definition of *sufficient statistic*: if **D** represents the data, then a statistic $t(\mathbf{D})$ is sufficient about θ if

$$\mathbf{D}|(\theta, t) \sim \mathbf{D}|t.$$

In sufficient dimension reduction, the "sufficiency" is defined similarly: there exists $\boldsymbol{\eta} \in \mathbb{R}^{p \times q}$, $q \leq p$ such that:

$$Y|\mathbf{X} \sim Y|(\eta_1^T \mathbf{X}, \eta_2^T \mathbf{X}, ..., \eta_q^T \mathbf{X}) \sim Y|\boldsymbol{\eta}^T \mathbf{X} \tag{1.2}$$

where $\boldsymbol{\eta} = [\eta_1, \eta_2, ..., \eta_q]$. The existence of $\boldsymbol{\eta}$ is clear since $\boldsymbol{\eta}$ can be a p by p identity matrix. There are some expressions equivalent to (1.2). For example, (1.2) can be interpreted as Y depends on $\mathbf{X}$ only through q dimensional subspace, that is,

$$Y \perp\!\!\!\perp \mathbf{X} \mid \boldsymbol{\eta}^T \mathbf{X} \tag{1.3}$$

where $\perp\!\!\!\perp$ indicates independence. Also, it is equivalent to

$$Y \perp\!\!\!\perp \mathbf{X} \mid P_{\mathcal{S}\{\boldsymbol{\eta}\}} \mathbf{X} \tag{1.4}$$

where $P_{\mathcal{S}\{\boldsymbol{\eta}\}}$ denotes the projection operator for $\mathcal{S}\{\boldsymbol{\eta}\}$; and $\boldsymbol{\eta}^T\mathbf{X}$ is a lower dimensional projection of $\mathbf{X}$ onto a subspace $S \subseteq R^p$ without the loss of information on the regression. The subspace

$$\mathcal{S}\{\boldsymbol{\eta}\} = \text{span}\{\eta_1, \eta_2, ..., \eta_q\}$$

mentioned above is defined as a *dimension reduction subspace.* However, the dimension reduction subspace is not unique since any space that contains the dimension reduction subspace is also a dimension reduction subspace. So we are looking for the smallest dimension reduction subspace which is the intersection of all the possible dimension reduction subspaces, defined as

$$\mathcal{S}_{Y|\mathbf{X}} = \cap \mathcal{S}_{DRS}$$

and is called the *central subspace.* And the dimension of the central subspace

$$\dim(\mathcal{S}_{Y|\mathbf{X}}) = d$$

is called the structural dimension. The central subspace is well defined under very mild condition (Cook, 1996; Yin, Li and Cook, 2008). We assume the existeness of the central subspace through this dissertation. Particularly, if $d = 0$, then the response is independent of all the predictors. If $d = 1$, a possible model is

$$Y|\mathbf{X} = \mu(\beta^T\mathbf{X}) + \sigma(\beta^T\mathbf{X})\epsilon,$$

where $\epsilon \perp\!\!\!\perp \mathbf{X}$, $\beta \neq 0$ and $\epsilon \sim N(0,1)$, such model is called the *Single-index Model (SIM)* (Wang, Xu and Zhu, 2012). If $d = 2$, then the model is

$$Y|\mathbf{X} = \mu(\beta_1^T\mathbf{X}, \beta_2^T\mathbf{X}) + \sigma\epsilon,$$

for example,

$$Y = (x_1 + x_2 + 1)(2x_3 - x_4) + \sigma\epsilon.$$

The basic idea of sufficient dimension reduction is to replace the predictors $\mathbf{X} \in \mathbb{R}^p$ with a lower dimensional projection $P_{\mathcal{S}}\mathbf{X}$ onto a subspace $\mathcal{S} \subseteq \mathbb{R}^p$ without the loss of information on the original regression of $Y|\mathbf{X}$. Subsequent modeling and prediction can then be built upon the reduced dimensional projection.

A short example will be illustrated here to demonstrate the concepts of central subspace and the structural dimension. Suppose the true model is:

$$Y = exp(0.75(X_1 + X_2 + 1)(2X_3 - X_4) + 1) + 0.5\epsilon$$

where $\mathbf{X} = (X_1, X_2, \ldots, X_{10})^T$, and $\epsilon \perp\!\!\!\perp \mathbf{X}$. Then, the central subspace is

$$\boldsymbol{\eta}^T = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & \ldots & 0 \\ 0 & 0 & 2 & -1 & 0 & \ldots & 0 \end{pmatrix}$$

and the structural dimension d is equal to 2. The main idea for the sufficient dimension reduction in this example is: we can replace the original 10 dimensional predictors $\mathbf{X}$ with the two dimensional linear combination of all predictors $\boldsymbol{\eta}^T\mathbf{X}$ to achieve the goal of dimension reduction without the loss of information on the regression. Apparently, in this example,

$$\boldsymbol{\eta}^T\mathbf{X} = \begin{pmatrix} X_1 + X_2 \\ 2X_3 - X_4 \end{pmatrix}$$

The concept of *central mean subspace* was first introduced by Cook and Li in 2002, which is a notion similar to the central subspace. The central mean subspace is designed to give a complete picture of the relationship between $\mathbf{X}$ and $E(Y|\mathbf{X})$ instead of $\mathbf{Y}$. A similar definition as in (1.4), if

$$Y \perp\!\!\!\perp E(\mathbf{Y} \mid \mathbf{X}) \mid P_{\mathcal{S}\{\boldsymbol{\eta}\}}\mathbf{X}, \tag{1.7}$$

then the intersection of all subspaces in (1.7) is defined as central mean subspace, denoted by $\mathcal{S}_{E(Y|\mathbf{X})}$. Since $E(Y|\mathbf{X})$ is a function of $\mathbf{X}$, it is obvious that (1.4) implies (1.7) and consequently $\mathcal{S}_{E(Y|\mathbf{X})}$ is a subspace of $\mathcal{S}_{Y|\mathbf{X}}$. Yin and Cook (2002) proposed the *central kth-moment dimension reduction subspace* as an extended central space which is based on $E(\mathbf{Y}^k|\mathbf{X})$. It was pointed out in Zhu and Zeng (2006) that, the central mean dimension reduction subspace for $E(e^{it\mathbf{Y}}|\mathbf{X})$ $(t \in \mathbb{R})$, when put together, recovers the central subspace. Yin and Li (2011) pointed out the fact that estimating the central mean subspaces $E[f(\mathbf{X})|Y]$ for a rich enough family of functions f is equivalent to estimating the central subspace $\mathcal{S}_{Y|\mathbf{X}}$ itself. The ensemble approach introduced by Yin and Li (2011) nicely built a nice bridge between the central mean subspace and the central subspace.

1.4. LITERATURE REVIEW

From the introduction, we can easily see that the goal of sufficient dimension reduction is to estimate and make statistical inferences about the central subspace or the central mean subspace and the structural dimension. Many methods have been developed to estimate $\mathcal{S}_{Y|\mathbf{X}}$ or $\mathcal{S}_{E(Y|\mathbf{X})}$. Essentially, there are global and local sufficient dimension methods. And we will first have an overview of these two categories and then provide details on some of the popular methods.

1.4.1. Global Methods. Most of the global sufficient dimension reduction methods can be classified into three categories (Wang, 2009): forward regression based, inverse moment based and joint moment methods. A typical example of joint moment methods is Principal Hessian Directions (pHd) (Li, 1992; Cook, 1998), which is based on the third moment matrix

$$E[(\mathbf{Y} - E(\mathbf{Y}))ZZ^T].$$

Forward regression based methods include the classical Ordinary Least Square (OLS) (Duan and Li, 1991), Fourier methods (Zhu and Zeng, 2006) and so on. Inverse regression methods include Slice Inverse Regression (SIR) (Li, 1991; Hsing and Carrol, 1992; Zhu and Ng, 1995), kernel estimate of SIR (Fang and Zhu, 1996), parametric inverse regression (Bura and Cook, 2001), inverse third moments (Yin and Cook, 2003), sliced average variance estimation (SAVE) (Cook and Weisberg, 1991), Directional regression (DR) (Li and Wang, 2007) and inverse regression based on minimum discrepancy approach (Cook and Ni, 2005). In contrast to high-dimensional forward regression models, inverse regression has the substantial advantage of avoiding the high dimensional predictors $\mathbf{X}$ by literally dealing with a one dimension to one dimension regression problem (Li, 1991). Sliced Inverse Regression (SIR) (Li, 1991) is so far one of the most popular sufficient dimension reduction methods. It is also a typical example of a method involving some form of spectral decomposition.

The general idea of spectral decomposition approach is (Wen and Cook, 2007): first find a symmetric population kernel matrix M such that

$$\text{span}(M) \subseteq \mathcal{S}_{Y|\mathbf{X}},$$

then spectrally decompose a consistent estimate $\widehat{M}$, and finally use the span of eigenvectors corresponding to the d largest eigenvalues of $\widehat{M}$ to estimate span(M). In this process, two critical issues are the determination of the structural dimension d and the estimation of the kernel matrix $\widehat{M}$ on a sample level. Li (1991), Schott (1994) and Bura and Cook (2001) have discussed details on the determination of structural dimension d. About the estimate of $\widehat{M}$, some nonparametric methods such as kernel and smoothing splines could be used. In SIR (Li, 1991), the idea of "slicing" which is proposed to estimate the kernel matrix is very simple and effective. Next we introduce the main idea of SIR.

Consider a general model as (1.1), and assume the structural dimension d is known. First of all, the development of most sufficient dimension reduction methods relies on a crucial condition imposed on the marginal distribution of $\mathbf{X}$, the so called *linearity condition*:

$$E(\mathbf{X}|\boldsymbol{\beta}^T\mathbf{X}) = P^T_{\boldsymbol{\beta}(\Sigma_{XX})}\mathbf{X} \tag{1.8}$$

where

$$\Sigma_{XX} = \mathbb{C}\text{ov}(\mathbf{X}),$$

$$P_{\boldsymbol{\beta}(\Sigma_{XX})} = \boldsymbol{\beta}(\boldsymbol{\beta}^T\Sigma_{XX}\boldsymbol{\beta})^{-1}\boldsymbol{\beta}^T\Sigma_{XX}$$

is the orthogonal projection operator w.r.t. inner product $(a,b)_\Sigma = a^T\Sigma b$ and $\boldsymbol{\beta}$ is an orthonormal basis for $\mathcal{S}_{Y|\mathbf{X}}$. This is a very mild condition. Hall and Li (1993) showed that when the dimension of $\mathbf{X}$ is large, for most directions β even a highly nonlinear regression is still nearly linear. In general, when $\mathbf{X}$ is elliptically symmetrically distributed, for example, $\mathbf{X}$ follows a multivariate normal distribution, the linearity condition holds (Eaton, 1986). The condition can also be induced by predictor transformation (Li and Yin, 2008), reweighting (Cook and Nachtsheim, 1994), and clustering (Li, Cook, and Nachtsheim, 2004). In addition, Linearity condition is imposed on the marginal distribution of X instead of the conditional distribution of $\mathbf{Y}|\mathbf{X}$ under the traditional regression modeling, and hence no pre-information about the link function is required (model free). Hall and Li (1993) considered the case when (1.8) is severely violated, and had a discussion about how to detect this violation using SIR.

In SIR, for simplicity, all predictors $\mathbf{X}$ are standardized as

$$\mathbb{Z} = \Sigma_{XX}^{-1/2}\{\mathbf{X} - E(\mathbf{X})\}.$$

Let $\nu = (\nu_1, \nu_2, \ldots, \nu_d)$ be an orthonormal basis for $\mathcal{S}_{Y|\mathbb{Z}}$, we have

$$\mathcal{S}_{Y|\mathbf{X}} = \Sigma_{XX}^{-1/2}\mathcal{S}_{Y|\mathbb{Z}}$$

and

$$\nu_i = \Sigma_{XX}^{1/2}\beta_i.$$

SIR (Li, 1991) showed that

$$\text{span}\{M\} \subseteq \mathcal{S}_{Y|\mathbb{Z}},$$

where the kernel matrix

$$M = \mathbb{C}\text{ov}\left(E(\mathbb{Z}|\mathbf{Y})\right).$$

On the sample level, SIR (Li, 1991) estimates this kernel matrix M by slicing the continuous response $\mathbf{Y}$ into H pieces with N_h observations in the h^{th} $(h = 1, 2, \ldots H)$ slice, and takes

$$\widehat{M} = \sum_{h=1}^{H} \widehat{f}_h \widehat{\mathbb{Z}}_h \widehat{\mathbb{Z}}_h^T$$

where the slice weight

$$\widehat{f}_h = \frac{N_h}{n}$$

and the standardize sample mean of each slice

$$\widehat{\mathbb{Z}}_h = \widehat{\Sigma}_{XX}^{-1/2}\{\mathbf{X}_h - \bar{\mathbf{X}}_h\}.$$

Here $\bar{\mathbf{X}}_h$ and the $\mathbf{X}_h$ are the sample mean and the design matrix in the h^{th} slice respectively. Then apply the spectral decomposition on $\widehat{M}\widehat{\nu}_i = \widehat{\lambda}_i\widehat{\nu}_i$ $(i = 1, \ldots, d)$ to find the the standardized central subspace which is the space spanned by the first d eigenvectors $\widehat{\nu}_1, \widehat{\nu}_2, \ldots, \widehat{\nu}_d$ corresponding to eigenvalues $\widehat{\lambda}_1 \geq \widehat{\lambda}_2 \geq \ldots \geq \widehat{\lambda}_d$. Finally, the central subspace can be obtained by

$$\widehat{\beta}_i = \widehat{\Sigma}_{XX}^{-1/2}\widehat{\nu}_i.$$

SIR is widely applied in the literature due to its simplicity and computational feasibility. SIR is also very robust to the selection of the number of slices (Li and Zhu, 2007). Details on this issue will be addressed in the later sections. However, there are two obvious limitations of SIR. Like most of the sufficient dimension reduction methods, SIR might fail under $n < p$ scenarios because the covariance matrix Σ_{XX} is not invertible. The other limitation of SIR is that it might fail to recover the central subspace when $E(\mathbf{X}|\mathbf{Y})$ degenerates. For example, when the link function is symmetric or $\mathbf{X}$ and β are symmetric about zero, i.e. $E(\mathbf{X}|\mathbf{Y}) = 0$, then there is no information we can obtain from M, then $\hat{\beta}$ is hence a poor estimate of β. In that case, second or higher moments based methods might be appropriate to use. Some inverse moment based methods such as sliced average variance estimation (SAVE) (Cook and Weisberg, 1991) which uses

$$M = E\{I_p - Var(\mathbb{Z}|\mathbf{Y})\}$$

as its kernel matrix is developed to address this issue. However, SAVE is not very efficient in estimating monotone trends for small to moderate sample size (Li and Wang, 2007). Directional regression (DR) (Li and Wang, 2007) is proposed to overcome these difficulties by using

$$2E\{E^2(\mathbb{Z}\mathbb{Z}^T|\mathbf{Y})\} + 2E^2\{E(\mathbb{Z}|\mathbf{Y})E(\mathbb{Z}^T|\mathbf{Y})\}$$

$$+2\mathrm{E}\{\mathrm{E}(\mathrm{Z}^T|\mathbf{Y})E(\mathbb{Z}|\mathbf{Y})\}E\{E(\mathbb{Z}|\mathbf{Y})E(\mathbb{Z}^T|\mathbf{Y})\} - 2I_p$$

as its kernel matrix.

1.4.2. Local Methods. However, all the above methods normally require a relatively large sample size and a specified distribution of the predictor $\mathbf{X}$ (linearity condition). Kernel based methods are developed to address the above issues. Minimum average variance estimation (MAVE) (Xia et al. 2002; Xia, 2007) and Slice Regression (Wang and Xia, 2008) both belong to local sufficient dimension reduction methods which use kernel smoothing techniques. As one of the earliest and most fundamental methods, MAVE is widely applied in other areas such as time series, economics and bioinformatics. The idea of MAVE is motivated by SIR (Li, 1991), average derivative estimation (ADE) (Hardle and Stoker, 1989) and spline smoothing methods. Here, we briefly discuss the idea of MAVE method.

Consider a general model as shown on (1.1) with zero mean for the error term. The objective function of MAVE is

$$E\{\mathbf{Y} - E(\mathbf{Y}|B^T\mathbf{X})\}^2 \tag{1.9}$$

subject to $BB^T = I$ and the goal is to minimize (1.9) in order to get the solution $\mathbf{B} = (\beta_1, \ldots, \beta_d)$ as the basis for central mean subspace. On the sample level, following the idea of local linear smoothing estimation, MAVE approximates (1.9) as

$$\sigma_B^2(B^T\mathbf{X}_0) = \sum_{i=1}^{n} [\mathbf{Y}_i - \{a_0 + b_0^T B^T(\mathbf{X}_i - \mathbf{X}_0)\}]^2 \omega_{io} \tag{1.10}$$

Here $a_0 + b_0^T B^T(\mathbf{X}_i - \mathbf{X}_0)$ is the local linear expansion of $E(\mathbf{Y}_i|B^T\mathbf{X}_i)$ at point $\mathbf{X}_0$ and the weights ω_{io} is defined as

$$K_h\{B^T(\mathbf{X}_i - \mathbf{X}_0)\}/\sum K_h B^T(\mathbf{X}_l - \mathbf{X}_0)\}$$

with $\sum_{i=1}^{n} \omega_{io} = 1$, where $K_h(\cdot)$ is a p-dimensional kernel function with bandwidth h. Xia et al. (2002) gave two choices of ω_{io} by using a multidimensional kernel weight and a refined kernel weight. Under mild conditions (Xia et al., 2002), it can be shown that

$$\sigma_B^2(B^T\mathbf{X}_0) - \hat{\sigma}_B^2(B^T\mathbf{X}_0) = O_p(1).$$

Therefore the original objective function can be obtained by

$$min_{\substack{B:BB^T=I \\ a_j, b_j, j=1,\ldots,n}} (\sum_{j=1}^{n} \sum_{i=1}^{n} [\mathbf{Y}_i - \{a_j + b_j^T B^T(\mathbf{X}_i - \mathbf{X}_j)\}]^2 \omega_{ij}) \tag{1.11}$$

where $b_j^T = (b_{j1}, \ldots, b_{jd})$.

Considering the prior group information, the group wise dimension reduction procedure proposed by Li, Li and Zhu (2010) is actually applying idea of MAVE (Xia et al., 2002) into groupwise dimension reduction. Details about Li, Li and Zhu (2010) will be provided in the next section. MAVE works better for a small sample size and it is able to estimate the central mean subspaces exhaustively (Yin and Li, 2011). Moreover, it does not require the linearity condition. However, MAVE is sensitive to extreme values (Wang and Xia, 2008) and can infer only about the central mean subspace, which means MAVE only gives limited central subspace information. To overcome such limitation, density MAVE (DMAVE) (Xia, 2007) and Sliced Regression (Wang and Xia, 2008) are introduced. On the other hand, MAVE requires kernel smoothing and is computationally slow (Li, Zha and Chiaromonte, 2005). Unlike global methods and local methods, Li, Zha and Chiaromonte (2005) proposed a new SDR approach by estimating contour directions of small variations in the response.

1.5. EXTENSION

The future research and application of sufficient dimension reduction are very useful and promising. Despite many existing successful methods based on regular settings, there are many extensions under the framework of sufficient dimension reduction. In this section, we give a brief review on several important extensions.

- Response

  First of all, we can consider a complex form of responses. Most of the methods we mentioned earlier are only applicable to univariate response. However, it is very common that the response might be in a special structure, such as multidimensional structure. Cook and Setodji (2003) proposed a model free test of dimension for reduced rank in multivariate regression. Li, Wen and Zhu (2008) proposed a projective resampling method for dimension reduction with multivariate responses, and Zhu, Zhu and Wen (2010) also worked on multivariate regression problem under the framework of dimension reduction.

- Predictors

  Intuitively, we can consider some complex forms for predictors, such as categorical predictors (Li, Cook and Chiaromonte, 2003; Wen and Cook, 2007) or predictor with a matrix structure (Li, Kim and Altman, 2010). More commonly in genetic analysis, predictors can be in group structures. It is wildly recognized that many of the biological units naturally fall into groups. So the incorporation of prior group information can greatly increase the statistical efficiency. Recently, Li et al. (2010) proposed a groupwise dimension reduction which incorporates the prior group information when the predictors under investigation fall naturally into several groups. However, this method fails under the $n < p$ case. The first part of this dissertation provides a possible solution by conducting variable selection within sufficient dimension reduction for grouped predictors when $n << p$. More details will be provided in Section 3.

- Nonlinear SDR

  As we mentioned earlier, the central subspace is always composed by a linear combination of all predictors, but it is possible the structure is nonlinear. This normally happens especially when the predictors are in a complex format or $n < p$. So many nonlinear sufficient dimension reduction methods (Wong and Li, 1992; Wu, Liang and Mukherjee, 2010; Zhu and Li, 2011; Li, Artemiou and Li, 2011) are and will be developed to address this issue.

- Discriminant Analysis

  The linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) are the most popular existing methods for discrimination. However, the idea of sufficient dimension reduction is also useful for discriminate analysis (Cook and Yin, 2001). Let's assume $\mathbf{Y}$ as a discrete random variable taking C distinct values to indicate the C classes. According to the idea of discriminant analysis, we need to assign $\mathbf{X}$ to the class having the largest posterior probabilities $Pr(\mathbf{Y} = c|\mathbf{X} = x)$. Theorically, we define the discriminant subspace (DS) S such that

  $$D(\mathbf{Y}|\mathbf{X}) = D(\mathbf{Y}|P_S\mathbf{X}),$$

  where

  $$D(\mathbf{Y}|\mathbf{X}) = argmax_c Pr(\mathbf{Y} = c|\mathbf{X}).$$

  The central discriminant subspace (CDS) $S_{D(\mathbf{Y}|\mathbf{X})}$ is the intersection of all discriminant subspace. The CDS is naturally a subset of the central subspace for the regression of $\mathbf{Y}$ on $\mathbf{X}$. The structure of SIR and SAVE suggests that they might be quite useful for constructing summary plots in discriminant analysis (Cook and Yin, 2001). For example, SIR is equivalent to LDA and SAVE is equivalent to QDA in the population for the purpose of constructing principal predictors and summary plots. Cook and Yin (2001) reviewed on graphical methods that can be viewed as pre-processors, aiding the analyst's understanding of the data and the choice of a final classifier.

- Variable Selection

  Last but not the least, we can incorporate variable selection methods into the framework of sufficient dimension reduction in order to construct **model free** variable selections. The idea of combining variable selection methods and sufficient dimension reduction methods has become more popular especially when the model is sparse. More details will be discussed in the following section.

2. VARIABLE SELECTION WITHIN SDR

2.1. VARIABLE SELECTION

Variable selection, also known as feature selection in machine learning, is the process of selecting a subset of relevant predictor variables for use in model construction. A simple model is always easier for interpretation. Removing the excess variables not only reduces the noise to the precise estimation (Tong, 2010), but also alleviates the collinearity issue caused by having too many predictors (Fan and Lv, 2009). Moreover, this can greatly save computation cost caused by high dimensional data. As one of the most important dimension reduction approaches, many variable selection approaches have been developed. Essentially, there are two types of methods: test based variable selection methods and shrinkage methods.

2.1.1. Test-Based Methods. Traditional best subset selection is one of the classical test based variable selection methods. The basic idea of best subset selection is to select a subset that can optimize a specified criterion. One can use forward, backward or stepwise regression methods as the pattern of subset changing at each step. Many criterion have been proposed for choosing the best subset such as: Akaike Information Criterion (AIC) (Akaike, 1973), Bayesian Information Criterion (BIC) (Schwarz, 1978), Adjusted Coefficient of Determination Criterion, Residual Mean Square Criterion, and Mallows's C_p Criterion (Mallows, 1973).

2.1.2. Shrinkage Methods. Since test based variable selection need to run over all possible variable subsets, it is computationally too expensive for many modern statistical applications (Fan, 2009). An alternative variable selection class is shrinkage method. The idea of shrinkage methods is to apply a specified penalty rule on all predictors to the regression model. The shrinkage method can shrink the unnecessary predictors to zero. One advantage of shrinkage methods is that they are computationally feasible.

A general form of *penalized least square* (PLS) problem as following:

$$min_{\beta}(||\mathbf{Y} - \mathcal{X}\boldsymbol{\beta}||_2^2 + \sum_{j=1}^{p} p_{\lambda}(|\beta_j|)) \tag{2.1}$$

where $\mathbf{Y} = (Y_1, \ldots, Y_n)^T$ is centered response vector and the centered design matrix is $\mathcal{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_n)^T$, p_λ here is a penalty function with the tuning parameters λ. Commonly used parameters selection methods include cross-validation, generalized cross-validation, AIC, BIC and RIC (Shi and Tsai, 2002). Details about the selection of tuning parameters will be discussed in the next section. Meanwhile, many methods are proposed to develop different forms of penalty function p_λ.

Lasso (Tibshirani, 1996) is perhaps the most classical shrinkage method. The idea of Lasso (2.2) is by imposing the L_1 penalty $||\boldsymbol{\beta}||_1$ to the ordinary least square in order to balance the fit of the model and the number of predictors. The first term in (2.2) represents the loss function minimized in the ordinary least squares, the second term is the lasso penalty function while the multiplier $\lambda > 0$ is the penalty constant. Large value of λ will set some components β_j exactly to 0. Lasso is very popular since it is computationally feasible and also it is capable of producing a sparse model. However, Lasso doesn't perform well if predictors are highly correlated. In that case, Lasso tends to randomly select only one variable from each correlated group. Lasso also fails when the number of significant predictors is greater than the sample size, since lasso selects at most n variables before it saturates (Zou and Hastie, 2005).

$$min_\beta\{||\mathbf{Y} - \mathcal{X}\boldsymbol{\beta}||_2^2 + \lambda||\boldsymbol{\beta}||_1\} \tag{2.2}$$

Prior to Lasso, Ridge Regression (Tikhonov, 1943) (2.3) applys the L_2 penalty to the regression problem. Ridge regression can perform better than Lasso when the number of significant predictors is greater than the sample size or models with correlated predictors. However, the ridge regression method doesn't work well for sparse models due to the issue of overselecting predictors.

$$||\mathbf{Y} - \mathcal{X}\boldsymbol{\beta}||_2^2 + \lambda||\boldsymbol{\beta}||_2^2 \tag{2.3}$$

In order to remedy the disadvantages of lasso and ridge regression as mentioned above, Elastic Net (Zou and Hastie, 2005) (2.4) is proposed by combining L_1 and L_2 penalty to

OLS. Elastic Net method outperforms Lasso under correlated predictors models.

$$||\mathbf{Y} - \mathcal{X}\boldsymbol{\beta}||_2^2 + \lambda_1||\boldsymbol{\beta}||_1 + \lambda_2||\boldsymbol{\beta}||_2^2 \tag{2.4}$$

There are other forms of penalty functions such as L_q penalty ($q \geq 1$), the smoothly clipped absolute deviation (SCAD) (Fan and Li, 2001) and minimax concave penalty (MCP) (Zhang, 2010). Fan and Li (2001) advocates penalty functions that give estimators with the property of sparsity, unbiasedness and continuity.

2.2. MODEL FREE VARIABLE SELECTION

Most of the existing variable selection methods are model based. We assume the underlying true model is known up to a finite dimensional parameter or the imposed working model is usefully similar to the true model (Li, 2008). However, the true model might be in a complex form and it is usually unknown. This means if the underlying modeling assumption is badly violated, then none of these variable selection methods would work well. In fact, people usually use the terms "variable selection" and "model selection" interchangeably (Li, Cook and Nachtsheim, 2005). The goal for model selection is usually for further prediction. However, if our goal is identifying the explanatory variables that have detectable effects instead of future prediction, then variable selection does not always have to be part of model selection. For a variable selection method that does not require any underlying true model, it is called "model free variable selection".

It has been shown that the general framework of sufficient dimension reduction is useful for variable selection (Bondell and Li, 2009) since no pre-specified underlying models between **Y** and **X** are required. Model free variable selection can be achieved through the framework of SDR (Li, 1991, 2000; Cook, 1998). This idea can be simply illustrated in the following way. In variable selection, suppose we decompose the predictors as

$$\mathbf{X}^T = (\mathbf{X}_1^T, \mathbf{X}_2^T) \tag{2.5}$$

where $\mathbf{X}_1$ corresponding to p_1 insignificant elements of $\mathbf{X}$, $\mathbf{X}_2$ corresponding to the remaining $p_2 = p - p_1$ significant variables. Equivalently, we want to find $\mathbf{X}_2$ such that

$$Y \perp\!\!\!\perp \mathbf{X}_1 \mid \mathbf{X}_2 \tag{2.6}$$

which implies that, given $\mathbf{X}_2$, $\mathbf{X}_1$ contains no further information about Y. In sufficient dimension reduction as mentioned in Section one, let span $(\boldsymbol{\eta}) = \mathcal{S}_{Y|\mathbf{X}}$, $d = \dim(\mathcal{S}_{Y|\mathbf{X}})$, and partition

$$\boldsymbol{\eta}^T = (\mathbf{B}_1^T, \mathbf{B}_2^T). \tag{2.7}$$

According to the above partition of $\mathbf{X}$ (2.5), then it leads (2.7) to $\mathbf{B}_1 = 0$. Hence, the motivation for imposing the framework of sufficient dimension reduction into variable selection approaches is clear. Similar to the classification of other variable selection methods, there are essentially two types of model free variable selection methods. That is, model free test-based methods and model free shrinkage methods.

2.2.1. Model Free Test-Based Methods. Variable selection is an important step in model-based regression. However, testing the significance of subsets of predictors was not available in sufficient dimension reduction until Cook (2004). Cook (2004) showed that the form of (2.6) is equivalent to

$$P_H \mathcal{S}_{Y|\mathbf{X}} = O_p \tag{2.8}$$

Where $H = \text{span}((I_{p_1}, 0)^T)$ is the subspace of $\mathbb{R}^p$ corresponding to the co-ordinates $\mathbf{X}_1$ and O_p indicates the origin in $\mathbb{R}^p$. Cook (2004) used SIR (Li, 1991) to develop *marginal coordinate hypothesis testing* (MCH) as the first model free test-based variable selection method. MCH used (2.8) as null hypothesis to construct test statistic

$$T_n(H) = ntrace(P_{\hat{H}} \hat{M} P_{\hat{H}}) \tag{2.9}$$

$$\hat{H} = \text{span}(\hat{\Sigma}^{-1/2} \alpha_x),$$

where α_x is a p by r used-selected basis for H with rank r. The $\hat{M}$ here is the same as the estimate kernel matrix in SIR (Li, 1991). Chen and Li (1998) proposed the approximate sliced inverse regression-based t test.

Test-based methods typically incorporate the test into a variable subset search procedure (Bondell and Li, 2009). In the spirit of most test-based methods, Li, Cook and Nachtsheim (2005) proposed another model free variable selection method by applying the backward elimination procedure in a standard normal theory to the Gridded χ^2 -test. First, let $\mathbf{X} = (x_1, \mathbf{X}_2^T)^T$, $\eta = (\eta_1, \eta_2^T)^T$ and define the population residuals

$$r_{1|2} = x_1 - E(x_1|\mathbf{X}_2)$$

$$r_{y|2} = y - E(y|\mathbf{X}_2).$$

Li, Cook and Nachtsheim (2005) showed that under the residual coverage condition,

$$r_{y|2} \perp\!\!\!\perp r_{1|2} \Longrightarrow y \perp\!\!\!\perp x_1|\mathbf{X}_2.$$

The application of this proposition and backward elimination leads to a model free variable screening procedure. There are two components for the implementation of variable screening process: the determination of a smoothing method and an independence test. Li, Cook and Nachtsheim (2005) used the global first-order ordinary least square (OLS) fit as a smoother. Then define the population OLS residual

$$e_{y|2} = y - E(y) - \beta_{y|2}^T\{\mathbf{X}_2 - E(\mathbf{X}_2)\}$$

$$e_{1|2} = x_1 - E(x_1) - \beta_{1|2}^T\{\mathbf{X}_2 - E(\mathbf{X}_2)\}$$

where $\beta_{y|2}$ and $\beta_{1|2}$ are the population OLS vectors from the regression of y on $\mathbf{X}_2$ and x_1 on $\mathbf{X}_2$, respectively. Similar to the previous proposition, under the residual coverage condition,

$$e_{y|2} \perp\!\!\!\perp e_{1|2} \Longrightarrow y \perp\!\!\!\perp x_1|\mathbf{X}_2.$$

Li, Cook and Nachtsheim (2005) used the Gridded χ^2 -test as the independence test to $\hat{e}_{y|2}$ and $\hat{e}_{1|2}$.

2.2.2. Model Free Shrinkage Methods. However, model free test-based methods are computationally intensive especially when the dimension p is large. To remedy this deficiency, model free shrinkage methods are proposed by reformulating SDR as a penalized regression problem (Ni, Cook and Tsai, 2005; Li and Nachtsheim, 2006; Li and Yin, 2008). Li (2007) proposed a unified approach combining SDR and shrinkage estimation to produce sparse estimators of the central subspace. Wang et al. (2012) proposed a distribution-weighted lasso method for the single-index model. The motivation for combining SDR with penalized regression is: central subspace is formed by linear combinations of **ALL** the original predictors and hence it is hard to interpret the result. For example (Li and Nachtsheim, 2006), consider true model

$$Y = exp(-0.759\boldsymbol{\eta}^T\mathbf{X} + 1) + 0.5\epsilon \tag{2.10}$$

where $\mathbf{X} = (X_1, X_2, \ldots, X_6)^T$, $\boldsymbol{\eta} = (1, -1, 0, 0, 0, 0)^T$ and $\epsilon \sim N(0, 1)$. The solution of the central subspace given by SIR (Li, 1991) is

$$\hat{\boldsymbol{\eta}} = (0.765, -0.638, -0.079, 0.029, -0.003, -0.001)^T \tag{2.11}$$

We can see that the coefficients for the last four predictors given by SIR are significantly smaller than the first two predictors, but it still fails to shrink the insignificant predictors to zero. This is a critical issue especially when the model is sparse. Notice that Lasso or Elastic Net are able to shrink useless variables to zero. Ni, Cook and Tsai (2005) applied Lasso penalty under the framework of sufficient dimension reduction to obtain a shrinkage sliced inverse regression estimator. Li and Nachtsheim (2006) combined the least angle regression (Efron et al., 2004) (LARS) algorithm and SIR to produce sparse SIR algorithm. However, these methods all rely on special sufficient dimension reduction methods (e.g. SIR).

Li (2007) developed a unified approach within the context of sufficient dimension reduction to produce sparse estimates of the central subspace. The idea of Li (2007) can be described as follows. For all methods taking the spectral decomposition approaches, let M be the corresponding kernel matrix, $\nu_i^T G \nu_j = 1$ if $i = j$ and 0 otherwise, where G is a positive definite matrix. We can always convert condition

$$M\nu_i = \lambda_i \Sigma_{XX} \nu_i (i = 1, \ldots, d)$$

to a regression-type optimization problem and then employ a penalty to the regression, such as Lasso.

$$\hat{\beta} = argmin_{\beta} \sum_{i=1}^{p} ||G^{-1} m_i - \beta\beta^T m_i||_G^2 \tag{2.12}$$

subject to $\beta^T G \beta = I_d$, and $||\beta||_1 \leq \lambda$. Here, m_i is the column of the kernel matrix $M^{1/2}$, and $i = 1, \ldots, p$.

However, all the above model free shrinkage methods fail when the sample size is smaller than the total number of predictors due to the limitations of those SDR methods they adopted. Li and Yin (2008) developed ridge SIR estimator using L_2 regularization to SIR to solve this problem. The idea of ridge SIR estimator is the following. For a nonnegative constant τ, let

$$G_\tau(A, C) = \sum_{y=1}^{h} \hat{f}_y ||(\bar{\mathbf{X}}_y - \bar{\mathbf{X}}) - \hat{\Sigma}_{\mathbf{XX}} A C_y||^2 + \tau vec(A)^T vec(A) \tag{2.13}$$

where $vec(\cdot)$ is a matrix operator that stacks all columns of the matrix to a single vector. The first term in (2.13) is the least-square form of SIR (Cook, 2004), and the second term in (2.13) is a form similar to the L_2 regularization. Let $(\hat{A}, \hat{C})$ be the solution to the minimum $G_\tau(A, C)$. Then span$(\hat{A})$ is a *ridge SIR estimator* of the central subspace $\mathcal{S}_{Y|\mathbf{X}}$.

However, the ridge SIR estimator is not capable of variable selection because it generates linear combinations of all the predictors. Hence, Li and Yin (2008) also proposed the sparse ridge SIR estimator by utilizing L_1 regularization to the least square formulation of SIR. The *sparse ridge SIR estimator* can be obtained by minimizing

$$G_\lambda(\alpha) = \sum_{y=1}^{h} \hat{f}_y ||(\bar{\mathbf{X}}_y - \bar{\mathbf{X}}) - \hat{\Sigma}_{\mathbf{XX}} diag(\alpha) \hat{A} \hat{C}_y||^2 \tag{2.14}$$

over α, subject to $\sum_{j=1}^{p} |\alpha_j| \leq \lambda$, for some nonnegative constant λ. Here the sparse ridge SIR estimator of the central subspace $\mathcal{S}_{Y|\mathbf{X}}$ is defined as

$$\text{span}(diag(\hat{\alpha}\hat{A})),$$

where $\hat{\alpha} = (\hat{\alpha}_1, \ldots, \hat{\alpha}_p) \in \mathbb{R}^p$.

Other than the methods described above, there are some extensions for model free shrinkage methods. For example, Bondell and Li (2009) proposed a general shrinkage estimation strategy for the entire inverse regression estimation (IRE) family. The *shrinkage inverse regression estimator* of the central subspace $\mathcal{S}_{Y|\mathbf{X}}$ is defined as

$$\text{span}\{diag(\hat{\alpha})\hat{\eta}\},$$

which can be obtained by minimizing

$$G_s(\alpha) = n\{vec(\hat{\theta}) - vec(diag(\alpha)\hat{\eta}\hat{\gamma})\}^T V_n \{vec(\hat{\theta}) - vec(diag(\alpha)\hat{\eta}\hat{\gamma})\} \tag{2.15}$$

subject to $\sum_{j=1}^{p} |\alpha_j| \leq \tau$, $\tau \geq 0$. In addition, Wang et al. (2012) developed a new shrinkage method for the single-index model. Details for that paper will be provided in the next section.

3. SPARSE GROUP SUFFICIENT DIMENSION REDUCTION

3.1. LITERATURE REVIEW

In the previous section, we have introduced the idea of model free variable selection and described several model free variable selection approaches. However, none of those model-free variable selection methods take the prior group (predictor network) information into account. Intuitively, methods which ignore the group or cluster information are not capable of providing a complete solution and normally result in a loss of power. In addition, the analysis results may be difficult to interpret. Prior group information is common in gene pathway analysis where genes naturally fall into groups. More discussions about gene pathway analysis will be provided in the following section.

In this section, we adopt the idea of sparse group lasso (Friedman et al., 2010) to the framework of the sufficient dimension reduction for regression problems with grouped covariates. We propose a method called the *sparse group sufficient dimension reduction* (sgSDR) to conduct group and within group variable selection simultaneously without assuming a specific model structure on the regression function. Simulation studies show that our method is comparable to the sparse group lasso under the regular linear model setting, and outperforms sparse group lasso with higher true positive rates and substantially lower false positive rates when the regression function is nonlinear or (and) the error distributions are non-Gaussian. An analysis of a glioblastoma microarray data is included for illustration of our method.

Let's first review some variable selection methods which consider the prior group information. The idea of test-based group variable selection methods is usually minimizing a specified loss function by imposing a penalty function on the group parameters. This raises the question of how to penalize a group of parameters. The group lasso proposed by Yuan and Lin (2006) overcomes that problem by minimizing the following penalized least squares regression:

$$\frac{1}{2}||\mathbf{y} - \sum_{g=1}^{G} \mathcal{X}^{(g)}\boldsymbol{\beta}^{(g)}||_2^2 + \lambda \sum_{g=1}^{G} \sqrt{p_g}||\boldsymbol{\beta}^{(g)}||_2, \tag{3.1}$$

where $\mathbf{y} = (y_1, \ldots, y_n)$ is the observed centered response vector, $\mathcal{X}^{(g)}$ is the submatrix of the centered design matrix $\mathcal{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^T$ with columns corresponding to the predictors in the gth group, $\boldsymbol{\beta}^{(g)}$ the coefficient vector of that group with p_g as its length. Also, here

$$\sum_{g=1}^{G} \sqrt{p_g}||\boldsymbol{\beta}^{(g)}||_2 = \sqrt{p_1}\sqrt{\beta_{11}^2 + \beta_{12}^2 + \ldots + \beta_{1p_1}^2} + \ldots + \sqrt{p_G}\sqrt{\beta_{G1}^2 + \beta_{G2}^2 + \ldots + \beta_{Gp_G}^2},$$

and β_{gk} denotes the k^{th} element in the g^{th} group with $k = 1, \ldots, p_g$ and $g = 1, \ldots, G$. Particularly, if $p_1 = p_2 = \ldots = p_G = 1$, then

$$\sum_{g=1}^{G} \sqrt{p_g}||\boldsymbol{\beta}^{(g)}||_2 = |\beta_1| + |\beta_2| + \ldots + |\beta_p|.$$

In other words, this objective function is reduced to regular lasso if the group structure is ignored. Notice that the rescaling factor p_g makes the penalty level proportional to the group size, which ensures that small groups are not overwhelmed by large groups in group selections. The group lasso penalty has been investigated in multiple studies (Bakin, 1999; Meier et al., 2008; Huang et al., 2009). The sparsity of the solution is determined by the tuning parameter λ. A smaller tuning parameter value will result in more shrinkage on groups selection. Detailed discussions about the tuning parameters selection will be given in next subsection.

However, group lasso assumes that $\mathcal{X}^{(g)}$ is orthonormal, that is, the data is orthonormal within each group. Simon and Tibshirani (2011) pointed out this orthonormalization changes the problem. They also proposed an improved version of group lasso, called the standardized group lasso. Standardized group lasso considers (3.2) which penalizes the fit of each group $\mathcal{X}^{(g)}\boldsymbol{\beta}^{(g)}$ rather than the individual coefficients $\boldsymbol{\beta}^{(g)}$,

$$\frac{1}{2}||\mathbf{y}-\sum_{g=1}^{G}\mathcal{X}^{(g)}\boldsymbol{\beta}^{(g)}||_2^2+\lambda\sum_{g=1}^{G}\sqrt{p_g}||\mathcal{X}^{(g)}\boldsymbol{\beta}^{(g)}||_2. \tag{3.2}$$

Both group lasso and standardized group lasso are useful methods for identifying important groups. However, it is not capable of selecting important predictors within each group, which will be an issue when p_g is large.

Friedman et al. (2010) proposed the sparse group lasso (SGL) which may achieve sparsity of both groups and within each group by minimizing the following penalized least squares regression:

$$\frac{1}{2}||\mathbf{y}-\sum_{g=1}^{G}\mathcal{X}^{(g)}\boldsymbol{\beta}^{(g)}||_2^2+\lambda_1\sum_{g=1}^{G}\sqrt{p_g}||\boldsymbol{\beta}^{(g)}||_2+\lambda_2||\boldsymbol{\beta}||_1. \tag{3.3}$$

Sparse group lasso is capable of selecting important groups and important predictors within the selected groups simultaneously. Objective function (3.3) achieves this goal by imposing L_2 penalty to group parameters for group selections and the L_1 penalty to all β for within group variable selection. Sparse group lasso can be viewed as an extension to group lasso (Yuan and Lin, 2006). Hence, it is clear that sparse group lasso is reduced to the group lasso when $\lambda_2 = 0$, and regular lasso when $\lambda_1 = 0$. Furthermore, sparse group lasso might lead to better predictions since it takes the cluster structure into consideration; and also, its within-group variable selection aspect can lead to more parsimonious models and hence interpretable results. However, all the above lasso-based methods assume a **linear** relationship between the response and the predictors, and may **not** be robust to non-Gaussian errors. In other words, they might fail if the modeling assumptions are violated. But in real life, most modeling problems are very complicated and not necessarily described by a linear model with Gaussian errors. Therefore, we propose a sparse group sufficient dimension reduction method to overcome these limitations.

Li et al. (2010) proposed the groupwise dimension reduction within the context of sufficient dimension reduction which incorporates the prior grouping information into the estimation of the central mean subspace. Li et al. (2010) first defined $\tau_1 \oplus \cdots \oplus \tau_g$ as a groupwise mean dimension reduction subspace with respect to $\{S_1, \ldots, S_g\}$ such that

$$E(\mathbf{Y}|\mathbf{X}) = E(\mathbf{Y}|P_{\tau_1}\mathbf{X}, \ldots, P_{\tau_1}\mathbf{X}), \tag{3.4}$$

where the subspace $\tau_l \subseteq S_l$ for $l = 1, \ldots, g$ and $\{S_1, \ldots, S_g\}$ are the subspaces of $\mathbb{R}^p$ that form an orthogonal decomposition of $\mathbb{R}^p$. Thus $\{S_1, \ldots, S_g\}$ satisfy

$$S_1 \oplus \cdots \oplus S_g = \mathbb{R}^p.$$

Then the intersection of all groupwise mean dimension reduction subspace is defined as groupwise central mean dimension reduction subspace, denoted by

$$S_{E(\mathbf{Y}|\mathbf{X})}(S_1, \ldots, S_g) = \tau_1^* \oplus \cdots \oplus \tau_g^* \tag{3.5}$$

for some subspaces $\tau_1^*, \ldots, \tau_g^*$. Groupwise dimension reduction (Li et al., 2010) is an extension to MAVE (Xia et al., 2002). Recall (1.9), (1.10) and (1.11) in MAVE (Xia et al., 2002), as a grouping version of MAVE, groupwise dimension reduction showed that groupwise central mean subspace can be recovered by estimating $D_{v_l}E(\mathbf{Y}|V)$ such that

$$\mathrm{span}(\oplus_{l=1}^{g} E\{D_{v_l}E(\mathbf{Y}|V)D_{v_l}^T E(\mathbf{Y}|V)\}) = S_{E(\mathbf{Y}|\mathbf{X})}(S_1, \ldots, S_g) \tag{3.6}$$

where $v_l = (v_{l_1}, \ldots, v_{lp_l})^T$ represents the evaluation of the random vector of $V_l = \gamma_l^T\mathbf{X}$ with $\gamma_l \in \mathbb{R}^{p\times p_l}$ as a basis matrix of S_l and D_{v_l} denotes the differential operator

$$(\partial/\partial_{v_{l_1}}, \ldots, \partial/\partial_{v_{lp_l}})^T.$$

It also showed that the left hand side of (3.6) is equivalent to

$$\mathrm{span}(E\{(\oplus_{l=1}^{g} D_{v_l})E(\mathbf{Y}|V)(\oplus_{l=1}^{g} D_{v_l})^T E(\mathbf{Y}|V)\}) \tag{3.7}$$

where D_{v_l} is treated as ordinary vectors of numerals. Equation (3.7), makes clear the lose connection between groupwise dimension reduction and MAVE. Simulation studies and real data analyses show that the groupwise dimension reduction approach can substantially increase the estimation accuracy and enhance the estimates interpretability. However, their method is limited to the dimension reduction of the conditional mean function ($\mathbb{E}(Y|\mathbf{X})$), and is not capable of variable selections. The *sparse group sufficient dimension reduction* (sgSDR) method we propose in this article can conduct variable selection in the general dimension reduction context (not limited to the conditional mean function) while incorporating the group knowledge, and can also be applied to the $n << p$ setting.

3.2. SPARSE GROUP SUFFICIENT DIMENSION REDUCTION

3.2.1. Methodology. In this paper, we propose a method called the sparse group sufficient dimension reduction (sgSDR), which conducts both group and within-group variable selections simultaneously under the framework of sufficient dimension reduction. We focus on the following general single-index model:

$$Y = g(\boldsymbol{\beta}^T\mathbf{X}, \epsilon) \tag{3.8}$$

where $\epsilon \perp\!\!\!\perp \mathbf{X}$. Without loss of generality, we assume that $\mathbf{X}$ is centered with $\mathbb{E}(\mathbf{X}) = 0$, and also suppose that $\mathbf{X}$ can be slitted into G groups,

$$\mathbf{X}^T = (\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \cdots, \mathbf{X}^{(G)}),$$

where $\mathbf{X}^{(g)}$ is a p_g-dimensional row vector, for $g = 1, \ldots, G$, and $\sum\limits_{g=1}^{G} p_g = p$. Wang et al. (2012) proposed the distribution-transformation least squares estimator in a single-index model under large p small n setting. Following Wang et al. (2012), we consider the following minimization problem:

$$\frac{1}{2}||\mathcal{F}_n(\mathbf{y}) - \sum_{g=1}^{G} \mathcal{X}^{(g)}\boldsymbol{\beta}^{(g)}||_2^2 + \lambda_1 \sum_{g=1}^{G} \sqrt{p_g}||\boldsymbol{\beta}^{(g)}||_2 + \lambda_2||\boldsymbol{\beta}||_1, \tag{3.9}$$

where

$$\mathcal{F}_n(\mathbf{y}) = (F_n(y_1), \ldots, F_n(y_n))^T$$

and $\mathcal{X}$ are all centered, and $F_n(.)$ is the empirical distribution function. We call the solution ($\boldsymbol{\beta}^{(g)}$) of (3.9) the sparse group sufficient dimension reduction estimator (sgSDR). Equation (3.9) is based on the following observation.

Proposition 1. *Under the linearity condition, and assume that $\mathbf{\Sigma}_s$, the marginal covariance matrix of all the significant predictors (denoted by $\mathbf{X}_s$ here for easy of exposition) is invertible, then*

$$\mathbf{\Sigma}_s^{-1}\,\mathbb{C}\text{ov}\{\mathbf{X}_s, F(Y)\} = c\boldsymbol{\beta}_s, \tag{3.10}$$

where $\boldsymbol{\beta}_s$ consists all non-zero coefficients of $\boldsymbol{\beta}$ from (3.8)*, $c \in \mathbb{R}^1$ is a constant, $F(Y)$ is the cumulative distribution function of Y.*

Proof: The proof borrows heavily from Theorem 2.1 of Li and Duan (1989), hence we include it here for reference.

Theorem 3.1. *(Li and Duan, 1989) Under model* (1.1) *and linearity condition* (1.8)*, assume $\Omega = (a, b)$ is a nonempty convex set in $\mathbb{R}^{p+1}$, the criterion function $L(\theta, y)$ is convex in θ with probability 1 and the risk function $R(a, b)$ as the expectation of the loss function $L(\theta, y)$, that is, $R(a, b) = E\{L(a+b^T\mathbf{x}, y)\}$. The minimization problem: minimize $R(a, b)$ over $(a, b) \in \Omega$ has a solution β^* such that β^* is proportional to β, that is,*

$$\beta^* = \gamma\beta,$$

for some scalar γ.

The proof of the above theorem is similar to (Li and Duan, 1989) and it can be shown as the following: Since

$$\begin{aligned} R(a, b) &= E\{L(a + b^T\mathbf{X}, y)\} \\ &= E[E\{L(a + b^T\mathbf{x}, \mathbf{y})|(\beta^T\mathbf{x}, \epsilon)\}] \\ &= E[E\{L(a + b^T\mathbf{x}, g(\alpha + \beta^T\mathbf{x}, \epsilon))|(\beta^T\mathbf{x}, \epsilon)\}] \end{aligned}$$

where $g(\cdot)$ is any link function. Applying Jensen's inequality, since $L(\theta, y)$ is convex in θ with probability 1 and then we have

$$\begin{aligned} R(a,b) &\geq E\{L[E\{(a+b^T\mathbf{x}, g(\alpha+\beta^T\mathbf{x},\epsilon))|(\beta^T\mathbf{x},\epsilon)\}]\} \\ &= E\{L(a+E(b^T\mathbf{x}|\beta^T\mathbf{x},\epsilon),\mathbf{y})\} \end{aligned}$$

Due to the linearity condition, the conditional expectation $E(b^T\mathbf{x}|\beta^T\mathbf{x})$ is linear in $\beta^T\mathbf{x}$, i.e.

$$E(b^T\mathbf{x}|\beta^T\mathbf{x}) = c + d\beta^T\mathbf{x}.$$

Therefore,

$$E\{L(a+b^T\mathbf{X},y)\} \geq E\{L(a+(c+d\beta^T\mathbf{x}),\mathbf{y})\}$$

for some constants c and d. Hence, it is obvious to see the result $\beta^* = \gamma\beta$.

Proof of Proposition 1: Let $S_{F(\mathbf{Y})|\mathbf{X}}$ denote the central subspace for the regression of $F(\mathbf{Y})$ versus $\mathbf{X}$, let η be the orthonormal basis of $S_{F(\mathbf{Y})|\mathbf{X}}$ and let β be an orthonormal basis of $\mathcal{S}_{Y|\mathbf{X}}$. Since

$$Y \perp\!\!\!\perp \mathbf{X} \mid P_{\mathcal{S}\{\boldsymbol{\eta}\}}\mathbf{X}$$

implies

$$F(Y) \perp\!\!\!\perp \mathbf{X} \mid P_{\mathcal{S}\{\boldsymbol{\eta}\}}\mathbf{X}.$$

We have

$$S_{F(Y)|\mathbf{X}} \subseteq \mathcal{S}_{Y|\mathbf{X}}.$$

In order to show that (3.10) holds, we only need to show that

$$\boldsymbol{\Sigma}_s^{-1}\,\mathbb{C}\text{ov}\{\mathbf{X}_s, F(Y)\} \in S_{F(Y)|\mathbf{X}}.$$

Since the loss function for ordinary least square is convex, this is an obvious result from the previous theorem (Li and Duan, 1989).

Proposition 2. *The above proposition is also true for any monotonic transformation of response* $\mathbf{Y}$*.*

Proof: The proof of this proposition follows because

$$\mathbf{\Sigma}_s^{-1}\mathbb{C}\mathrm{ov}\{\mathbf{X}_s, h(Y)\} \in S_{h(Y)|\mathbf{X}},$$

where $h(\cdot)$ is any transformation of y. This proposition implies that the empirical cumulative distribution function used in our method can be replaced by any other transformation of $\mathbf{Y}$. The reason we choose $F(\mathbf{Y})$ is due to its simplicity. More discussion about the monotonic transformation of $\mathbf{Y}$ will be given later.

3.2.2. Selection of Tuning Parameters. The sparsity of the solution is determined by the tuning parameters λ. Specifically, in (3.9), λ_1 controls the sparsity of group selection while the number of variables selected within each group depends on the value of λ_2. The larger value of λ_1 implies more shrinkage on group parameters which will result in fewer groups being selected. Similarly, smaller value of λ_2 implies less shrinkage on individual parameters which will result in more variables within groups being selected. And vice versa. Intuitively, more shrinkage might lead to loss of important predictors while less shrinkage will end up with interpretation difficulty and inaccurate prediction. This raises the issue of how to balance the selection of two tuning parameters λ_1 and λ_2. In this paper, to select the two tuning parameters, λ_1 and λ_2, we employ the commonly used five-fold cross validation as well as a modified BIC-type criterion.

3.2.2.1. Cross validation. Cross validation is a classical model validation technique for evaluating how the results of a statistical analysis will generalize to future data set. The traditional implementation of cross validation is to split the data set into two complimentary subsets, one subset is treated as training set for performing the analysis on while the other subset is treated as validation set or testing set for validating purpose. Cross validation includes K-fold cross-validation, two fold cross-validation, repeated random sub-sampling validation, and Leave-one-out cross-validation (please refer to: An Introduction to the Bootstrap; Efron and Tibshirani, 1993).

For K-fold cross-validation, the value of K is typically chosen as 5 or 10. In sparse group sufficient dimension reduction, we notice that the five-fold cross validation and ten-fold cross validation do not make a much difference. So in this paper, five-fold cross validation is applied for less computational time. Simon et al. (2012) pointed out that it is time consuming to work on two tuning parameters at the same time. Hence, Simon et al. (2012) suggested that when using $\lambda_1 = 19\lambda_2$, the simulation performance is the best. But there are no theoretical justification of this special λ_1 to λ_2 ratio, and also the λ_1 to λ_2 ratio needs to be adjusted when the scenarios vary. So we decide to run over all possible combinations of λ_1 and λ_2 on a grid instead of fixing the λ_1 to λ_2 ratio.

The algorithm can be described in the following way:

1. We first provide a wide range of values for both λ_1 and λ_2

2. Next randomly choose four fifths of the data set for training and the rest of the data will be used for testing purpose.

3. Then apply all combinations of λ_1 and λ_2 on the λ_1-λ_2 grid from step one to the training subset.

4. Obtain the value of $\hat{\beta}$ from sparse group sufficient dimension reduction method for one combination of λ_1 and λ_2.

5. Compute the estimate response $\hat{Y}_{testing}$ from the testing data set of each of the parameter combination selected from previous step.

6. The parameter combination with the minimum $|\hat{Y}_{testing} - Y_{testing}|$ will be finally selected, where $Y_{testing}$ is the true response from the testing subset.

This might cost a little bit longer computing time than applying the 19 times ratio directly, but it generally leads to a more precise result.

Cross validation is widely used due to its simple implemention. However, it only yields meaningful results if the validation set and training set are drawn from the same population. Particularly, the lasso-typed methods do not appear to be consistent in variable selection if cross validation is applied for tuning parameter selection (Wang and

Leng, 2009; Chand, 2012). Hence, we also consider a modified BIC approach for tuning parameter selection.

3.2.2.2. Modified bayesian information criterion. As one of the most popular model selection tools, BIC (Schwarz, 1978) has been widely used as a tuning parameter selection method. Schwarz defined the BIC as:

$$BIC_{\lambda}(\gamma) = \log{(\hat{\sigma}^2)} + \log(n) \times \frac{p}{n} \tag{3.11}$$

where $\hat{\sigma}^2$ is the residual variance and p is the total number of parameters. The candidate model with the minimum BIC value will be finally selected. In the model selection process, adding more parameters to a new model will always increase the likelihood (the first term in (3.11)). But for the law of parsimony in statistical modeling, a complex model will lead to greater variance of the estimates. (3.11) achieves the goal of balancing the model simplicity and the goodness of fit. Several published works have shown that the traditional BIC can identify the true model consistently when the predictor dimension is fixed (Nishii, 1984; Yang, 2005).

However, there are no theoretical results showing that the traditional BIC is also consistent with a diverging number of parameters or whether this selection is true for penalized-type methods, such as Lasso type problems (Wang and Leng, 2009). In order to remedy these two deficiencies, Wang and Leng (2009) suggested a modified Bayesian information criterion (BIC) criterion to choose the value of the tuning parameters.

$$BIC_{\lambda}(\gamma) = \log{(\hat{\sigma}^2_S)} + |S| \times \log(n) \times \frac{c_n}{n} \tag{3.12}$$

where $|S|$ is the number of significant selected parameters in the model, $c_n > 0$ and (3.12) is reduced to the traditional BIC (3.11) when $c_n = 1$. In (3.12), $\hat{\sigma}^2_S$ is the ratio of sums of squares of error for the significant parameters of the model (SSE_S) to the sample size, that is,

$$\hat{\sigma}^2_S = inf_{\beta s}(||\mathbf{Y} - \mathbf{X}_S\beta_S||^2/n).$$

Wang and Leng (2009) also prove that this modified BIC will identify the model consistently under both a diverging number of parameters model and penalized estimators.

However, traditional BIC and AIC are designed for $p < n$ case and might not perform well when the number of predictors is greater than the sample size. As proposed by An et al. (2009), an extra constant c is added in the first term in order to avoid the excessively over fitted models particularly when p is much greater than n.

$$BIC_{\lambda}(\gamma) = \log\left(\hat{\sigma}_S^2 + C_0\right) + |S| \times \log(n) \times \frac{c_n}{n} \tag{3.13}$$

where C_0 is a positive constant. There is no theoretical properties about the specific values of C_0, and hence the choice of C_0 relies on one's empirical experience.

In sparse group sufficient dimension reduction, we apply a modified BIC method following Wang et al. (2009) and An et al. (2009). The BIC criterion is defined as:

$$BIC_{\lambda_1,\lambda_2}(\gamma) = n \times \log\left(\frac{RSS_{\lambda_1,\lambda_2}}{n} + c\right) + \log(n) \times \hat{p}_s \times \frac{c_n}{n}, \tag{3.14}$$

where $\hat{p}_s$ denotes the total number of selected significant predictors (those with nonzero estimated coefficients) using λ_1 and λ_2.

$$RSS_{\lambda_1,\lambda_2} = \|Response - \mathbf{X}\hat{\boldsymbol{\beta}}\|_2^2$$

with

$$Response = Y$$

for SGL and

$$Response = \mathcal{F}_n(\mathbf{y})$$

for sgSDR. Here

$$c = c_0 \times Var(Response)$$

where c_0 is simply set as 0.001 throughout all our simulations. In theory, it is required in Wang et al. (2009) that $c_n \to \infty$. Under sgSDR, it is quite desirable when

$$c_n = n^{0.98}$$

is used for the simulation results. Simulation studies suggest that this modified BIC method outperforms the traditional five-fold cross validation method for the group selection.

3.2.3. Simulation. In this section, we compare the performance of our method with the sparse group lasso. We adopt the SLEP package (Liu et al., 2009) to implement our method. We considered linear models, nonlinear models and generalized linear models with Gaussian and non-Gaussian errors. We use the average true positive rate (TPR = the ratio of the number of correctly declared active variables to the number of truly active variables); and the average false positive rate (FPR = the ratio of the number of falsely declared active variables to the total number of truly inactive variables) as evaluation measurements to summarize variable selection results from 100 simulation runs.

3.2.3.1. Model I. For a fair comparison, we first consider a regular linear model as Simon et al. (2012) discussed in their paper. The predictor $\mathbf{X}$ is generated from $N(0, I_p)$, ϵ is standard normal and independent of $\mathbf{X}$, the univariate response Y is constructed as:

$$Y = \sum_{g=1}^{G} (\boldsymbol{\beta}^{(g)})^T \mathbf{X}^{(g)} + \sigma\epsilon, \tag{3.15}$$

where $G = 10$, σ is set to make the signal to noise ratio as 2. And the coefficients for the first l group are $\boldsymbol{\beta}^{(g)} = (1, 2, 3, 4, 5, 0, \ldots, 0)^T$, for $g = 1, \ldots, l$, with l varying from 1 to 3; and all zeros for the rest of $G - l$ groups. Following Simon et al. (2012), we took $n = 60$, $p = 1500$. Table 3.1 provides the average true positive and false positive rates on variable and group selections, respectively. As shown on Table 3.1, the performances of sgSDR and SGL are comparable in the sense that the average TPRs and FPRs on both group level and variable level are very close to each other.

Table 3.1. sgSDR: Linear Model I With Gaussian Error

		$l=1$		$l=2$		$l=3$	
	Method	TPR	FPR	TPR	FPR	TPR	FPR
Cross Validation	sgSDR (var)	0.754	0.126	0.640	0.321	0.584	0.357
	SGL (var)	0.753	0.100	0.641	0.317	0.562	0.328
	sgSDR (group)	0.990	0.790	0.951	0.868	0.924	0.842
	SGL (group)	0.980	0.800	0.970	0.891	0.906	0.847
Modified BIC	sgSDR (var)	0.680	0.031	0.341	0.021	0.265	0.031
	SGL (var)	0.640	0.034	0.300	0.034	0.268	0.032
	sgSDR (group)	1.00	0.674	0.930	0.782	0.907	0.890
	SGL (group)	1.00	0.989	0.995	0.968	0.942	0.940

3.2.3.2. Model II. We now consider a variation of Model I. We take $p = 2000$, $G = 10$, Y is still generated as in (3.15), however, the predictors now are mildly correlated, ϵ follows $Cauchy(1)$ distribution, and $\boldsymbol{\beta}^{(g)} = (-2, 3, 0, \dots, 0)^T$, for $g = 1, \dots, l$, with l varying from 1 to 3; and zeros otherwise. Specifically, within each group, $\mathbf{X}^{(g)} = (X_1^{(g)}, \dots, X_{200}^{(g)})$ are all generated as independent standard normal random variables except $X_3^{(g)}$, which is generated to be correlated with X_1^g and X_2^g by:

$$X_3^{(g)} = \frac{2}{3}X_1^{(g)} + \frac{2}{3}X_2^{(g)} + \frac{1}{3}e_g, \tag{3.16}$$

where e_g follows standard normal distribution. A different version of Model II was considered by Wang et al. (2012). Table 3.2 shows the simulation results with $n = 60$ from

Table 3.2. sgSDR: Linear Model II With Cauchy Error and Correlated Predictors

		$l = 1$		$l = 2$		$l = 3$	
	Method	TPR	FPR	TPR	FPR	TPR	FPR
Cross Validation	sgSDR (var)	1.00	0.024	0.985	0.035	0.920	0.038
	SGL (var)	0.722	0.416	0.713	0.412	0.741	0.425
	sgSDR (group)	1.00	0.950	1.000	0.920	1.000	0.977
	SGL (group)	0.833	0.825	0.750	0.799	0.853	0.825
Modified BIC	sgSDR (var)	0.910	0.006	0.800	0.009	0.750	0.015
	SGL (var)	0.295	0.023	0.302	0.021	0.243	0.022
	sgSDR (group)	1.000	0.253	0.870	0.395	0.840	0.544
	SGL (group)	0.620	0.501	0.600	0.485	0.603	0.513

100 simulation runs. We can see that our method (sgSDR) is more robust under the Cauchy random error, which tends to yield relatively higher TPR and significantly lower FPR, compared with SGL with respect to variable selections. With $p = 2000$ in our setting, SGL provides a FPR about 40% higher than our method, which means that about 800 more inactivate variables are mistakenly selected as significant variables by SGL. On the group level, sgSDR outperforms SDR with higher TPR and lower FPR with either tuning parameter selection method. The modified BIC method can significantly decrease the group FPR while its TPR it is not as high as those selected by the cross validation method.

3.2.3.3. Model III. In this example, the linear model (3.15) is reconsidered with larger sample size, larger dimension p and more groups, that is, $n = 200$, $p = 5000$ and $G = 50$. The predictors are generated by $N(0, \mathbf{\Sigma})$, where $\mathbf{\Sigma} = \big(.5^{|i-j|}\big)$, $i, j = 1, \ldots, p$. We consider l $(5, 10, 15)$ significant groups, with $\boldsymbol{\beta}^{(g)} = (3, 1.5, 2, \ldots, 0)^T$, $g = 1, \ldots, l$. The results are shown on Table 3.3. Similar conclusions as Model II on both variable level and group level can be drawn here.

3.2.3.4. Model IV. We now compare the performances of sgSDR and SGL for nonlinear models under the standard normal and Cauchy errors. We consider the following model:

$$Y = \exp\big(\sum_{g=1}^{G} \mathbf{X}^{(g)}\boldsymbol{\beta}^{(g)} + 3\epsilon\big) \tag{3.17}$$

The predictors $\mathbf{X}$ and the coefficients $\boldsymbol{\beta}$ are set up exactly the same as those of Model II, and $\epsilon \sim N(0, 1)$ or standard Cauchy, respectively. As shown on Table 3.4, our method outperforms SGL with significantly lower FPR and slightly higher TPR. For models with nonlinear regression function and Cauchy errors, SGL fails completely, the average FPR for SGL is above 99%, which implies that it mistakenly selected over 1900 inactive predictors as significant ones. The tuning parameter method, modified BIC performs very well in these two cases on both variable selection level and group level, with significant higher TPR (almost 1) and lower FPR.

Table 3.3. sgSDR: Linear Model III With Cauchy Error

		$l = 1$		$l = 2$		$l = 3$	
	Method	TPR	FPR	TPR	FPR	TPR	FPR
Cross Validation	sgSDR (var)	0.980	0.031	0.880	0.046	0.770	0.065
	SGL (var)	0.724	0.268	0.620	0.198	0.655	0.284
	sgSDR (group)	1.000	0.684	1.000	0.730	0.985	0.780
	SGL (group)	0.810	0.632	0.870	0.660	0.836	0.749
Modified BIC	sgSDR (var)	0.930	0.0025	0.600	0.005	0.315	0.006
	SGL (var)	0.093	0.013	0.1311	0.0128	0.154	0.011
	sgSDR (group)	1.000	0.182	0.900	0.300	0.570	0.293
	SGL (group)	0.400	0.311	0.350	0.281	0.335	0.273

Table 3.4. sgSDR: Nonlinear Model IV with Gaussian and Cauchy Error

		$l = 1$		$l = 2$		$l = 3$	
	Method	TPR	FPR	TPR	FPR	TPR	FPR
Gaussian Error Cross Validation	sgSDR (var)	0.990	0.032	0.990	0.028	0.977	0.031
	SGL (var)	0.980	0.871	0.973	0.955	0.960	0.985
	sgSDR (group)	1.00	0.894	1.000	0.957	1.000	0.971
	SGL (group)	1.000	0.993	1.000	1.000	1.000	1.000
Gaussian Error Modified BIC	sgSDR (var)	0.900	0.003	0.965	0.005	0.888	0.012
	SGL (var)	0.800	0.437	0.918	0.798	0.983	0.955
	sgSDR (group)	0.980	0.223	0.990	0.431	0.960	0.574
	SGL (group)	1.000	1.000	1.000	1.000	1.000	1.000
Cauchy Error Cross Validation	sgSDR (var)	1.000	0.017	0.980	0.040	0.921	0.026
	SGL (var)	1.000	1.000	1.000	0.996	1.000	1.000
	sgSDR (group)	1.000	0.933	1.000	0.900	1.000	0.952
	SGL (group)	1.000	1.000	1.000	1.000	1.000	1.000
Cauchy Error Modified BIC	sgSDR (var)	0.900	0.001	0.825	0.009	0.767	0.022
	SGL (var)	1.000	1.000	1.000	0.999	1.000	1.000
	sgSDR (group)	1.000	0.111	0.950	0.387	0.900	0.628
	SGL (group)	1.000	1.000	1.000	1.000	1.000	1.000

3.3. GENE PATHWAY DATA ANALYSIS

3.3.1. Basics of Genetics. Genetics has become an indispensable component in modern biology and medicine (Griffiths et al., 2000). Genetics researches have made great contributions to the many aspects of the agriculture, biology, disease research and society. Most of the genetic studies aim to detect the associations between gene expressions and the occurrence or progression of disease phenotypes. In this section, we will first introduce some basic concepts in genetics studies, then have a brief discussion about some existing gene pathway analysis methods, and finally apply our method sgSDR to a gene pathway data set.

Deoxyribonucleic acid (DNA), which was discovered by Watson and Crick (1953), known as one of the most key and basic components in genetics studies. DNA is a biological molecule which contains the hereditary material in human beings and almost all other organisms. The DNA molecule (Watson and Crick, 1953) has a double-stranded helix consisting of two long polymers of simple units called nucleotides, molecules with backbones made of deoxyribose sugar and phosphate groups, along with any of the four nitrogenous base attached to the sugars (see Figure 3.1, image courtesy of 23andme (May 2013), https://www.23andme.com/gen101/genes). These four nitrogenousbase are: A (adenine), T (thymine), G (guanine), C (cytosine). In a DNA double helix, each type of nucleobase on one strand bonds with just one type of nucleobase on the other strand. This is called complementary base pairing. For example, ‘A’ always pairs with ‘T’, and ‘G’ always pairs with ‘C’. (see Figure 3.2, image courtesy of 23andme (May 2013), https://www.23andme.com/gen101/genes)

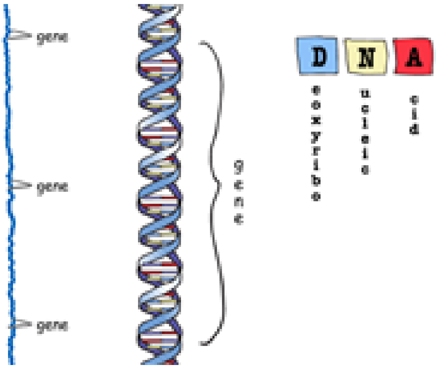


Figure 3.1. Structure of the DNA Molecule

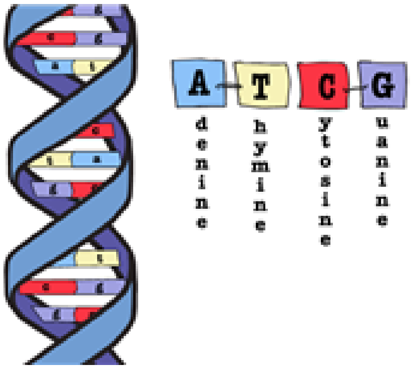


Figure 3.2. Pairing Base

A chromosome is an organized structure of the long molecules of DNA and protein found in the nucleus of cells. Humans have 23 pairs of chromosomes (see Figure 3.3, image courtesy of 23andme (May 2013), https://www.23andme.com/gen101/genes), chimpanzees have 24 pairs and bananas have 11 pairs of chromosomes. A chromosome consists of a single, very long DNA helix on which thousands of genes are encoded. In other words, genes are the short subunits of DNA containing in each chromosome. Each person has the same set of genes, about 20,000 in all. (see Figure 3.4, image courtesy of Wikipedia (2013), http://en.wikipedia.org/wiki/Intron)

Figure 3.3. Chromosomes: Human

Genes make proteins which results in different phenotypes. In the process of transfering the information from gene to protein requires two steps: the DNA on which the gene resides must be firstly transcribed from DNA to messenger RNA (mRNA) and then it translated from mRNA to protein. This process is called gene expression or we can say a gene is expressed when it is active in making protein. The total complement of genes in an organism or cell is called as its genome.

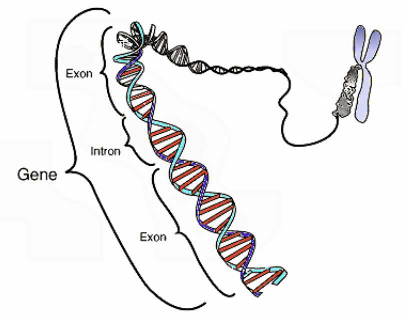


Figure 3.4. Gene

3.3.2. Microarray Technology. Genetic association studies aim to detect the associations between gene expressions and the occurrence or progression of disease phenotypes. Recent developments in microarray techniques make it possible to profile gene expressions on a whole genome scale, simultaneously measuring expressions of thousands or tens of thousands of genes. The DNA microarray, also known as DNA chip, is used to obtain the measurement of gene expression levels for all known genes in a genome (Griffiths et al., 2008). Affymetrix is one of the major companies produce gene chips in United States. Each gene chip contain 6.5 million of locations with millions of DNA strands built up in each location, and each DNA probe is 25 base pairs in length (See Figure 3.5, image courtesy of Affymetrix Image Library (2013), www.affymetrix.com). The process of measuring the gene expression by using Affymetrix gene expression arrays can be briefly described below (Olbricht, 2010) (See Figure 3.6 and Figure 3.7, image courtesy of Affymetrix Image Library (2013), www.affymetrix.com): 25 prespecified base pair DNA probes from a reference genome are first chosen as targets from a specific organism genes; then a fluorescent dye labeled mRNA sample is used to hybridized to the target array through the complementary base pairing principle; then it comes up wih two types of probes with a perfect match probe (PM) and a mismatch probe (MM); after hybridization, mRNA transcription levels for each probe are measured in the form of a quantitative intensity reading; and finally the Robust Multi-array Average (RMA) method is applied to summarize and normalize all the probe reading into one number for each gene, which makes it easier for further analysis.

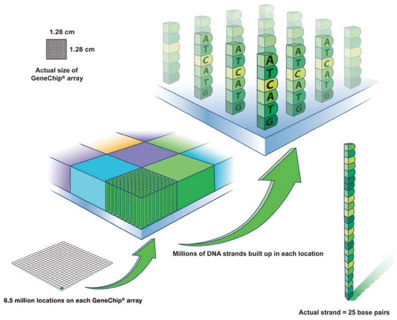

1.28 cm
1.28 cm
Actual size of GeneChip® array
ATCATG
Millions of DNA strands built up in each location
6.5 million locations on each GeneChip® array
Actual strand = 25 base pairs

Figure 3.5. Affymetrix Gene Chip Microarray

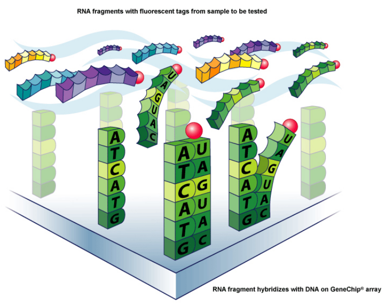

RNA fragments with fluorescent tags from sample to be tested
RNA fragment hybridizes with DNA on GeneChip® array

Figure 3.6. Hybridization of mRNA Sample to the Array: 1

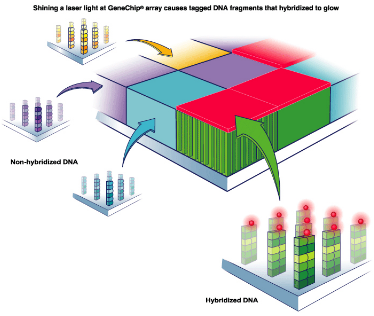

Shining a laser light at GeneChip® array causes tagged DNA fragments that hybridized to glow
Non-hybridized DNA
Hybridized DNA

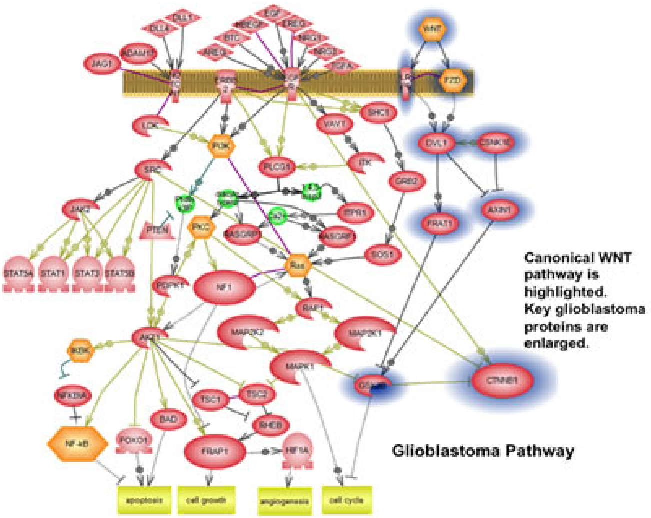
Figure 3.7. Hybridization of mRNA Sample to the Array: 2

3.3.3. Gene Pathway Analysis Reviews. New challenges arise for the analysis of microarray data due to the large number of genes surveyed and often the relatively small sample sizes. A large amount of existing approaches (to list a few: Alon et al., 1999; Dudoit et al., 2002; Nguyen and Rocke, 2002; Rosenwald et al., 2003) has been developed to identify a small subset of genes or linear combinations of genes which are often referred to as super genes, that have influential effects on diseases. Such studies can lead to a better understanding of the genetic causation of diseases and better predictive models.

However, since the presence of cluster structure of genes (gene pathways) was ignored, these methods are insufficient to dissect the complex genetic structure of many common diseases. It is well known that most biological units such as genes behave interactively by groups, that is, the pathway or genetic regulatory network (GRN). Here the clusters are composed of co-regulated genes with coordinated functions. Gene annotation databases, such as KEGG (Ogata et al., 2000), Reactome (Matthews et al., 2008), PID (http://pid.nci.nih.gov/) and BioCyc (Karp et al., 2005), group functionally relevant genes into biological pathways. Since it is commonly believed that genes carry out their functions through intricate pathways of reactions and interactions, intuitively, pathway-based analysis can offer an attractive alternative to improve the power of gene (or SNP)-based methods, and may help us to identify relevant subsets of genes in meaningful biological pathways underlying complex diseases.

There is considerable interest in pathway-based analyses (to list a few: Manoli et al., 2006; Wang et al., 2007; Li and Li, 2008; Wei and Pan, 2008; Ma and Kosorok, 2009; Pan et al., 2010; Zhu and Li, 2011). Pathway-based approaches in microarray data analysis often yield biological insights that are otherwise undetectable by focusing only on genes with the strongest evidence of differential expressions. Most pathway-based methods focus on identifying meaningful biological pathways underlying complex diseases, assuming that if a pathway (cluster) is strongly associated with the phenotype, then all genes within that pathway are associated with the phenotype. However, if only a subset of genes within a pathway contributes to the outcome, then these methods may result in loss of power. Our sparse group sufficient dimension reduction is developed to address this problem, where pathway selection and within pathway gene selection can be achieved simultaneously.

3.3.4. A Real Data Analysis. We hereby apply our method to a survival analysis for glioblastoma patients (Horvath et al., 2006) using gene expression profiles with about 1500 genes and 33 pathways. (See Figure 3.8, image courtesy of Next Generation Pharmaceutical (NGP), http://www.ngpharma.com/article/Mechanistic-Disease-Modeling (2013))



Canonical WNT pathway is highlighted. Key glioblastoma proteins are enlarged.

Glioblastoma Pathway

Figure 3.8. Glioblastoma

Glioblastoma is the most common and aggressive malignant brain tumor in humans. Patients with this disease have a median survival time of approximately 15 months from the time of diagnosis despite various treatments such as surgery, radiation and chemotherapy. Consisting of two independent sets of clinical tumor samples of $n = 55$ and $n = 65$, the dataset was obtained by Affymetrix HG-U133A arrays, and processed by the RMA method (Irizarry et al., 2003). As Pan et al. (2010) pointed out, the two datasets were somewhat different from each other, and they only used dataset one in their analysis. Following Pan et al. (2010), we also focus on the 50 patients with observed survival times from dataset one, and took the log survival time (in days) as the response variable in our analysis and the gene expression profiles as predictors. Our goal is to simultaneously identify significant pathways and genes within those pathways that are strongly associated with the survival time from glioblastoma.

We merged the gene-expression data with the 33 regulatory pathways recorded in the KEGG database. Among the 1668-node of the 33 pathways, 1507 (Entrez ID) out of 22283 genes (Probe ID) are identified on the HG-U133A chip. Following Li and Li (2008), Pan et al. (2010), and Zhu and Li (2011), we only use these 1507 genes in our following analysis. When there are multiple probe set ids corresponding to a single Entrez KEGG id, we took the average expression levels of those probe ids.

We compared our result with Li and Li (2008). As reported on Table 3.5, our pathway selection is similar to that of Li and Li (2008) except for pathway 6, 13, 18; 17 and 27 (Cell cycle, Extracellular matrix-receptor interaction, Gap junction, Complement and coagulation cascades, Type I diabetes mellitus). Among those five pathways, the first three pathways were selected by our method but not by Li and Li (2008), while the latter two were selected by Li and Li (2008) only. As reported in Sun, et al. (2012), the entire tumor growth profile in brain cancer is a collective behavior of cells regulated by the cell cycle pathway (pathway 6). The study result from Phillips laboratory (UCSF) shows that heparan sulfate proteoglycans (HSPGs) in extracellular matrix (pathway 13) can change tumor cell behavior including proliferation, invasion and recruitment of inflammatory cells. Zhu and Li (2011) ranked all the 33 pathways according to their significance, pathway 17 and 27 which were only selected by Li and Li (2008), ranked 30th and 28th, respectively, suggesting that they are not very important pathways.

MAPK signaling pathway (pathway 1), Cytokine-cytokine receptor interaction pathway (pathway 3), Neuroactive ligand-receptor interaction pathway (pathway 5), and Complement and coagulation cascades (pathway 18) were ranked as the top 4 significant pathways related to the brain cancer by Zhu and Li (2011) using a nonlinear dimension reduction method. Our pathway selection is consistent with Zhu and Li (2011), since all these 4 pathways are selected by sgSDR. For the within pathway gene selection, our method selected 85 unique genes. Among them, 10 genes are the same as that of Li and Li (2008), i.e., MAP3K7, CX3CL1, SYNJ2, UBE2E1, SMURF2, CLDN6, IRF3, IL21R, PCK1, FOXO1A. And FOXO1A was also identified by Pan et al. (2010) as one of the significant transcription factors associated with glioblastoma.

Table 3.5. Pathway Selections for Glioblastoma Data

Group	Pathway Name	sgSDR	Li and Li
1	MAPK signaling pathway	✓	✓
2	Calcium signaling pathway	✓	✓
3	Cytokine-cytokine receptor interaction	✓	✓
4	Phospatidylinositol signaling system	✓	✓
5	Neuroactive ligand-receptor interaction	✓	✓
6	Cell cycle	✓	
7	Ubiquitin mediated proteolysis	✓	✓
8	Apopttosis	✓	✓
9	Wnt signaling pathway	✓	✓
10	Transforming growth factor-beta signaling pathway	✓	✓
11	Axon guidance	✓	✓
12	Focal adhesion	✓	✓
13	Extracellular matrix-receptor interaction	✓	
14	Cell adhension molecules	✓	✓
15	Adherens junction	✓	✓
16	Tight junction	✓	✓
17	Gap junction		✓
18	Complement and coagulation cascades	✓	
19	Toll-like receptor signaling pathway	✓	✓
20	Jak-STAT signaling pathway	✓	✓
21	Natural killer cell mediated cytotoxicity	✓	✓
22	Circadian rhythm		
23	Regulation of actin cytotoxicity	✓	✓
24	Insulin signaling pathway	✓	✓
25	Adipocytokine signaling pathway	✓	✓
26	Type II diabetes mellitus	✓	✓
27	Type I diabetes mellitus		✓
28	Alzheimer's disease		
29	Prion diseases		
30	Cocaine addition		
31	Unknown		
32	Unknown		
33	Unknown		

3.4. DISCUSSION

In this dissertation, we propose a method called sgSDR within the framework of sufficient dimension reduction which could conduct group and within group variable selection simultaneously. Our method is comparable to the sparse group lasso (Friedman et al., 2010; Simon et al., 2012) for the linear models, and outperform it when the regression function is nonlinear. Also, our method is robust to the error distributions. A glioblastoma data is used to illustrate the applications of our method to the gene pathway analysis. As extensions to this paper, there are some possible future work.

- The best transformation of $\mathbf{Y}$.

  As shown in the second proposition in this section, any monotonic transformation of the response $\mathbf{Y}$ works for sgSDR. It is natural to wonder whether there is a best transformation of $\mathbf{Y}$. Examples of such topic can be refer to Yin and Li (2011).

- The consistency of modified BIC.

  The modified BIC used in this section can lead to a significantly lower FPR for both variable and group level selection. However, comparing to cross validation the TPR of modified BIC is slightly lower in a few cases.

- The consistency of our group and variable selections.

  The asymptotic properties of sgSDR and SGR deserve further investigations.

4. A NOTE ON CUMULATIVE MEAN ESTIMATION

4.1. INTRODUCTION

For many-valued or continuous Y, the standard practice in SDR is to replace the response Y with a discrete version $\check{Y}$ by partitioning the range of Y into h non-overlapping slices, then work on $\check{Y}$ and assume that $\mathcal{S}_{\check{Y}|\mathbf{X}} = \mathcal{S}_{Y|\mathbf{X}}$. However, this assumption is not always true, and the differences between the working and target regressions can be significant. Moreover, even under the case of equality, we might still face the loss of power since we use only the information retained in $\check{Y}$, discarding all the intra-slice information.

The number of slices h is a tuning parameter much like the tuning parameter encountered in the smoothing literature (Li, 1987; Härdle et al., 1988). Experience indicates that good results are often obtained by choosing h to be somewhat larger than $d+1$, trying a few different values of h as necessary. However, beyond empirical experience, how to select the optimal h is an open problem.

For the menthods mentioned in the first two sections, most of the sliced based methods such as sliced average variance estimation (SAVE) (Cook and Weisberg, 1991) and Directional regression (DR) (Li and Wang, 2007) heavily rely on the choice of the total number of slices h (Li and Zhu, 2007). Results from the above methods might vary significantly if a different h is applied. To the best of our knowledge, there are no existing criterion about the selection of the number of slices in the literature.

According to the empirical studies of SIR (Li, 1991), it is suggested that the number of slices h needs to be at least larger than the structural dimension d. Intuitively, more slices will lead to a better estimation since the integrity of $\mathcal{S}_{Y|\mathbf{X}}$ is well preserved. However, as pointed out by Zhu and Ng (1995), the increasing number of slices h will result in larger asymptotic variance since the number of data points within each slice is smaller. To preserve the integrity of $\mathcal{S}_{Y|\mathbf{X}}$ and maintain the estimation accuracy simultaneously, Zhu et al. (2010) proposed a method called the *cumulative slicing estimation* (CUME) which sums up all possible estimations relating to $\mathbb{E}(\mathbf{X}I(Y \leq \tilde{y}))$ for all $\tilde{y}$ in the support of Y to avoid the otherwise subjective selection of h. They showed that the estimator of

CUME enjoys the common $\sqrt{n}$ convergence rate and is more efficient comparing to SIR and other first-moment slicing estimation methods.

4.2. CUMULATIVE SLICING ESTIMATION (CUME)

In this subsection, we give a brief review of CUME (Zhu et al., 2010). For ease of exposition, we assume hereafter that $\mathbb{E}(\mathbf{X}) = 0$. Define

$$\mathbf{m}(\tilde{y}) = \mathbb{E}(\mathbf{X}I(Y \leq \tilde{y})) \tag{4.1}$$

for $\tilde{y} \in \mathbb{R}^1$. To preserve the integrity of $\mathcal{S}_{Y|\mathbf{X}}$, let $\tilde{Y}$ be an independent copy of Y and the kernel matrix for CUME is given by:

$$\mathbf{M} = \mathbb{E}[\mathbf{m}(\tilde{Y})\mathbf{m}^T(\tilde{Y})w(\tilde{Y})], \tag{4.2}$$

where $w(.)$ is a nonnegative weight function which is often set as 1.

Assuming the linearity condition (1.8), the column space of $\mathbf{M}$ is a subset of $\mathbf{\Sigma}\mathcal{S}_{Y|\mathbf{X}}$, where $\mathbf{\Sigma} = \mathbb{C}\text{ov}(\mathbf{X})$. At the sample level, suppose $(\mathbf{X}_i, Y_i)$, $i = 1, \cdots, n$ are independent copies of $(\mathbf{X}, Y)$, we can estimate $\mathbf{M}$ by

$$\mathbf{M}_n = \frac{1}{n}\sum_{i=1}^{n}\mathbf{m}_n(Y_i)\mathbf{m}_n^T(Y_i)w(Y_i),$$

where

$$\mathbf{m}_n(Y_i) = \frac{1}{n}\sum_{j=1}^{n}(\mathbf{X}_j - \bar{\mathbf{X}})I(Y_j \leq Y_i),$$

and

$$\bar{\mathbf{X}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{X}_i.$$

Let

$$\hat{\mathbf{\Sigma}} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{X}_i - \bar{\mathbf{X}})(\mathbf{X}_i - \bar{\mathbf{X}})^T$$

be the sample predictor variance, assuming a known d, the CUME estimator of $\mathcal{S}_{Y|\mathbf{X}}$ is constructed by the d eigenvectors of $\hat{\mathbf{\Sigma}}^{-1}\mathbf{M}_n$ corresponding to its d largest eigenvalues.

Zhu et al. (2010) also studied the asymptotic properties of the CUME estimator as shown in the following theorems.

Theorem 4.1. *Suppose* $max_{1 \leq i \leq p} E(\mathbf{X}_i^8) < \infty$ *uniformly for* p*, and then*

$$||\mathbf{\Sigma}_n^{-1}\mathbf{M}_n - \mathbf{\Sigma}^{-1}\mathbf{M}|| = O(pn^{-1/2}\log n)$$

almost surely where $|| \cdot ||$ *is the Frobenius norm.*

The above theorem shows the strong consistency when $p = O(pn^{-1/2}\log n)$. Particularly, the convergence rate of $||\mathbf{\Sigma}_n^{-1}\mathbf{M}_n - \mathbf{\Sigma}^{-1}\mathbf{M}||$ is reduced to $n^{-1/2}\log n$ when the dimension p is fix.

Theorem 4.2. *Assume the following regularity conditions:*

1. $max_{1 \leq i \leq p} E(\mathbf{X}_i^8) < \infty$ *uniformly for* p*;*

2. *The minimum eigenvalue of* $\mathbf{\Sigma}$ *satisfies* $\lambda_{min}(\mathbf{\Sigma}) > 0$*;*

3. *The largest eigenvalue of* $\mathbf{M}$ *satisfies* $\lambda_{max}(\mathbf{M}) < \infty$ *holds uniformly for* p*;*

4. $E\{\gamma^T T(\mathbf{X}, Y)\gamma\} \to G > 0$ *for any unit length* γ*;*

5. $p = o(n^{1/2})$.

Then

$$\sqrt{n}\gamma^T(\mathbf{\Sigma}_n^{-1}\mathbf{M}_n - \mathbf{\Sigma}^{-1}\mathbf{M})\gamma \to N(0, G)$$

in distribution.

Here

$$T(\mathbf{X}, Y) = \mathbf{\Sigma}^{-1}\{\mathbf{X}\mathbf{X}^T - E\mathbf{X}\mathbf{X}^T - (\mathbf{X} - E\mathbf{X})E\mathbf{X}^T - E\mathbf{X}(\mathbf{X} - E\mathbf{X})^T\}\mathbf{\Sigma}^{-1}\mathbf{M}$$

$$\text{-}\mathbf{\Sigma}^{-1}[2m(Y)m^T(Y)\omega(Y) + 2E\{(Y \leq \tilde{Y})m^T(\tilde{Y})\omega(\tilde{Y})|\mathbf{X}, Y\}$$

$$+2\mathrm{E}\{\mathrm{m}()\mathrm{X}^T Y^T I(Y \leq \tilde{Y})\omega(\tilde{Y})|\mathbf{X}, Y\} - 6\mathbf{M}]$$

and $\tilde{Y}$ is an independent copy of Y. This result states the asymptotic normality holds when $p = O(n^{1/2})$.

4.3. THE ENSEMBLE ESTIMATORS' APPROACH

The idea of ensemble approach is based on the fact that estimating the central mean subspace for a rich enough family of functions is the same as estimating the central subspace itself (Yin and Li, 2011). Several methods have been developed to combine central mean subspaces into the central subspace such as Cook and Li (2002), Yin and Cook (2002), Zhu and Zhu (2009), Xia (2007), Fukumizu, Bach and Jordan (2009), Wang and Xia (2008), Zhu and Zeng (2006). Yin and Li (2011) introduced a general method for combining estimators of a family of central mean subspaces into a single estimator of the central subspace using the MAVE-type procedures as basic estimators for the central mean subspaces. This ensemble estimators' approach (Yin and Li, 2011) unifies the central mean subspace (Cook and Li, 2002), the central moment subspace (Yin and Cook, 2002), Fourier transform estimators (Zhu and Zeng, 2006) and sliced regression (Wang and Xia, 2008) into a coherent system.

The main result of Yin and Li (2011) is summarized by the following theorem.

Theorem 4.3. *Let $\mathscr{J}$ be a family of functions $f : \Omega_Y \to \mathbb{F}$, F_Y be the distribution function of Y, $L_2(F_Y)$ be the class of functions $f(Y)$ with finite variances and $(f_1, f_2) = \mathbb{E}[f_1(Y)f_2(Y)]$ as the inner product. Let $\mathcal{S}_{\mathbb{E}[f(Y)|\mathbf{X}]}$ be the central mean subspace for the conditional mean $\mathbb{E}[f(Y)|\mathbf{X}]$, as defined in Cook and Li (2002). If $\mathscr{J}$ is a subset of $L_2(F_Y)$ that is dense in $\mathscr{B}$, where $\mathscr{B} = \{I_B : B\, is\, a\, Borel\, set\, in\, \Omega_Y\}$, then we have:*

$$\operatorname{span}\{\mathcal{S}_{[\mathbb{E}(f(Y)|\mathbf{X})]} : f \in \mathscr{J}\} = \mathcal{S}_{Y|\mathbf{X}}. \tag{4.3}$$

Hence, for a sufficiently rich family of $f(Y)$, the conditional mean subspaces $\mathcal{S}_{[\mathbb{E}(f(Y)|\mathbf{X}]}$, when put together, can recover the central subspace $\mathcal{S}_{Y|\mathbf{X}}$. Such family $\mathscr{J}$ is said as characterizing the central subspace. Yin and Li showed that both the Fourier transformation method proposed by Zhu and Zeng (2006), and the sliced regression (SR) proposed by Wang and Xia (2008) are special examples of the above ensemble estimators.

Example 4.4. *Zhu and Zeng (2006) used Fourier transformation or characteristic function $m(x,t)$ of the conditional density function $f_{(Y|X)}(y|x)$ which is defined as*

$$m(x,t) = E[T(Y,t)|X = x] = \int exp\{\imath ty\} f_{Y|X}(y|x)dy.$$

Zhu and Zeng (2006) showed that

$$\mathcal{S}_{Y|\mathbf{X}} = \sum_{t \in \mathbb{R}} S_{E[T(Y,t)|X]}$$

where $T(Y)$ is a transformation of Y. More specifically, $T(Y,t) = exp(\imath tY)$ in Zhu and Zeng's paper (2006). In Yin and Li (2011) notation, that is,

$$\mathscr{J} = \{f_t(y) = e^{\imath ty} : t \in \mathbb{R}^1\},$$

where $\imath$ is the imaginary unit. This family is dense in $L_2(F_Y)$. Zhu and Zeng also pointed out that it is not necessary to use all the possible transformation of Y, a properly chosen transformation family is good enough to recover the entire central subspace by collecting the central mean subspace of $T(Y)$ versus X.

Example 4.5. *In sliced regression (Wang and Xia, 2008), Wang and Xia proved that for any matrix $\boldsymbol{B}$,*

$$Y \perp\!\!\!\perp \mathbf{X} \mid \boldsymbol{B}^T\mathbf{X}$$

is equivalent to

$$P(Y \leq y|\mathbf{X} = x) = P(Y \leq y|\boldsymbol{B}^T\mathbf{X} = B^T x)$$

for all $y \in \mathbb{R}^1$ and $x \in \mathbb{R}^p$. In Yin and Li (2011) notation, well known that the above transformation family is dense in $L_2(F_Y)$. This result shows that the central subspace of Y is related closely to the central mean subspace of $I(Y \leq y)$. Moreover, the central subspace $\mathcal{S}_{Y|\mathbf{X}}$ can be fully recovered as long as the central mean subspace of $I(Y \leq y)|\mathbf{X}$ is estimated for all $y \in \mathbb{R}^1$.

The ensemble approach enjoys many advantages such as being able to estimate the central subspace, estimation accuracy and easy computation. However, there are some limitations of ensemble approach: the choice of $\mathscr{J}$ and the determination of the number m. It is difficult to derive a general criterion for all possible families. Also, the choice of m varies from a case to case basis. Theoretically, within the computational capacity, the larger m, the better results.

4.4. CUME REVISIT: THE ENSEMBLE ESTIMATORS' APPROACH

In this section, we revisit CUME via the ensemble approach's perspective (Yin and Li, 2011). We are now ready to demonstrate that CUME also belongs to the family of the ensemble estimators. Let

$$\mathscr{J} = \{f_t(y) = I_{(-\infty,t)}(y) : t \in \mathbb{R}^1\},$$

let $\boldsymbol{\eta}_t = \boldsymbol{\Sigma}^{-1}m(t)$ be the population coefficient vector from the ordinary least squares fit of of $f_t(Y)$ on $\mathbf{X}$ without the intercept term. Following the result of Duan and Li (1991), it is not hard to prove that the above OLS coefficient falls into the central mean subspace $\mathcal{S}_{\mathbb{E}[f_t(Y)|\mathbf{X}]}$. We use the following proposition to conclude our discussion of CUME in this section.

Proposition 3. *Let $\mathscr{J} = \{f_t(y) = I_{(-\infty,t)}(y) : t \in \mathbb{R}^1\}$, for $t = Y_1, \ldots, Y_n$, $\boldsymbol{\Sigma}^{-1}m(t) \in \mathcal{S}_{\mathbb{E}[f_t(Y)|\mathbf{X}]}$, and the column space of $\hat{\boldsymbol{\Sigma}}^{-1}\mathbf{M}_n$ provides a consistent estimator of $\mathcal{S}_{Y|\mathbf{X}}$.*

Proof: The proof partly relied on Theorem 1 of Cook and Li (2002), hence we include it here for reference.

Theorem 4.6. *(Cook and Li (2002)) Let γ be a basis matrix for $\mathcal{S}_{E(Y|Z)}$, assume that $E(\mathbb{Z}|\gamma^T\mathbb{Z})$ is a linear function of $\mathbb{Z}$ and let $\boldsymbol{\beta}$ be defined as (4.4) using an exponential family objective function (4.5), where $R(a,b)$ is a risk function and Φ is a strictly convex function. Then*

$$\boldsymbol{\beta} \in \mathcal{S}_{E(Y|Z)}$$

$$(\alpha, \beta) = argmin_{a,b}R(a,b) \tag{4.4}$$

$$L(a + b^T\mathbb{Z}, Y) = -Y(a + b^T\mathbb{Z}) + \Phi(a + b^T\mathbb{Z}) \tag{4.5}$$

The proof of the above proposition can be shown as the following:

- Proof of $\mathbf{\Sigma}^{-1}m(t) \in \mathcal{S}_{\mathbb{E}[f_t(Y)|\mathbf{X}]}$:

  Since $f_t(y) = I_{(-\infty,t)}(y)$, $\mathbf{\Sigma}^{-1}m(t)$ is the population coefficient vector from the ordinary least square fit of $f_t(Y)$ on $\mathbf{X}$ without the intercept term. By Theorem 3.1 and Theorem 1 of Cook and Li (2002), we can show that $\mathbf{\Sigma}^{-1}m(t) \in \mathcal{S}_{\mathbb{E}[f_t(Y)|\mathbf{X}]}$.

- Proof of consistency of $\hat{\mathbf{\Sigma}}^{-1}\mathbf{M}_n$:

  Since $\mathscr{J} = \{f_t(y) = I_{(-\infty,t)}(y) : t \in \mathbb{R}^1\}$ is dense in $L_2(F_Y)$, by Theorem 4.3, we have $\text{span}\{\mathcal{S}_{[\mathbb{E}(f(Y)|\mathbf{X})]} : f \in \mathscr{J}\} = \mathcal{S}_{Y|\mathbf{X}}$. Let $\hat{m}(t)$ denote the corresponding sample estimate of $m(t)$, since $\hat{\mathbf{\Sigma}}^{-1}\widehat{m(t)}$ is a consistent estimate of $\mathcal{S}_{[\mathbb{E}(f(Y)|\mathbf{X})]}$, $\hat{\mathbf{\Sigma}}^{-1}\mathbf{M}_n$ is also a consistent estimate of $\mathcal{S}_{Y|\mathbf{X}}$.

According to the above proposition, we can see that CUME is also a special example of the family of the ensemble estimators.

4.5. COVARIANCE INVERSE REGRESSION ESTIMATION (CIRE)

In this subsection, we will introduce the idea of covariance inverse regression estimation methods as well as how it motivates our method. For most sufficient dimension applications, if the response Y is discrete or categorical, conditional sample means can be used to estimate $E(X|Y)$. Meanwhile, if the response Y is continuous, $E(X|Y)$ is usually estimated by replacing Y with a discrete version of Y through partitioning Y into h slices as introduced by Li (1991). For example, in SIR, let $\Phi(y)$ defined as in (4.6), Li (1991) showed that the expected value of $\Phi(Y)$ in the s^{th} slice,

$$\xi_s = E\{\Phi(Y)|J_s = 1\} = \mathbf{\Sigma}^{-1}\{E(X|J_s = 1) - E(X)\} \in \mathcal{S}_{Y|\mathbf{X}}$$

and

$$\text{span}(\text{Var}(\xi(y))) \subseteq \mathcal{S}_{Y|\mathbf{X}}$$

where Y is substituted by the h slices of Y in the sample level and

$$\Phi(y) = \mathbf{\Sigma}^{-1}\{E(X|Y = y) - E(X)\} = \mathbf{\Sigma}^{-1/2}E(Z|Y = y) \in \mathcal{S}_{Y|\mathbf{X}}. \tag{4.6}$$

However, it is clear that replacing Y by a sliced version of Y can result in loss of information. In order to recover information missed by the sliced means, Cook and Ni (2006) proposed an improved method by incorporating the intraslice covariances into the central subspace estimation. This method (Cook and Ni, 2006) can be described in the following way.

We first define

$$\varsigma_s(Y) = \Phi(y) - \xi_s = \mathbf{\Sigma}^{-1}\{E(X|Y = y) - E(X|J_s = 1)\} \in \mathcal{S}_{Y|\mathbf{X}}$$

and intraslice covariance

$$\mathbb{C}\mathrm{ov}\{\varsigma_s(Y), Y|J_s = 1\} = \mathbf{\Sigma}^{-1}\,\mathbb{C}\mathrm{ov}\{X, Y|J_s = 1\} \in \mathcal{S}_{Y|\mathbf{X}}$$

where

$$J_s(Y) = \begin{cases} 1 & Y\,is\,in\,slice\,s \\ 0 & otherwise \end{cases}$$

The intraslice covariance is added to ς_s to remedy the loss of information through slicing.

$$\beta_s = \mathbf{\Sigma}^{-1}\,\mathbb{C}\mathrm{ov}(X, YJ_s) = f_s\mathbf{\Sigma}^{-1}\,\mathbb{C}\mathrm{ov}(\mathbf{X}, Y|J_s = 1) + f_s\mathbb{E}(Y|J_s = 1)\xi_s$$

On the sample level, consider a random sample $(\mathbf{X}_i, Y_i)$ for $i = 1, \ldots, n$. For traditional methods where intraslice variance does not involve,

$$\hat{\varsigma}_s = \hat{\mathbf{\Sigma}}^{-1/2}\frac{\sum\limits_{i=1}^{n} J_s(Y_i)\hat{Z}_i}{\sum\limits_{i=1}^{n} J_s(Y_i)}.$$

For CIRE, the basis in slice s can be estimated as

$$\hat{\beta}_s = \hat{\mathbf{\Sigma}}^{-1/2}\frac{1}{n}\sum_{i=1}^{n} Y_i J_s(Y_i)\hat{Z}_i$$

where $\hat{\mathbf{\Sigma}}$ is the sample covariance matrix of $\mathbf{X}$ and $\hat{Z}_i = \hat{\mathbf{\Sigma}}^{-1/2}(X_i - (\bar{\mathbf{X}}))$. It is easy to see that comparing to other tradition SDR methods, CIRE considers the specific values of Y not just the discretized version of Y. This is meaningful especially when the value of Y fluctuates significantly in a big range.

4.6. COVARIANCE CUMULATIVE SLICING ESTIMATION (COCUM)

4.6.1. The Method.

As Cook and Ni (2006) pointed out, the use of

$$\boldsymbol{\eta}_t = \Sigma^{-1} m(t)$$

discards the intra-slice information which might result in a loss of power. In this subsection, we propose a method called the *covariance cumulative slicing estimation* (COCUM) which incorporates the intra-slice information into the estimation of the central subspace $\mathcal{S}_{f_t(Y)|\mathbf{X}}$.

To recover the intra-slice covariance information, following Cook and Ni (2006), we take

$$\mathbf{m}_c(\tilde{y}) = \mathbb{E}(\mathbf{X}YI(Y \leq \tilde{y})) \tag{4.7}$$

for $\tilde{y} \in \mathbb{R}^1$. The kernel matrix for COCUM, $\mathbf{M}_c$, is constructed similar as (4.2) except replacing $\mathbf{m}(\tilde{y})$ with $\mathbf{m}_c(\tilde{y})$.

Let $F_t = P(Y \leq t)$ and denote

$$\boldsymbol{\beta}_t = \mathbf{\Sigma}^{-1}\mathbb{E}(\mathbf{X}YI(Y \leq t)),$$

it is easy to show that $\boldsymbol{\beta}_t$ can be decomposed as

$$F_t\mathbf{\Sigma}^{-1}\,\mathbb{C}\text{ov}(\mathbf{X}, Y|I(Y \leq t) = 1) + \mathbb{E}(Y|I(Y \leq t) = 1)\boldsymbol{\eta}_t.$$

Proposition 4. *Assuming the common linearity condition, then $\boldsymbol{\beta}_t \in \mathcal{S}_{Y|\mathbf{X}}$, for $t \in \mathbb{R}^1$. Furthermore, comparing with CUME, the column space of $\boldsymbol{\Sigma}^{-1}\mathbf{M}_c$ always encloses that of $\boldsymbol{\Sigma}^{-1}\mathbf{M}$.*

Proof: Let

$$g_t(y) = yI(y \le t) = \begin{cases} y & y \le t \\ 0 & y > t \end{cases}$$

$$h_t(y) = I(y \le t) = \begin{cases} 1 & y \le t \\ 0 & y > t \end{cases}$$

and

$$m(t) = \boldsymbol{\Sigma}^{-1}\,\mathbb{C}\mathrm{ov}\{h_t(y), \mathbf{X}\}$$

$$m_c(t) = \boldsymbol{\Sigma}^{-1}\,\mathbb{C}\mathrm{ov}\{g_t(y), \mathbf{X}\}.$$

In the previous section, we have proved that

$$\boldsymbol{\Sigma}^{-1}\,\mathbb{C}\mathrm{ov}\{\mathbf{X}, h_t(Y)\} \in S_{h_t(Y)|\mathbf{X}},$$

where $h(\cdot)$ is any transformation of y, and hence we can obtain that

$$m(t) \in S_{h_t(Y)|X}$$

and

$$m_c(t) \in S_{g_t(Y)|X}.$$

It is clear that for $\alpha \in \mathbb{R}^{p\times d}$,

$$g_t(y) \perp\!\!\!\perp \mathbf{X} \mid \alpha^T\mathbf{X} \Rightarrow h_t(y) \perp\!\!\!\perp \mathbf{X} \mid \alpha^T\mathbf{X},$$

but the other direction does not necessarily hold. Therefore, we can conclude that

$$S_{h_t(Y)|\mathbf{X}} \subseteq S_{g_t(Y)|\mathbf{X}}.$$

Also since it is shown in the previous proposition that

$$\mathbf{\Sigma}^{-1}\mathbf{M}_c = S_{g_t(Y)|\mathbf{X}}$$

and

$$\mathbf{\Sigma}^{-1}\mathbf{M} = S_{h_t(Y)|\mathbf{X}}.$$

Therefore,

$$\mathbf{\Sigma}^{-1}\mathbf{M} \subseteq \mathbf{\Sigma}^{-1}\mathbf{M}_c.$$

Proposition 4 suggests that theoretically COCUM always outperform CUME since it recovers more of the central subspace.

4.6.2. The Asymptotic Properties. The asymptotic properties of COCUM are shown in this subsection.

Theorem 4.7. *Let X_i be the i^{th} coordinate of $\mathbf{X}$, suppose $max_{1\leq i\leq p}E(X_i^8Y^8) < \infty$ uniformly for p, and then*

$$||\mathbf{\Sigma}_n^{-1}\mathbf{M}_n^c - \mathbf{\Sigma}^{-1}\mathbf{M}_c|| = o(pn^{-1/2}\log n)$$

almost surely where $||\cdot||$ is the Frobenius norm, where $\mathbf{M}_n^c = \frac{1}{n}\sum_{i=1}^{n}\mathbf{m}_c(Y_i)\mathbf{m}_c^T(Y_i)w(Y_i)$.

Proof: This proof is similar to CUME (Zhu et al., 2010). Here, note that $\mathbf{\Sigma} = \mathbb{C}\text{ov}(\mathbf{X})$, and $\mathbf{m}_c(\tilde{y})$ as defined in (4.7).

Step 0**:** Note that

$$\mathbf{\Sigma}_n^{-1}\mathbf{M}_n^c - \mathbf{\Sigma}^{-1}\mathbf{M}_c = \mathbf{\Sigma}^{-1}(\mathbf{\Sigma}-\mathbf{\Sigma}_n)\mathbf{\Sigma}_n^{-1}(\mathbf{M}_n^c-\mathbf{M}_c)+\mathbf{\Sigma}^{-1}(\mathbf{\Sigma}-\mathbf{\Sigma}_n)\mathbf{\Sigma}_n^{-1}\mathbf{M}_c+\mathbf{\Sigma}^{-1}(\mathbf{M}_n^c-\mathbf{M}_c).$$

Therefore, it suffices to study the convergence order of $||\mathbf{\Sigma}_n - \mathbf{\Sigma}||$ and $||\mathbf{M}_n^c - \mathbf{M}_c||$.

Step 1**:** Show

$$||\mathbf{M}_n^c - \mathbf{M}_c|| = o(p\,logn/\sqrt{n})$$

almost surely.

Write $\mathbf{M}_n^c$ as a standard U-Statistic as:

$$U_{n1} = \frac{2}{n(n-1)(n-2)} \sum_{i<j<k} \{\mathbf{X}_j \mathbf{X}_k^T Y_j Y_k^T I(Y_j \leq Y_i) I(Y_k \leq Y_i) w(Y_i)$$

$$+\mathrm{X}_i \mathbf{X}_k^T Y_i Y_k^T I(Y_i \leq Y_j) I(Y_k \leq Y_j) w(Y_j)$$

$$+\mathrm{X}_j \mathbf{X}_i^T Y_j Y_i^T I(Y_j \leq Y_k) I(Y_i \leq Y_k) w(Y_k)\}$$

The projection of U_{n1} is:

$$\hat{U}_{n1} = \sum_{i=1}^{n} E(\mathbf{M}_n^c | \mathbf{X}_i, Y_i) - (n-1) E(\mathbf{M}_n^c).$$

Step 1.1**:** Show

$$||U_{n1} - \hat{U}_{n1}|| = o(p\,logn/n) \tag{4.8}$$

almost surely. Let $R_n = U_{n1} - \hat{U}_{n1}$ is also a U-Statistic, according to Serfling (page 183),

$$var(R_n) = \{\frac{6}{n(n-1)(n-2)}\}\{\frac{3(n-3)(n-4)}{2}\xi_1 + 3(n-3)\xi_2 + \xi_3\},$$

where

$$\xi_c = var\{h_c(\mathbf{X}_1, \ldots, \mathbf{X}_c)\}$$

and $h_1 = 0$ in this case and hence $\xi_1 = 0$. Therefore,

$$var(R_n) = \frac{18(n-3)\xi_2 + 6\xi_3}{n(n-1)(n-2)}.$$

According to Lemma B in Serfling (page 68), since $max_{1 \leq i \leq p} E(X_i^8 Y^8) < \infty$ uniformly, so we can imply that $E|\sum_{i=1}^{p} X_i Y|^8 = O(p^4)$, and hence $||\xi_i|| = O(p^2)$ for $i = 2, 3$. Refer Serfling (page 182) for more details about h_c. Therefore, we have $var(U_{n1} - \hat{U}_{n1}) = O(\frac{p^2}{n^2})$.

Step 1.2**:** In this step, a stronger consistency will be proved. Based on Serfling (page 189), let

$$\lambda_n = (\frac{p\,logn}{n})^{-1}.$$

It suffices to show that, for any $\varepsilon > 0$, $\lambda_n||R_n|| < \varepsilon$ holds almost surely, for $n \to \infty$. In other words, that is

$$P(limsup\lambda_n||R_n|| > \varepsilon) = 0. \tag{4.9}$$

Applying Borel-Cantelli Lemma:

$$\sum_{k=0}^{\infty} P(\lambda_{2^{k+1}} max_{2^k \leq n \leq 2^{k+1}} ||R_n|| > \varepsilon) < \infty. \tag{4.10}$$

Since R_n is a reverse martingale (Serfling (page 177)), then

$$P(sup_{j\geq n}||R_j|| > t) \leq t^{-2}E||R_n||^2.$$

Also since the k^{th} term of (4.10) is bounded by

$$\varepsilon^{-2}\lambda_{2^{k+1}}E|U_{2^k} - \hat{U}_{2^k}|^2 = O((k+1)^{-2}).$$

Therefore (4.10) is convergent. Borel-Cantelli Lemma shows that

$$||R_n|| = o(p\,logn/n)$$

almost surely. Therefore,

$$||U_{n1} - \hat{U}_{n1}|| = o(p\,logn/n).$$

Step 2**:** We will show that

$$||\hat{U}_{n1} - \mathbf{M}_c|| = O(p\,logn/\sqrt{n}) \tag{4.11}$$

almost surely. We rewrite $\hat{U}_{n1}$ in the following way:

$$\hat{U}_{n1} = \frac{2}{n}\sum_{i=1}^{n}[E\{\mathbf{X}_j\mathbf{X}_k^T Y_j Y_k^T I(Y_j \leq Y_i)I(Y_k \leq Y_i)w(Y_i)|\mathbf{X}_i, Y_i\}$$

$$+\text{E}\{\text{X}_i\mathbf{X}_k^T Y_i Y_k^T I(Y_i \leq Y_j)I(Y_k \leq Y_j)w(Y_j)|\mathbf{X}_i, Y_i\}$$

$$+\mathrm{E}\{\mathrm{X}_j\mathbf{X}_i^TY_jY_i^TI(Y_j \leq Y_k)I(Y_i \leq Y_k)w(Y_k)|\mathbf{X}_i, Y_i\}] - 5\mathbf{M}_c,$$

that is

$$\hat{U}_{n1} = 2I_1 + 2I_2 + 2I_3 + \mathbf{M},$$

where

$$I_1 = \frac{1}{n}\sum_{i=1}^{n}[E\{\mathbf{X}_j\mathbf{X}_k^TY_jY_k^TI(Y_j \leq Y_i)I(Y_k \leq Y_i)w(Y_i)|\mathbf{X}_i, Y_i\} - \mathbf{M}_c$$

$$=\mathrm{n}^{-1}\sum_{i=1}^{n}\mathbf{m}_c(Y_i)\mathbf{m}_c^T(Y_i)w(Y_i) - \mathbf{M}_c,$$

$$I_2 = \frac{1}{n}\sum_{i=1}^{n}[E\{\mathbf{X}_i\mathbf{X}_k^TY_iY_k^TI(Y_i \leq Y_j)I(Y_k \leq Y_j)w(Y_j)|\mathbf{X}_i, Y_i\} - \mathbf{M}_c$$

$$=\mathrm{n}^{-1}\sum_{i=1}^{n}\mathbf{m}_c(Y_i)\mathbf{m}_c^T(Y_i)w(Y_i) - \mathbf{M}_c,$$

and

$$I_3 = \frac{1}{n}\sum_{i=1}^{n}[E\{\mathbf{X}_j\mathbf{X}_i^TY_jY_i^TI(Y_j \leq Y_k)I(Y_i \leq Y_k)w(Y_k)|\mathbf{X}_i, Y_i\} - \mathbf{M}_c$$

$$=\mathrm{n}^{-1}\sum_{i=1}^{n}\mathbf{m}_c(Y_i)\mathbf{m}_c^T(Y_i)w(Y_i) - \mathbf{M}_c.$$

From the previously step, we can prove

$$||I_1|| = o(p\,logn/\sqrt{n})$$

almost surely. Similar arguments can be used to proof

$$||I_2|| = o(p\,logn/\sqrt{n})$$

and

$$||I_3|| = o(p\, logn/\sqrt{n}).$$

Step 3**:** A similar technique can be clearly applied to proof

$$||\mathbf{\Sigma}_n - \mathbf{\Sigma}|| = o(p\, logn/\sqrt{n})$$

almost surely.

This theorem shows that COCUM posesses the same asymptotic property as CUME. Zhu, Miao and Peng (2006), derived the strong consistency for the slicing estimation of the SIR matrix when $p = o(n^{1/4})$. However, the results from both CUME and COCUM are faster than Zhu, Miao and Peng (2006) obtained.

Theorem 4.8. *Assume the following regularity conditions:*

1. $max_{1\leq i\leq p}E(X_i^8Y^8) < \infty$ *uniformly for* p;

2. *The minimum eigenvalue of* $\mathbf{\Sigma}$ *satisfies* $\lambda_{min}(\mathbf{\Sigma}) > 0$;

3. *The largest eigenvalue of* $\mathbf{M}_c$ *satisfies* $\lambda_{max}(\mathbf{M}_c) < \infty$ *holds uniformly for* p;

4. $E\{\gamma^T T(\mathbf{X}, Y)\gamma\} \to G > 0$ *for any unit length* γ;

5. $p = o(n^{1/2})$.

Then

$$\sqrt{n}\gamma^T(\mathbf{\Sigma}_n^{-1}\mathbf{M}_n - \mathbf{\Sigma}^{-1}\mathbf{M})\gamma \to N(0, G)$$

in distribution.

Here

$$T(\mathbf{X}, \mathbf{Y}) = \mathbf{\Sigma}^{-1}\{\mathbf{X}\mathbf{X}^T - E\mathbf{X}\mathbf{X}^T - (\mathbf{X} - E\mathbf{X})E\mathbf{X}^T - E\mathbf{X}(\mathbf{X} - E\mathbf{X})^T\}\mathbf{\Sigma}^{-1}\mathbf{M}_c$$

$$\text{-}\mathbf{\Sigma}^{-1}[2\mathbf{m}_c(Y)\mathbf{m}_c^T(Y)\omega(Y) + 2E\{\mathbf{X}YI(Y \leq \tilde{Y})\mathbf{m}_c^T(\tilde{Y})\omega(\tilde{Y})|\mathbf{X}, Y\}$$

$$+2\mathrm{E}\{\mathrm{m}_c(\tilde{Y})\mathbf{X}^T Y^T I(Y \le \tilde{Y})\omega(\tilde{Y})|\mathbf{X}, Y\} - 6\mathbf{M}_c]$$

and $\tilde{Y}$ is an independent copy of Y.

Proof: This proof is similar to CUME (Zhu et al., 2010). The entire proof contains the following steps:

Step 1.1**:** Show

$$||\mathbf{\Sigma}_n - \mathbf{\Sigma} - T_{n1}|| = O_p(p/n), \tag{4.12}$$

where

$$T_{n1} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{X}_i\mathbf{X}_i^T - E\mathbf{X}\mathbf{X}^T - (\bar{\mathbf{X}} - E\mathbf{X})E\mathbf{X}^T - E\mathbf{X}(\bar{\mathbf{X}} - E\mathbf{X})^T.$$

Reason: note that

$$P(||\mathbf{\Sigma}_n - \mathbf{\Sigma} - T_{n1}|| > \epsilon) = P(||(\bar{\mathbf{X}} - E\mathbf{X})(\bar{\mathbf{X}} - E\mathbf{X})^T|| > \epsilon)$$

$$\le E||(\bar{\mathbf{X}} - E\mathbf{X})^T(\bar{\mathbf{X}} - E\mathbf{X})||/\epsilon$$

$$= \mathrm{E(X\text{-}EX)}^T(\mathbf{X} - E\mathbf{X})/(n\epsilon)$$

$$=\mathrm{O}_p(p/n).$$

Step 1.2**:** Show

$$||\mathbf{M}_n^c - \mathbf{M}_c - T_{n2}|| = O_p(p/n), \tag{4.13}$$

where

$$T_{n2} = \frac{2}{n}\sum_{i=1}^{n} \mathbf{m}_c(Y_i)\mathbf{m}_c(Y_i)^T\omega(Y_i)$$

$$+ \ 2\mathrm{n}^{-1}\sum_{i=1}^{n} E\{\mathbf{X}_i Y_i I(Y_i \le Y)\mathbf{m}_c^T(Y)\omega(Y)|\mathbf{X}_i, Y_i\}$$

$$+ \ 2\mathrm{n}^{-1}\sum_{i=1}^{n} E\{\mathbf{m}_c(Y)\mathbf{X}_i^T Y_i^T I(Y_i \le Y)\omega(Y)|\mathbf{X}_i, Y_i\} - 6\mathbf{M}_c$$

Since $max_{1\leq i\leq p}E(X_i^2Y^2)<\infty$ uniformly for p, then

$$||\mathbf{M}_n^c-\mathbf{M}_c-T_{n2}||=||U_{n1}-\hat{U}_{n1}||=O_p(p/n)$$

Step 1.3**:**

$$||(\mathbf{\Sigma}_n^{-1}\mathbf{M}_n^c-\mathbf{\Sigma}^{-1}\mathbf{M}_c)+\mathbf{\Sigma}_n^{-1}T_{n1}\mathbf{\Sigma}^{-1}\mathbf{M}_c-\mathbf{\Sigma}_n^{-1}T_{n2}||$$

$$=\left\|\mathbf{\Sigma}_n^{-1}(\mathbf{M}_n^c-\mathbf{M}_c-T_{n2})-\mathbf{\Sigma}_n^{-1}(\mathbf{\Sigma}_n-\mathbf{\Sigma}-T_{n1})\mathbf{\Sigma}^{-1}\mathbf{M}_c\right||$$
$$=\mathrm{O}_p(p/n).$$

Step 2**:** In this step, we will prove the normality for

$$\gamma^T(\mathbf{\Sigma}_n^{-1}\mathbf{M}_n^c-\mathbf{\Sigma}^{-1}\mathbf{M}_c)\gamma$$

where γ is a unit-length vector. Let $\frac{T(\mathbf{X}_i,Y_i)}{n}$ as the i^{th} summand in $\mathbf{\Sigma}^{-1}T_{n1}\mathbf{\Sigma}^{-1}\mathbf{M}_c-\mathbf{\Sigma}^{-1}T_{n2}$, and let

$$\mathbb{Z}_{ni}=\gamma^TT(\mathbf{X}_i,Y_i)\gamma/\sqrt{n},$$

for $i=1,2,\ldots,n$.

Step 2.1**:** We have

$$\sum_{i=1}^n var(\mathbb{Z}_{ni})=E\{\gamma^TT(\mathbf{X}_i,Y_i)\gamma\}\longrightarrow G \tag{4.14}$$

as

$$E\{\gamma^TT(\mathbf{X}_i,Y_i)\gamma\}\longrightarrow G.$$

Also, for any given $\epsilon>0$, and as $n\to\infty$,

$$\sum_{i=1}^n E|\mathbb{Z}_{ni}|^2I(|\mathbb{Z}_{ni}|\geq\epsilon)$$

$$=\text{n E}(|\mathbb{Z}_{n1}|^2I(|\mathbb{Z}_{n1}|\geq\epsilon))$$

$$\leq n(E|\mathbb{Z}_{n1}|^4)^{1/2}P\{|\mathbb{Z}_{n1}| \geq \epsilon\}$$

$$\leq \{E|\gamma^T T(\mathbf{X}, Y)\gamma|^4\}^{1/2}P\{|\gamma^T T(\mathbf{X}, Y)\gamma| \geq \sqrt{n}\epsilon\}$$

$$\leq \lambda_{max}^{1/2}\{E|T^4(\mathbf{X}, Y)|\}P\{|\gamma^T T(\mathbf{X}, Y)\gamma| \geq \sqrt{n}\epsilon\}$$

$$\leq \lambda_{max}^{1/2}\{E|\mathbf{\Sigma}^{-1}T_1\mathbf{\Sigma}^{-1}\mathbf{M}_c|^4 + E|\mathbf{\Sigma}^{-1}T_2|^4\}P\{|\gamma^T T(\mathbf{X}, Y)\gamma| \geq \sqrt{n}\epsilon\}$$

$$\leq \lambda_{max}^{1/2}\{E|T_1(\mathbf{X})|^4 + E|T_2(\mathbf{X}, Y)|^4\}P\{|\gamma^T T(\mathbf{X}, Y)\gamma| \geq \sqrt{n}\epsilon\}$$

Apply condition 2 and condition 3 in the theorem, then the last inequality holds. Since $max_{1\leq i\leq p}E(X_i^8 Y^8) < \infty$ uniformly for p, we have

$$\lambda_{max}^{1/2}\{E|T_1(\mathbf{X})|^4 + E|T_2(\mathbf{X}, Y)|^4\} = O(p).$$

Moreover, the Markov inequality entails that

$$P\{|\gamma^T T(\mathbf{X}, Y)\gamma| \geq \sqrt{n}\epsilon\} \leq \frac{E\{|\gamma^T T(\mathbf{X}, Y)\gamma|\}}{\sqrt{n}\epsilon}.$$

Therefore,

$$\sum_{i=1}^{n} E|\mathbb{Z}_{ni}|^2 I(|\mathbb{Z}_{ni}| \geq \epsilon) = O(p/\sqrt{n}) \to 0$$

Together with (4.14), we can see that $\sum_{i=1}^{n}\mathbb{Z}_{ni}$ satisfies the conditions of the Lindeberg-Feller central limit theorem.

Step 2.2**:** Show

$$\gamma^T\{(\mathbf{\Sigma}_n^{-1} - \mathbf{\Sigma}^{-1})T_{n1}\mathbf{\Sigma}^{-1}\mathbf{M}_c - (\mathbf{\Sigma}_n^{-1} - \mathbf{\Sigma}^{-1})T_{n2}\}\gamma$$

is bounded. Firstly, by Cauchy-Schwarz inequality, we have

$$|\gamma^T(\mathbf{\Sigma}_n^{-1} - \mathbf{\Sigma}^{-1})T_{n1}\mathbf{\Sigma}^{-1}\mathbf{M}_c\gamma| \leq ||\gamma^T(\mathbf{\Sigma}_n^{-1} - \mathbf{\Sigma}^{-1})||||T_{n1}\mathbf{\Sigma}^{-1}\mathbf{M}_c\gamma||.$$

Since all the elements of $\mathbf{\Sigma}_n^{-1} - \mathbf{\Sigma}^{-1}$ are of the rate $n^{-1/2}$, and then it is easy to get that

$$||\gamma^T(\mathbf{\Sigma}_n^{-1} - \mathbf{\Sigma}^{-1})|| = O_p(\sqrt{p/n}).$$

Rewrite $\mathbf{\Sigma}^{-1}\mathbf{M}_c\gamma := \alpha$ and $||\alpha|| < \infty$. Therefore,

$$||T_{n1}\mathbf{\Sigma}^{-1}\mathbf{M}_c\gamma|| \leq O_p(\sqrt{p/n})$$

and hence

$$|\gamma^T(\mathbf{\Sigma}_n^{-1} - \mathbf{\Sigma}^{-1})T_{n1}\mathbf{\Sigma}^{-1}\mathbf{M}_c\gamma| = O_p(p/n).$$

Similarly, we have

$$|\gamma^T(\mathbf{\Sigma}_n^{-1} - \mathbf{\Sigma}^{-1})T_{n2}\gamma| = O_p(p/n).$$

Both of them show the convergence rate $O_p(p/n)$ of $\gamma^T\{(\mathbf{\Sigma}_n^{-1} - \mathbf{\Sigma}^{-1})T_{n1}\mathbf{\Sigma}^{-1}\mathbf{M}_c - (\mathbf{\Sigma}_n^{-1} - \mathbf{\Sigma}^{-1})T_{n2}\}\gamma$. Therefore, $\sqrt{n}\gamma^T(\mathbf{\Sigma}_n^{-1}\mathbf{M}_n - \mathbf{\Sigma}^{-1}\mathbf{M})\gamma$ is asymptotically normal.

The above theorem shows that the asymptotic normality holds for $p = o(n^{1/2})$, which is better than the rate $p = o(n^{1/3})$ in some literature (Fan and Peng, 2004; Zhu and Zhu, 2009).

4.6.3. The Determination of d. One of the goals for sufficient dimension reduction is to estimate the structural dimension. Many methods have been developed to determine the structural dimension, such as Li (1991), Schott (1994), Bura and Cook (2001), Zhu, Miao and Peng (2006), Zhu, Wang, Zhu and Ferre (2010). We list the details for some of the above methods.

The idea of the criterion used by Schott (1994) is based on a testing process. The test procedure with

$$H_0: \ k = m \ v.s. \ H_1: \ k > m \tag{4.15}$$

test the value of m starting at 0. The value of m increases by 1 each time until the null hypothesis is rejected. Define

$$\hat{W}_1 = \hat{\Omega}^{-1/2}\hat{\Delta}\hat{\Omega}^{-1/2};$$

$$\hat{W}_2 = \sum_{i=1}^{h}(\hat{\Omega}^{-1/2}\hat{\Omega}_i\hat{\Omega}^{-1/2} - \hat{\tau}_i I)^2/h;$$

$$\hat{W}_3 = \hat{W}_1 + \hat{W}_2,$$

where $\hat{\Delta} = \sum_{i=1}^{h}(\bar{\mathbf{X}}_i - \bar{\mathbf{X}})(\bar{\mathbf{X}}_i - \bar{\mathbf{X}})^T/h$, h is the total number of slices with $\hat{\Omega}_i$ as the sample covariance matrix from the i^{th} slice and $\hat{\Omega} = (\hat{\Omega}_1 + \ldots + \hat{\Omega}_h)/h$. Also, $\hat{\tau}_i$ is specified as

$$\hat{\tau}_i = trace(P^*\hat{\Omega}^{-1/2}\hat{\Omega}_i\hat{\Omega}^{-1/2}P^*)/(p-m)$$

and P^* is the eigen-projection of

$$W_2^* = \sum_{i=1}^{h}(\Omega^{-1/2}\Omega_i\Omega^{-1/2} - I)^2/h$$

corresponding to its $p-m$ smallest latent roots and p is the dimension of predictors. The test statistic for testing (4.15) is the average of the $p-m$ smallest latent roots of $\hat{W}_i$. This testing procedure carries on until H_0 is rejected.

BIC-type methods are also popular, especially for high-dimensional covariates. The procedure is easy to implement and the estimate is consistent. Both Zhu, Miao and Peng (2006) and Zhu, Wang, Zhu and Ferre (2010) applied the BIC-type method to determine the structural dimension d. Zhu, Miao and Peng (2006) sets

$$G(k) = \frac{n}{2}\sum_{i=1+min(\tau,k)}^{p}(log\hat{\theta}_i + 1 - \hat{\theta}_i) - \frac{C_n k(2p-k+1)}{2} \tag{4.16}$$

where $\hat{\theta}_1 \geq \hat{\theta}_2 \geq \ldots \geq \hat{\theta}_p$ are the eigenvalues of $\widehat{\Omega} = \widehat{\mathbb{C}\mathrm{ov}}\{E(\mathbf{X}|y)\} + I_p$. The second term on (4.16) is a penalty term with C_n as a penalty constant which is specified by a data-driven manner.

Also, Zhu, Wang, Zhu and Ferre (2010) uses

$$G(k) = \frac{n\sum_{l=1}^{k}\{log(\hat{\lambda}_l + 1) - \hat{\lambda}\}}{2\sum_{l=1}^{p}\{log(\hat{\lambda}_l + 1) - \hat{\lambda}\}} - 2C_n\frac{k(k+1)}{2p}. \tag{4.17}$$

Here, $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \ldots \geq \hat{\lambda}_p$ are the eigenvalues of the kernel matrix $\hat{M}$. Interested readers may refer to Zhu, Wang, Zhu and Ferre (2010) for details of $\hat{M}$. For both (4.16) and (4.17), the estimate structural dimension $\hat{K}$ is defined as the maximizer of $G(k)$ over $= 1, \ldots, p$.

Following Zhu, Zhu and Feng (2010), we use a modified BIC-type method for COCUM. Define

$$G(k) = n \sum_{i=1}^{k} \lambda_{ni}^2 / \sum_{i=1}^{p} \lambda_{ni}^2 - C_n k(k+1)/2 \tag{4.18}$$

where $\hat{\lambda}_{n1} \geq \hat{\lambda}_{n2} \geq \ldots \geq \hat{\lambda}_{np}$ are the sample eigenvalues of kernel matrix. In COCUM, the kernel matrix is

$$\mathbf{M} = \mathbb{E}[\mathbf{m}_c(\tilde{Y})\mathbf{m}_c^T(\tilde{Y})w(\tilde{Y})], \tag{4.19}$$

and $\mathbf{m}_c$ is defined in (4.7). And the estimated dimension $\hat{K}$ is defined as

$$\hat{K} = arg\ max_{1 \leq k \leq p} G(k). \tag{4.20}$$

The idea of the determination of the value of the penalty constant C_n is: if C_n is too small, then this modified BIC method tends to overestimate the dimension K; if C_n is too large, then this modified BIC method tends to underestimate the dimension K. A data-driven manner is needed to choose an appropriate value for C_n under a certain method. As pointed out in Zhu, Zhu and Feng (2010):

Theorem 4.9. *If $C_n/n \to 0$ as $n \to \infty$ and $C_n \to \infty$, then $\hat{K} - K = O(1)$.*

In COCUM, we let

$$C_n = 0.5log(n)$$

which mostly leads to satisfactory results and also satisfies the consistency result from the previous theorem also holds.

4.7. SIMULATION STUDIES

In this section, we compare the performance of COCUM with CUME. We considered several different models with the design matrix generated from normal, Cauchy and

Gamma distribution. To evaluate the performance of different methods, various criteria are used. In this paper, we use the ratio of square multiple correlation coefficient to the dimension d followed by Li and Dong (2009):

$$\frac{\rho^2}{d} = \frac{trace\{(\hat{\beta}^T\mathbf{\Sigma}\hat{\beta})^{-1}(\hat{\beta}^T\mathbf{\Sigma}\beta)(\beta^T\mathbf{\Sigma}\hat{\beta})(\beta^T\mathbf{\Sigma}\beta)^{-1}\}}{d} \tag{4.21}$$

as evaluation measurements to summarize simulation results from 1000 runs. Here $\hat{\beta}$ is the estimate for true β and $\mathbf{\Sigma}$ is the covariance matrix for predictor $\mathbf{X}$. The idea is the ρ^2 gets closer to the true dimension d if the sample version $\boldsymbol{\beta}^T\mathbf{X}$ and its true value have a linear relation and it is 0 if they are uncorrelated. Hence, the closer the ratio is to 1, the better fit of the model. In addition, the frequencies of estimated structural dimension over the 1000 trials are given as the measurements of the modified BIC performance. All numbers reported in the frequency table are multiplied by 10.

4.7.1. Model I. For a fair comparison, we first consider a regular linear model as CUME (Zhu, Zhu and Feng, 2010) discussed in their paper. The predictor $\mathbf{X}$ is generated from $N(0, I_p)$, ϵ is standard normal and independent of $\mathbf{X}$, the univariate response Y is constructed as:

$$Y = \mathbf{X}\boldsymbol{\beta} + 4\epsilon, \tag{4.22}$$

where $\boldsymbol{\beta} = (1, 1, 1, 1, 0, \ldots, 0)^T$. Following Zhu, Zhu and Feng (2010), we took all combinations of $n = 200, 400$ and 600, $p = 10, 15$ and 20. Table 4.1 provides the average correlation coefficient ratio (4.21) and the standard deviation for both CUME and COCUM respectively. As shown on Table 4.1, under the regular linear model with normally generated random predictors, the performances of CUME and COCUM are comparable in the sense that the average ratio and standard deviation are very close to each other. In addition, we observe that, with the increase of sample size, the performances of both CUME and COCUM improve, and deteriorate with the increase of p. The frequencies of structural dimension estimate $\hat{d}$ is provided in Table 4.2 which indicates the good performances of the use of the modified BIC method.

Table 4.1. COCUM: Model I

		$n = 200$		$n = 400$		$n = 600$	
	Method	ratio	deviation	ratio	deviation	ratio	deviation
$p = 10$	CUME	0.82	0.08	0.90	0.05	0.93	0.03
	COCUM	0.80	0.09	0.88	0.05	0.92	0.03
$p = 15$	CUME	0.74	0.10	0.85	0.05	0.90	0.04
	COCUM	0.72	0.10	0.83	0.06	0.89	0.04
$p = 20$	CUME	0.69	0.10	0.82	0.06	0.87	0.04
	COCUM	0.64	0.10	0.79	0.06	0.85	0.04

Table 4.2. COCUM: Model I Dimension Estimate

p	n	$d = 1$		$d > 1$	
		CUME	COCUM	CUME	COCUM
10	200	89.7	100	10.3	0
10	400	98.1	100	1.9	0
10	600	99.4	100	0.6	0
15	200	54.8	99.8	45.2	0.2
15	400	79.8	100	20.2	0
15	600	93.2	100	6.8	0
20	200	16.7	98.7	83.3	1.3
20	400	42.2	100	57.8	0
20	600	63.4	100	36.6	0

4.7.2. Model II. We now consider a more complicated model with two dimensions ($d = 2$). The predictor $\mathbf{X} = (X_1, X_2, X_3, \ldots, X_p)$ is still generated from $N(0, I_p)$, ϵ is standard normal and independent of $\mathbf{X}$, the univariate response Y is constructed as:

$$Y = 1.5(5 + X_1)(2 + X_2 + X_3) + 0.5\epsilon. \tag{4.23}$$

In this case, $\boldsymbol{\beta} = (\beta_1, \beta_2)$ where $\beta_1 = (1, 0, \ldots, 0)^T$ and $\beta_2 = (0, 1, 1, 0, \ldots, 0)^T$. The simulation results are shown on Table 4.3 and Table 4.4, we observe that the performance of COCUM is slightly better than CUME in this case. In addition, CUME has the serious problem of underestimating the structural dimension d.

4.7.3. Model III. Let's try another interesting model with two dimensions ($d = 2$) with the predictor $\mathbf{X} = (X_1, X_2, X_3, \ldots, X_p)$ generated from $N(0, I_p)$, and standard normal ϵ. The univariate response Y is constructed as:

$$Y = 4sin((0.25X_1 + 1)^2) + 0.5(X_2 + X_5 + 1)^2 + 0.2\epsilon. \tag{4.24}$$

In this case, $\boldsymbol{\beta} = (\beta_1, \beta_2)$ where $\beta_1 = (1, 0, \ldots, 0)^T$ and $\beta_2 = (0, 1, 0, 0, 1, 0, \ldots, 0)^T$. Simulation results are shown on Table 4.5 and Table 4.6. The modified BIC method perfectly works in this case.

Table 4.3. COCUM: Model II

		$n = 200$		$n = 400$		$n = 600$	
	Method	ratio	deviation	ratio	deviation	ratio	deviation
$p = 10$	CUME	0.71	0.22	0.80	0.18	0.85	0.14
	COCUM	0.75	0.19	0.84	0.15	0.88	0.11
$p = 15$	CUME	0.66	0.19	0.74	0.17	0.79	0.14
	COCUM	0.69	0.18	0.78	0.14	0.83	0.12
$p = 20$	CUME	0.63	0.16	0.71	0.16	0.76	0.14
	COCUM	0.64	0.16	0.74	0.14	0.79	0.12

Table 4.4. COCUM: Model II Dimension Estimate

p	n	$d = 1$		$d = 2$		$d > 2$	
		CUME	COCUM	CUME	COCUM	CUME	COCUM
10	200	100	1	0	98.5	0	0.5
10	400	100	0.3	0	99.7	0	0
10	600	100	0.2	0	99.8	0	0
15	200	100	0	0	82.9	0	17.1
15	400	100	0	0	98.1	0	1.9
15	600	100	0	0	99.3	0	0.7
20	400	100	0	0	74	0	26
20	600	100	0	0	91	0	9

Table 4.5. COCUM: Model III

		$n = 200$		$n = 400$		$n = 600$	
	Method	ratio	deviation	ratio	deviation	ratio	deviation
$p = 10$	CUME	0.90	0.08	0.95	0.04	0.96	0.02
	COCUM	0.89	0.09	0.94	0.04	0.96	0.03
$p = 15$	CUME	0.85	0.10	0.92	0.05	0.94	0.04
	COCUM	0.84	0.11	0.91	0.05	0.94	0.03
$p = 20$	CUME	0.93	0.04	0.95	0.03	0.93	0.04
	COCUM	0.92	0.04	0.94	0.03	0.92	0.04

Table 4.6. COCUM: Model III Dimension Estimate

p	n	$d = 1$		$d = 2$		$d > 2$	
		CUME	COCUM	CUME	COCUM	CUME	COCUM
10	200	28.4	0.1	71.6	99.9	0	0
10	400	3.3	0	96.7	100	0	0
10	600	0.2	0	99.8	100	0	0
15	200	6.1	0	93.9	99.7	0	0.3
15	400	0.1	0	99.9	100	0	0
15	600	0	0	100	100	0	0
20	200	0	0	100	100	0	0
20	400	0	0	100	100	0	0
20	600	0	0	100	99.9	0	0.1

4.7.4. Model IV. In the following model, we consider predictors generated from non-Gaussian distribution. Let's first work on a model with two dimensions ($d = 2$) with the predictor $\mathbf{X} = (X_1, X_2, X_3, \ldots, X_p)$ generated from $Cauchy(1)$, and standard normal ϵ. The univariate response Y is constructed as:

$$Y = 0.5(X_1 + 1)^2 + (0.5 + (X_2 + 1.5)^2) + 0.5\epsilon. \tag{4.25}$$

In this case, $\boldsymbol{\beta} = (\beta_1, \beta_2)$ where $\beta_1 = (1, 0, \ldots, 0)^T$ and $\beta_2 = (0, 1, 0, \ldots, 0)^T$. As the simulation results shown on Table 4.7 and Table 4.8, COCUM significantly outperforms CUME with over 60 percents higher correlation ratio than CUME for any combination of n and p. The mode of all $\hat{d}$ is always 2 for this model.

Table 4.7. COCUM: Model IV

		$n = 200$		$n = 400$		$n = 600$	
	Method	ratio	deviation	ratio	deviation	ratio	deviation
$p = 10$	CUME	0.29	0.23	0.29	0.23	0.30	0.23
	COCUM	0.86	0.33	0.91	0.29	0.93	0.27
$p = 15$	CUME	0.22	0.21	0.23	0.21	0.23	0.22
	COCUM	0.81	0.36	0.87	0.32	0.89	0.31
$p = 20$	CUME	0.17	0.19	0.17	0.19	0.18	0.20
	COCUM	0.76	0.38	0.85	0.35	0.86	0.33

Table 4.8. COCUM: Model IV Dimension Estimate

p	n	$d = 1$		$d = 2$		$d > 2$	
		CUME	COCUM	CUME	COCUM	CUME	COCUM
10	200	4	44.8	76.8	45.5	19.2	9.4
10	400	0.8	42.1	54	47.8	45.2	10.1
10	600	0.1	39.5	37.5	52.8	62.4	7.7
15	200	0.4	39	54.8	41.8	44.8	19.2
15	400	0.1	37.6	28.9	45	71	17.4
15	600	0	38	15.1	46.8	84.9	15.2
20	200	0	37	39.5	37.4	60.5	25.6
20	400	0	33.2	11	43.6	89	23.2
20	600	0	30.5	3.9	46	96.1	23.5

4.7.5. Model V. Here, we consider another model with two dimensions ($d = 2$) with the predictor $\mathbf{X} = (X_1, X_2, X_3, \ldots, X_p)$ generated from $Cauchy(1)$, and standard normal ϵ. The univariate response Y is constructed as:

$$Y = 0.5(X_3 + 1)^2 + \sqrt{0.5 + (X_2 - X_5 + 1.5)^2} + 0.5\epsilon. \tag{4.26}$$

In this case, $\boldsymbol{\beta} = (\beta_1, \beta_2)$ where $\beta_1 = (0, 0, 1, 0, \ldots, 0)^T$ and $\beta_2 = (0, 1, 0, 0, 1, 0, \ldots, 0)^T$. The simulation results on Table 4.9 and Table 4.10 also show that the performance of COCUM is better in this case. The mode of all $\hat{d}$ is always 2 for this model.

4.7.6. Model VI. In addition to Cauchy distribution generated random variables, we will also consider Gamma distribution random variables in the next two simulation models. Here, we let the predictor $\mathbf{X} = (X_1, X_2, X_3, \ldots, X_p)$ generated from $Gamma(0.1, 10)$, and standard normal ϵ. The univariate response Y is constructed as:

$$Y = (10 + (X_2 + 0.5)^2)\sqrt{(0.25 + X_1)} + 0.5\epsilon. \tag{4.27}$$

In this case, $\boldsymbol{\beta} = (\beta_1, \beta_2)$ where $\beta_1 = (1, 0, \ldots, 0)^T$ and $\beta_2 = (0, 1, 0, \ldots, 0)^T$. Table 4.11 and Table 4.12 demonstrate the good performances of COCUM. In addition, CUME has the serious problem of underestimating the structural dimension d.

Table 4.9. COCUM: Model V

		$n = 200$		$n = 400$		$n = 600$	
	Method	ratio	deviation	ratio	deviation	ratio	deviation
$p = 10$	CUME	0.45	0.26	0.47	0.27	0.48	0.26
	COCUM	0.81	0.35	0.81	0.36	0.85	0.34
$p = 15$	CUME	0.38	0.24	0.40	0.25	0.41	0.24
	COCUM	0.78	0.37	0.82	0.35	0.82	0.35
$p = 20$	CUME	0.33	0.22	0.35	0.23	0.36	0.23
	COCUM	0.75	0.38	0.80	0.36	0.81	0.36

Table 4.10. COCUM: Model V Dimension Estimate

p	n	$d = 1$		$d = 2$		$d > 2$	
		CUME	COCUM	CUME	COCUM	CUME	COCUM
10	200	32.2	25	66.1	55.2	1.7	19.8
10	400	26	23.9	68.3	52.3	5.7	23.8
10	600	23	25.1	66.9	51.3	10.1	23.6
15	200	6.2	20.6	84.4	44.8	9.4	34.6
15	400	4.5	20.3	71.7	49.2	23.8	30.5
15	600	3.8	20.1	63.5	45.6	32.7	34.3
20	200	0.4	18.1	77	42.1	22.6	39.8
20	400	0.5	18.8	54.5	40.9	45	40.3
20	600	0.5	21.9	39	39.2	60.5	38.9

Table 4.11. COCUM: Model VI

		$n = 200$		$n = 400$		$n = 600$	
	Method	ratio	deviation	ratio	deviation	ratio	deviation
$p = 10$	CUME	0.52	0.16	0.57	0.25	0.63	0.32
	COCUM	0.93	0.15	0.96	0.08	0.98	0.04
$p = 15$	CUME	0.49	0.08	0.51	0.11	0.54	0.19
	COCUM	0.89	0.20	0.94	0.10	0.96	0.07
$p = 20$	CUME	0.48	0.04	0.49	0.07	0.51	0.10
	COCUM	0.85	0.22	0.92	0.14	0.95	0.08

Table 4.12. COCUM: Model VI Dimension Estimate

p	n	$d=1$		$d=2$		$d>2$	
		CUME	COCUM	CUME	COCUM	CUME	COCUM
10	200	99.5	8.3	0.5	91	0	0.7
10	400	100	1.8	0	97.2	0	1
10	600	100	0.3	0	99	0	0.7
15	200	90	3.6	10	90.1	0	6.3
15	400	99.2	0.8	0.8	95.4	0	3.8
15	600	100	0.1	0	95.9	0	4
20	200	50.6	1.4	49.4	75.7	0	22.9
20	400	83.5	0.3	16.5	87.6	0	12.1
20	600	95.9	0.1	4.1	94.3	0	5.6

4.7.7. Model VII. This is another model with Gamma distribution random variables and $d = 2$. Here, we let the predictor $\mathbf{X} = (X_1, X_2, X_3, \ldots, X_p)$ generated from $Gamma(0.1, 10)$, and standard normal ϵ. The univariate response Y is constructed as:

$$Y = (X_1 + 0.25) + 0.9log(0.25 + (0.5 + X_2)^2) + 0.5\epsilon. \tag{4.28}$$

In this case, $\boldsymbol{\beta} = (\beta_1, \beta_2)$ where $\beta_1 = (1, 0, \ldots, 0)^T$ and $\beta_2 = (0, 1, 0, \ldots, 0)^T$. The simulation results on Table 4.13 and Table 4.14 lead us to draw the same conclusion as on the previous model. Also, CUME has the serious problem of underestimating the structural dimension d.

Table 4.13. COCUM: Model VII

		$n = 200$		$n = 400$		$n = 600$	
	Method	ratio	deviation	ratio	deviation	ratio	deviation
$p = 10$	CUME	0.56	0.22	0.67	0.30	0.75	0.31
	COCUM	0.89	0.19	0.95	0.08	0.97	0.04
$p = 15$	CUME	0.51	0.12	0.57	0.21	0.65	0.27
	COCUM	0.84	0.21	0.92	0.11	0.94	0.06
$p = 20$	CUME	0.49	0.07	0.53	0.14	0.58	0.20
	COCUM	0.79	0.24	0.88	0.16	0.92	0.08

Table 4.14. COCUM: Model VII Dimension Estimate

p	n	$d = 1$		$d = 2$		$d > 2$	
		CUME	COCUM	CUME	COCUM	CUME	COCUM
10	200	99.8	19.1	0.2	80.9	0	0
10	400	100	7.3	0	92.7	0	0
10	600	100	2.6	0	97.4	0	0
15	200	89.2	5.1	10.8	94.5	0	0.4
15	400	98.7	1.4	1.	98.4	0	0.2
15	600	99.7	1.3	0.3	98.7	0	0
20	200	46.2	0.8	53.8	93.9	0	5.3
20	400	83	0.2	17	96.8	0	3
20	600	94.4	0.1	5.6	99.4	0	0.5

4.8. CONCLUSION

In this chapter, we proposed a new method called Covariance Cumulative Slicing Estimation (COCUM). Compared with most slicing methods, COCUM does not only recover the loss information caused by replacing the continuous predictors Y by a discrete version of Y, but also minimizes the variation results leading by choosing different numbers of slices h. Most importantly, COCUM considers the specific value of y in the kernel matrix. This means, COCUM is more robust to the outliers. The simulation results show that COCUM is comparable to CUME when the predictors are normally distributed. But COCUM outperforms CUME when the predictors do not follow the Gaussian distribution, such as Cauchy or Gamma distribution. Figure 4.1 shows the graph of the response Y from Model I to Model III and Figure 4.2 shows the graph of y from Model IV to Model VII. For the graphs, the x-axis represents the order of the data and the y-axis stands for the value of y. For brevity, we only use the case with $n = 800$ and $p = 10$. We can see that in the second figure, the response values vary significantly in a wide range; while the Y values on the first figure is more stable. Our simulation results indicate the advantages of incorporating the values of Y into the kernel matrix. Associated asymptotic results are also proven.

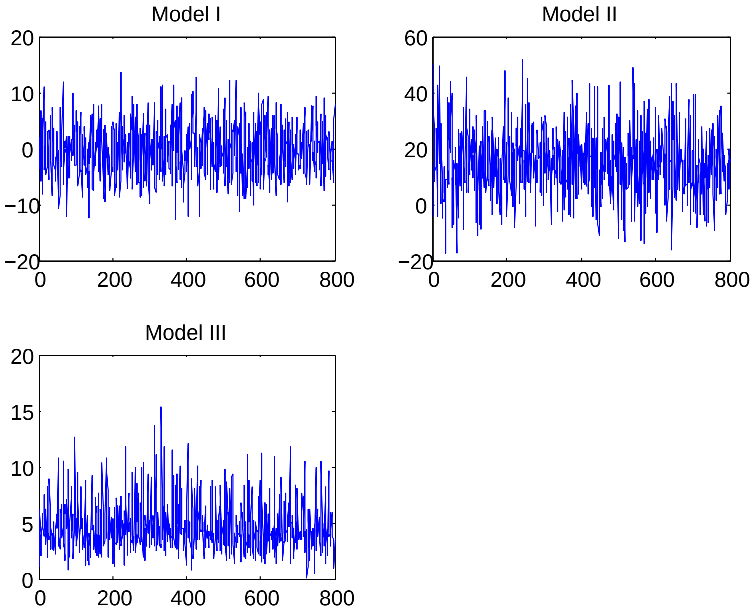


Figure 4.1. The graph of y: Model I - Model III

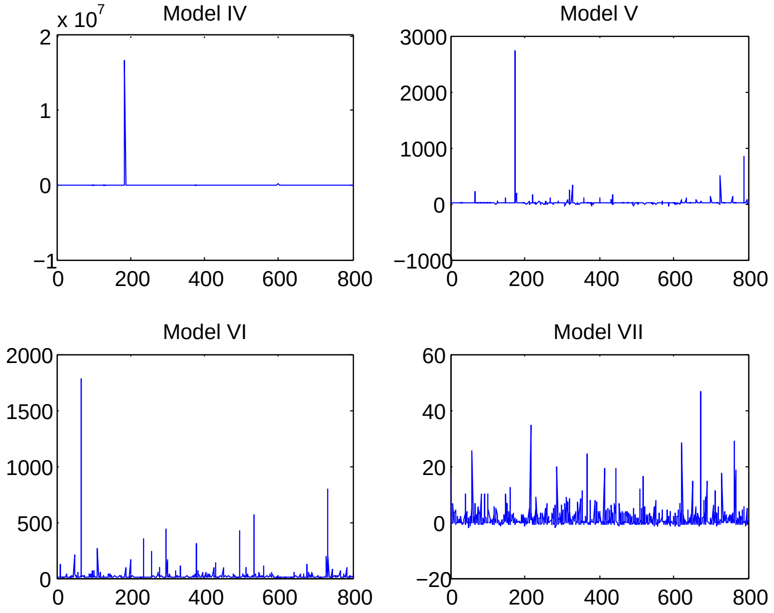


Figure 4.2. The graph of y: Model IV - Model VII

APPENDIX A

MATLAB ALGORITHM: sgSDR

```
%--------The following is the Matlab code for sgSDR--------------%
%----------------------------------------------------------------%
w_DSGL_CV10_output=[];
w_SGL_CV10_output=[];
w_DSGL_BIC0_output=[];
w_SGL_BIC0_output=[];
z_DSGL_CV10_output=[];
z_SGL_CV10_output=[];
z_DSGL_BIC0_output=[];
z_SGL_BIC0_output=[];
TPR_DSGL_CV10_output=[];
TPR_SGL_CV10_output=[];
TPR_DSGL_BIC0_output=[];
TPR_SGL_BIC0_output=[];
FPR_DSGL_CV10_output=[];
FPR_SGL_CV10_output=[];
FPR_DSGL_BIC0_output=[];
FPR_SGL_BIC0_output=[];
TPR_DSGL_CV10_GRP_output=[];
TPR_SGL_CV10_GRP_output=[];
TPR_DSGL_BIC0_GRP_output=[];
TPR_SGL_BIC0_GRP_output=[];
FPR_DSGL_CV10_GRP_output=[];
FPR_SGL_CV10_GRP_output=[];
FPR_DSGL_BIC0_GRP_output=[];
FPR_SGL_BIC0_GRP_output=[];
ngrp=10; % the # of groups
grpsize=200; % the group size
beta_true_grp=[1 1 0 0 0 0 0 0 0 0];  % true group info
for simu_loop=1:10
```

```
m=100;n=2000; % m is sample size %n is the dimension of beta
%randNum=10;
% --------------------generate random data--------------------%
A=randn(m,n);        % the data matrix
%noise=3.*randn(m,1);
noise=trnd(1,m,1);
%first group
A1=A(:,1);
A2=A(:,2);
A3=(2/3).*A1+(2/3).*A2+(1/3).*(randn(m,1));
A(:,3)=A3;
%second group
A201=A(:,201);
A202=A(:,202);
A203=(2/3).*A201+(2/3).*A202+(1/3).*(randn(m,1));
A(:,203)=A203;
xOrin=zeros(n,1); % true beta
xOrin(1:2,1)=[-2 3];
xOrin(201:202,1)=[-2 3];
%xOrin(401:402,1)=[-2 3];
y=exp(A*xOrin+noise);
%centering predictors
for k=1:n
    A(:,k)=A(:,k)-mean(A(:,k));
end
sample_size = size(A, 1);
[f x]=ecdf(y);
newf=f(2:end, 1);
newx=x(2:end, 1);
nsize=size(y,1);
```

```
F=[];
for k=1:nsize
c=y(k);
index=find(newx==c);
F=[F newf(index)];
end
F=F';
%centering F
F=F-mean(F);
%y=A*xOrin +...
    %noise*0.01;      % the response
%centering y
y=y-mean(y);
%----------------------- Set optional items -----------------------%
opts=[];
% Starting point
opts.init=2;        % starting from a zero point
% Termination
opts.tFlag=5;       % run .maxIter iterations
opts.maxIter=200;   % maximum number of iterations
% regularization
opts.rFlag=0;       % use input
% Normalization
opts.nFlag=0;       % without normalization
%opts.nFlag=1;       %with normalization
% Group Property (group 1)
opts.ind=[ [1, 200, sqrt(200)]', [201, 400, sqrt(200)]',...
    [401, 600, sqrt(200)]', [601, 800, sqrt(200)]', ...
[801, 1000, sqrt(200)]', [1001, 1200, sqrt(200)]',...
[1201, 1400, sqrt(200)]', [1401, 1600, sqrt(200)]',...
[1601, 1800, sqrt(200)]', [1801,2000, sqrt(200)]'];
%-----------------sgSDR Cross Validation 10 process---------------%
param1_range = [0.001 0.003 0.005 0.007 0.009 0.01 0.03 0.05 0.07 ...
    30 50 70 90 100 200 300 400 500];
param2_range = [0.001 0.003 0.005 0.007 0.009 0.01 0.03 0.05 0.07 ...
    30 50 70 90 100 200 300 400 500];
cv_fold_num = 5; % by default use 10-fold cross validation.w
lldiff=[];
cv_performance=zeros(length(param1_range),length(param2_range));
for cv_idx = 1: cv_fold_num
    te_index = cv_idx : cv_fold_num : sample_size;
    tr_index = setdiff( 1 : sample_size, te_index );
    cv_X_tr = A(tr_index, :);
    cv_X_te = A(te_index, :);
    cv_Y_tr = F(tr_index);
    cv_Y_te = F(te_index);
    cv_W = [];
    cv_C = [];
    lldiff=[];
    for rho1_idx = 1:length(param1_range)
        rho_1 = param1_range(rho1_idx);
        for rho2_idx = 1:length(param2_range)
            rho_2 = param2_range(rho2_idx);
            z=[rho_1,rho_2];
            [cv_w, cv_c,ValueL] = sgLeastR(cv_X_tr, cv_Y_tr,z,opts);
            cv_W = [cv_W cv_w];
            cv_C = [cv_C cv_c];
            diff=sum((cv_Y_te-cv_X_te*cv_w).*(cv_Y_te-cv_X_te*cv_w));
            lldiff=[lldiff diff];
        end
    end
   metric_arr=reshape(lldiff, length(param2_range),length(param1_range));
   metric_arrt=metric_arr’;
   cv_performance=cv_performance+metric_arrt;
end
[i j] = find(cv_performance == min(cv_performance(:)));
rho1_idx=i(1,1);
rho2_idx=j(1,1);
param1 = param1_range(rho1_idx);
param2 = param2_range(rho2_idx);
% use the selected
z=[param1,param2];
[w_DSGL_CV10,c1,ValueL] = sgLeastR(A, F, z,opts); %DSGL
% lamda 1 and lamda 2
z=z’;
z_DSGL_CV10_output=[z_DSGL_CV10_output z];
% estimate of beta from DSGL
w_DSGL_CV10_output=[w_DSGL_CV10_output w_DSGL_CV10];
%group info computation
w_DSGL_CV10_grp=[];        %used to create est. group info
for k=1:ngrp
    lgrp=1+grpsize*(k-1);
    ugrp=grpsize+grpsize*(k-1);
    if all(w_DSGL_CV10(lgrp:ugrp)==0)
        w_DSGL_CV10_grp(k)=0;
    else
        w_DSGL_CV10_grp(k)=1;
    end
end
%group evaluation
```

```
BETA=beta_true_grp~=0; %Ori_nonzero
beta1=w_DSGL_CV10_grp~=0; %Est_nonzero
TP_DSGL_CV10_grp=sum( ( (BETA==1) + (beta1==1) )==2 );
FP_DSGL_CV10_grp=sum( ( (BETA==0) + (beta1==1) )==2 );
FN_DSGL_CV10_grp=sum( ( (BETA==1) + (beta1==0) )==2 );
TN_DSGL_CV10_grp=sum( ( (BETA==0) + (beta1==0) )==2 );
TPR_DSGL_CV10_grp=TP_DSGL_CV10_grp/(TP_DSGL_CV10_grp+FN_DSGL_CV10_grp);
FPR_DSGL_CV10_grp=FP_DSGL_CV10_grp/(FP_DSGL_CV10_grp+TN_DSGL_CV10_grp);
% --------------------Sparse Group Lasso Cross Validation-----------------%
param1_range = [0.0005 0.001 0.003 0.005 0.007 0.009 0.01 0.03 0.05 0.07 ...
    30 50 70 90 100 200 300 400 500];
param2_range = [0.0005 0.001 0.003 0.005 0.007 0.009 0.01 0.03 0.05 0.07 ...
    30 50 70 90 100 200 300 400 500];
cv_fold_num = 5; % by default use 10-fold cross validation.w
lldiff=[];
cv_performance=zeros(length(param1_range),length(param2_range));
for cv_idx = 1: cv_fold_num
    te_index = cv_idx : cv_fold_num : sample_size;
    tr_index = setdiff( 1 : sample_size, te_index );
    cv_X_tr = A(tr_index, :);
    cv_X_te = A(te_index, :);
    cv_Y_tr = y(tr_index);
    cv_Y_te = y(te_index);
    cv_W = [];
    cv_C = [];
    lldiff=[];
    for rho1_idx = 1:length(param1_range)
        rho_1 = param1_range(rho1_idx);
        for rho2_idx = 1:length(param2_range)
            rho_2 = param2_range(rho2_idx);
            z_s=[rho_1,rho_2];
            [cv_w, cv_c,ValueL] = sgLeastR(cv_X_tr, cv_Y_tr,z_s,opts);
            cv_W = [cv_W cv_w];
            cv_C = [cv_C cv_c];
            diff=sum((cv_Y_te-cv_X_te*cv_w).*(cv_Y_te-cv_X_te*cv_w));
            lldiff=[lldiff diff];
        end
    end
   metric_arr=reshape(lldiff, length(param2_range),length(param1_range));
   metric_arrt=metric_arr’;
   cv_performance=cv_performance+metric_arrt;
end
[isgl jsgl] = find(cv_performance == min(cv_performance(:)));
rho1_idx_sgl=isgl(1,1);
rho2_idx_sgl=jsgl(1,1);
param1_sgl = param1_range(rho1_idx_sgl);
param2_sgl = param2_range(rho2_idx_sgl);
z_sgl=[param1_sgl,param2_sgl];
[w_SGL_CV10,c2,ValueL]=sgLeastR(A, y, z_sgl, opts); %SGL
% lamda 1 and lamda 2
z_sgl=z_sgl’;
z_SGL_CV10_output=[z_SGL_CV10_output z_sgl];
% estimate of beta from DSGL
w_SGL_CV10_output=[w_SGL_CV10_output w_SGL_CV10]; %SGL
%group info computation
w_SGL_CV10_grp=[];        %used to create est. group info
for k=1:ngrp
    lgrp=1+grpsize*(k-1);
    ugrp=grpsize+grpsize*(k-1);
    if all(w_SGL_CV10(lgrp:ugrp)==0)
```

```
        w_SGL_CV10_grp(k)=0;
    else
        w_SGL_CV10_grp(k)=1;
    end
end
%group evaluation
BETA=beta_true_grp˜=0; %Ori_nonzero
beta2=w_SGL_CV10_grp˜=0; %Est_nonzero
TP_SGL_CV10_grp=sum( ( (BETA==1) + (beta2==1) )==2 );
FP_SGL_CV10_grp=sum( ( (BETA==0) + (beta2==1) )==2 );
FN_SGL_CV10_grp=sum( ( (BETA==1) + (beta2==0) )==2 );
TN_SGL_CV10_grp=sum( ( (BETA==0) + (beta2==0) )==2 );
TPR_SGL_CV10_grp=TP_SGL_CV10_grp/(TP_SGL_CV10_grp+FN_SGL_CV10_grp);
FPR_SGL_CV10_grp=FP_SGL_CV10_grp/(FP_SGL_CV10_grp+TN_SGL_CV10_grp);
%----------------------------sgSDR BIC Process---------------------%
param1_range = [0.001 0.003 0.005 0.007 0.009 0.01 0.03 0.05 0.07 ...
    30 50 70 90 100 200 300 400 500];
param2_range = [0.001 0.003 0.005 0.007 0.009 0.01 0.03 0.05 0.07 ...
30 50 70 90 100 200 300 400 500];
BIC=[];
gama=0;  % need to try 0, 0.5 and 1
czero=0.1;
BIC_W=[];
BIC_C=[];
    for rho1_idx = 1:length(param1_range)
        rho_1 = param1_range(rho1_idx);
        for rho2_idx = 1:length(param2_range)
            rho_2 = param2_range(rho2_idx);
            z=[rho_1,rho_2];
            [BIC_w, BIC_c,ValueL] = sgLeastR(A,F,z,opts);
            BIC_W = [BIC_W BIC_w];
            BIC_C = [BIC_C BIC_c];
            RSS_lamda=sum((F-A*BIC_w).*(F-A*BIC_w));
            c=czero*var(F);
            M_size=BIC_w~=0;
            M_lamda=sum(M_size);
            difference=n-M_lamda;
            combination=nchoosek(n,M_lamda);
            bic=m*log(RSS_lamda/m+c)+log(m)*M_lamda+2*gama*combination;
            BIC=[BIC bic];
        end
    end
BIC_performance=reshape(BIC, length(param2_range),length(param1_range));
BIC_performance=BIC_performance';
[i j] = find(BIC_performance == min(BIC_performance(:)));
rho1_idx=i(1,1);
rho2_idx=j(1,1);
param1 = param1_range(rho1_idx);
param2 = param2_range(rho2_idx);
z=[param1,param2];
[w_DSGL_BIC0,c1,ValueL] = sgLeastR(A, F, z,opts); %DSGL
% lamda 1 and lamda 2
z=z';
z_DSGL_BIC0_output=[z_DSGL_BIC0_output z];
% estimate of beta from DSGL
w_DSGL_BIC0_output=[w_DSGL_BIC0_output w_DSGL_BIC0];
%group info computation
w_DSGL_BIC0_grp=[];        %used to create est. group info
for k=1:ngrp
    lgrp=1+grpsize*(k-1);
    ugrp=grpsize+grpsize*(k-1);
    if all(w_DSGL_BIC0(lgrp:ugrp)==0)
       w_DSGL_BIC0_grp(k)=0;
    else
        w_DSGL_BIC0_grp(k)=1;
    end
end
%evaluation
BETA=beta_true_grp~=0; %Ori_nonzero
beta3=w_DSGL_BIC0_grp~=0; %Est_nonzero
TP_DSGL_BIC0_grp=sum( ( (BETA==1) + (beta3==1) )==2 );
FP_DSGL_BIC0_grp=sum( ( (BETA==0) + (beta3==1) )==2 );
FN_DSGL_BIC0_grp=sum( ( (BETA==1) + (beta3==0) )==2 );
TN_DSGL_BIC0_grp=sum( ( (BETA==0) + (beta3==0) )==2 );
TPR_DSGL_BIC0_grp=TP_DSGL_BIC0_grp/(TP_DSGL_BIC0_grp+FN_DSGL_BIC0_grp);
FPR_DSGL_BIC0_grp=FP_DSGL_BIC0_grp/(FP_DSGL_BIC0_grp+TN_DSGL_BIC0_grp);
%--------------Sparse Group Lasso BIC Process---------------------%
param1_range = [0.0005 0.001 0.003 0.005 0.007 0.009 0.01 0.03 0.05 0.07 ...
    30 50 70 90 100 200 300 400 500];
param2_range = [0.0005 0.001 0.003 0.005 0.007 0.009 0.01 0.03 0.05 0.07 ...
    30 50 70 90 100 200 300 400 500];
BIC=[];
gama=0;  % need to try 0, 0.5 and 1
czero=0.1;
BIC_W=[];
BIC_C=[];
    for rho1_idx = 1:length(param1_range)
        rho_1 = param1_range(rho1_idx);
        for rho2_idx = 1:length(param2_range)
            rho_2 = param2_range(rho2_idx);
            z=[rho_1,rho_2];
            [BIC_w, BIC_c,ValueL] = sgLeastR(A,y,z,opts);
            BIC_W = [BIC_W BIC_w];
            BIC_C = [BIC_C BIC_c];
            RSS_lamda=sum((y-A*BIC_w).*(y-A*BIC_w));
            c=czero*var(y);
            M_size=BIC_w~=0;
            M_lamda=sum(M_size);
            difference=n-M_lamda;
            combination=nchoosek(n,M_lamda);
            bic=m*log(RSS_lamda/m+c)+log(m)*M_lamda+2*gama*combination;
            BIC=[BIC bic];
        end
    end
BIC_performance=reshape(BIC, length(param2_range),length(param1_range));
BIC_performance=BIC_performance’;
[i j] = find(BIC_performance == min(BIC_performance(:)));
rho1_idx=i(1,1);
rho2_idx=j(1,1);
param1 = param1_range(rho1_idx);
param2 = param2_range(rho2_idx);
z=[param1,param2];
[w_SGL_BIC0,c1,ValueL] = sgLeastR(A, y, z,opts); %DSGL
% lamda 1 and lamda 2
z=z’;
z_SGL_BIC0_output=[z_SGL_BIC0_output z];
% estimate of beta from DSGL
w_SGL_BIC0_output=[w_SGL_BIC0_output w_SGL_BIC0];
%group info computation
w_SGL_BIC0_grp=[];        %used to create est. group info
```

```
for k=1:ngrp
    lgrp=1+grpsize*(k-1);
    ugrp=grpsize+grpsize*(k-1);
    if all(w_SGL_BIC0(lgrp:ugrp)==0)
       w_SGL_BIC0_grp(k)=0;
    else
        w_SGL_BIC0_grp(k)=1;
    end
end
%evaluation
BETA=beta_true_grp~=0; %Ori_nonzero
beta4=w_SGL_BIC0_grp~=0; %Est_nonzero
TP_SGL_BIC0_grp=sum( ( (BETA==1) + (beta4==1) )==2 );
FP_SGL_BIC0_grp=sum( ( (BETA==0) + (beta4==1) )==2 );
FN_SGL_BIC0_grp=sum( ( (BETA==1) + (beta4==0) )==2 );
TN_SGL_BIC0_grp=sum( ( (BETA==0) + (beta4==0) )==2 );
TPR_SGL_BIC0_grp=TP_SGL_BIC0_grp/(TP_SGL_BIC0_grp+FN_SGL_BIC0_grp);
FPR_SGL_BIC0_grp=FP_SGL_BIC0_grp/(FP_SGL_BIC0_grp+TN_SGL_BIC0_grp);
% -------------------outputs computation-------------------------%
% evaluation for DSGL_CV10
%TPR=true positive rate=true declared positive/true positive=TDP/TP
%FPR=false positive  rate = declared false positives/true negative
%FDR = false discovery rate = declared false positive/ declared positive
Y=xOrin~=0; %Ori_nonzero
T1=w_DSGL_CV10~=0; %Est_nonzero
TP_DSGL_CV10=sum( ( (Y==1) + (T1==1) )==2 );
FP_DSGL_CV10=sum( ( (Y==0) + (T1==1) )==2 );
FN_DSGL_CV10=sum( ( (Y==1) + (T1==0) )==2 );
TN_DSGL_CV10=sum( ( (Y==0) + (T1==0) )==2 );
TPR_DSGL_CV10=TP_DSGL_CV10/(TP_DSGL_CV10+FN_DSGL_CV10);
```

```
FPR_DSGL_CV10=FP_DSGL_CV10/(FP_DSGL_CV10+TN_DSGL_CV10);
% evaluation for SGL_CV10
Y=xOrin~=0; %Ori_nonzero
T2=w_SGL_CV10~=0; %Est_nonzero
TP_SGL_CV10=sum( ( (Y==1) + (T2==1) )==2 );
FP_SGL_CV10=sum( ( (Y==0) + (T2==1) )==2 );
FN_SGL_CV10=sum( ( (Y==1) + (T2==0) )==2 );
TN_SGL_CV10=sum( ( (Y==0) + (T2==0) )==2 );
TPR_SGL_CV10=TP_SGL_CV10/(TP_SGL_CV10+FN_SGL_CV10);
FPR_SGL_CV10=FP_SGL_CV10/(FP_SGL_CV10+TN_SGL_CV10);
%evaluation for DSGL_BIC0
Y=xOrin~=0; %Ori_nonzero
T3=w_DSGL_BIC0~=0; %Est_nonzero
TP_DSGL_BIC0=sum( ( (Y==1) + (T3==1) )==2 );
FP_DSGL_BIC0=sum( ( (Y==0) + (T3==1) )==2 );
FN_DSGL_BIC0=sum( ( (Y==1) + (T3==0) )==2 );
TN_DSGL_BIC0=sum( ( (Y==0) + (T3==0) )==2 );
TPR_DSGL_BIC0=TP_DSGL_BIC0/(TP_DSGL_BIC0+FN_DSGL_BIC0);
FPR_DSGL_BIC0=FP_DSGL_BIC0/(FP_DSGL_BIC0+TN_DSGL_BIC0);
%evaluation for SGL_BIC0
Y=xOrin~=0; %Ori_nonzero
T4=w_SGL_BIC0~=0; %Est_nonzero
TP_SGL_BIC0=sum( ( (Y==1) + (T4==1) )==2 );
FP_SGL_BIC0=sum( ( (Y==0) + (T4==1) )==2 );
FN_SGL_BIC0=sum( ( (Y==1) + (T4==0) )==2 );
TN_SGL_BIC0=sum( ( (Y==0) + (T4==0) )==2 );
TPR_SGL_BIC0=TP_SGL_BIC0/(TP_SGL_BIC0+FN_SGL_BIC0);
FPR_SGL_BIC0=FP_SGL_BIC0/(FP_SGL_BIC0+TN_SGL_BIC0);
%collecting all the outputs
TPR_DSGL_CV10_output=[TPR_DSGL_CV10_output TPR_DSGL_CV10];
```

```
TPR_SGL_CV10_output=[TPR_SGL_CV10_output TPR_SGL_CV10];
TPR_DSGL_BIC0_output=[TPR_DSGL_BIC0_output TPR_DSGL_BIC0];
TPR_SGL_BIC0_output=[TPR_SGL_BIC0_output TPR_SGL_BIC0];
FPR_DSGL_CV10_output=[FPR_DSGL_CV10_output FPR_DSGL_CV10];
FPR_SGL_CV10_output=[FPR_SGL_CV10_output FPR_SGL_CV10];
FPR_DSGL_BIC0_output=[FPR_DSGL_BIC0_output FPR_DSGL_BIC0];
FPR_SGL_BIC0_output=[FPR_SGL_BIC0_output FPR_SGL_BIC0];
%colletcing group info output
TPR_DSGL_CV10_GRP_output=[TPR_DSGL_CV10_GRP_output TPR_DSGL_CV10_grp];
TPR_SGL_CV10_GRP_output=[TPR_SGL_CV10_GRP_output TPR_SGL_CV10_grp];
TPR_DSGL_BIC0_GRP_output=[TPR_DSGL_BIC0_GRP_output TPR_DSGL_BIC0_grp];
TPR_SGL_BIC0_GRP_output=[TPR_SGL_BIC0_GRP_output TPR_SGL_BIC0_grp];
FPR_DSGL_CV10_GRP_output=[FPR_DSGL_CV10_GRP_output FPR_DSGL_CV10_grp];
FPR_SGL_CV10_GRP_output=[FPR_SGL_CV10_GRP_output FPR_SGL_CV10_grp];
FPR_DSGL_BIC0_GRP_output=[FPR_DSGL_BIC0_GRP_output FPR_DSGL_BIC0_grp];
FPR_SGL_BIC0_GRP_output=[FPR_SGL_BIC0_GRP_output FPR_SGL_BIC0_grp];
end
%------------------mean-------------------------%
TPR_DSGL_CV10_meanavg=mean(TPR_DSGL_CV10_output)
TPR_SGL_CV10_meanavg=mean(TPR_SGL_CV10_output)
TPR_DSGL_BIC0_meanavg=mean(TPR_DSGL_BIC0_output)
TPR_SGL_BIC0_meanavg=mean(TPR_SGL_BIC0_output)

FPR_DSGL_CV10_meanavg=mean(FPR_DSGL_CV10_output)
FPR_SGL_CV10_meanavg=mean(FPR_SGL_CV10_output)
FPR_DSGL_BIC0_meanavg=mean(FPR_DSGL_BIC0_output)
FPR_SGL_BIC0_meanavg=mean(FPR_SGL_BIC0_output)
%------------------group mean----------------------%
TPR_DSGL_CV10_GRP_meanavg=mean(TPR_DSGL_CV10_GRP_output)
TPR_SGL_CV10_GRP_meanavg=mean(TPR_SGL_CV10_GRP_output)
```

```
TPR_DSGL_BIC0_GRP_meanavg=mean(TPR_DSGL_BIC0_GRP_output)

TPR_SGL_BIC0_GRP_meanavg=mean(TPR_SGL_BIC0_GRP_output)

FPR_DSGL_CV10_GRP_meanavg=mean(FPR_DSGL_CV10_GRP_output)

FPR_SGL_CV10_GRP_meanavg=mean(FPR_SGL_CV10_GRP_output)

FPR_DSGL_BIC0_GRP_meanavg=mean(FPR_DSGL_BIC0_GRP_output)

FPR_SGL_BIC0_GRP_meanavg=mean(FPR_SGL_BIC0_GRP_output)

\clearpage
```

APPENDIX B

MATLAB ALGORITHM: COCUM

```
%----------------The following code is for COCUM-----------------%
%----------------------------------------------------------------%
%-------------------generate random variables--------------------%
all_r_CUME=[];
all_r_COCUM=[];
all_dis_CUME=[];
all_dis_COCUM=[];
all_dim_est_CUME=[];
all_dim_est_COCUM=[];
for simu_loop=1:1000
n=400;  %sample size
p=10;   %the # of predictors
dim=2;
x=randg(2,n,p);
noise=(0.5).*randn(n,1);
truebeta=[1 0 0 0 0 0 0 0 0 0 ; 0 1 0 0 0 0 0 0 0 0 ]'; %true beta
x1=x(:,1);
x2=x(:,2);
x3=x(:,3);
y=0.5.*(x1+1).*(x1+1)+(0.5+(x2+15).*(x2+15))+noise;
%---------------------CUME Computation---------------------%
q = size(y,2);[n,p] = size(x);
Tmp = (inv(cov(x,1)))^(1/2);
z = (x - ones(n,1) * mean(x)) * Tmp;
LAMBDA = zeros(p);
    for ii = 1:q
        [a, pos] = sort(y(:,ii));
        zz = z(pos, :);
        muz = cumsum(zz)/n;
        LAMBDA = LAMBDA + muz' * muz;
    end

[U,S,V] = svd(LAMBDA/n);
Basis_CUME = Tmp*U(:,1:dim);
Sigval_CUME = diag(S);
%----------------------COCUM Computation---------------------------%
yy = y(pos);
zzyy = zeros(n,p);
for i = 1: n
  zzyy(i,:) = zz(i,:)*yy(i);
end
muzy = cumsum(zzyy)/n;
LAMBDA = muzy' * muzy;
[U,S,V] = svd(LAMBDA/n);
Basis_COCUM = Tmp*U(:,1:dim);
Sigval_COCUM = diag(S);
%----------------Evaluation of the performances------------------%
P_truebeta=truebeta*inv((truebeta'*truebeta))*truebeta';
P_estbeta_CUME=Basis_CUME*inv((Basis_CUME'*Basis_CUME))
*Basis_CUME';
P_estbeta_COCUM=Basis_COCUM*inv((Basis_COCUM'*Basis_COCUM))
*Basis_COCUM';
r_CUME=trace(P_truebeta*P_estbeta_CUME)/dim;
r_COCUM=trace(P_truebeta*P_estbeta_COCUM)/dim;
dis_CUME=trace(inv(Basis_CUME'*cov(x,1)*Basis_CUME)
*(Basis_CUME'*cov(x,1)*truebeta)
*inv(truebeta'*cov(x,1)*truebeta)
*(truebeta'*cov(x,1)*Basis_CUME));
dis_COCUM=trace(inv(Basis_COCUM'*cov(x,1)*Basis_COCUM)
*(Basis_COCUM'*cov(x,1)
*truebeta)*inv(truebeta'*cov(x,1)*truebeta)
*(truebeta'*cov(x,1)*Basis_COCUM));
all_r_CUME=[all_r_CUME r_CUME];
all_r_COCUM=[all_r_COCUM r_COCUM];
all_dis_CUME=[all_dis_CUME dis_CUME];
all_dis_COCUM=[all_dis_COCUM dis_COCUM];
%------------------the determination of dimension d------------------%
C=10*0.05*log(n);
allG_CUME=[];
allG_COCUM=[];
Slambda_CUME=Sigval_CUME.*Sigval_CUME;
Dif_CUME=log(Sigval_CUME+1)-Sigval_CUME;
for k=1:p
    %G_CUME=n*sum(Slambda_CUME(1:k))
    /sum(Slambda_CUME)-(C*k*(k+1)/p);
    G_CUME=n*sum(Dif_CUME(1:k))
    /(2*sum(Dif_CUME))-(C*k*(k+1)/p);
    allG_CUME=[allG_CUME G_CUME];
end
dim_est_CUME=find((allG_CUME) == max(allG_CUME(:)));
Slambda_COCUM=Sigval_COCUM.*Sigval_COCUM;
Dif_COCUM=log(Sigval_COCUM+1)-Sigval_COCUM;
for k=1:p
    %G_COCUM=n*sum(Slambda_COCUM(1:k))
    /sum(Slambda_COCUM)-(C*k*(k+1)/p);
    G_COCUM=n*sum(Dif_COCUM(1:k))
    /(2*sum(Dif_COCUM))-(C*k*(k+1)/p);
    allG_COCUM=[allG_COCUM G_COCUM];
end
dim_est_COCUM=find((allG_COCUM) == max(allG_COCUM(:)));
```

```
all_dim_est_CUME=[all_dim_est_CUME dim_est_CUME];
all_dim_est_COCUM=[all_dim_est_COCUM dim_est_COCUM];
end
%-------------------------all outputs------------------------%
avg_r_CUME=mean(all_r_CUME)
avg_r_COCUM=mean(all_r_COCUM)
std_r_CUME=std(all_r_CUME)
std_r_COCUM=std(all_r_COCUM)
avg_dis_CUME=mean(all_dis_CUME)/dim
avg_dis_COCUM=mean(all_dis_COCUM)/dim
std_dis_CUME=std(all_dis_CUME)
std_dis_COCUM=std(all_dis_COCUM)
avg_dim_est_CUME=mean(all_dim_est_CUME)
avg_dim_est_COCUM=mean(all_dim_est_COCUM)
summary_dimension_CUME=tabulate(all_dim_est_CUME)
summary_dimension_COCUM=tabulate(all_dim_est_COCUM)
plot(y)
```

BIBLIOGRAPHY

[1] Affymetrix R Image Library (2009), "GeneChip single feature and GeneChip hybridization images," *www.affymetrix.com*, May 2013.

[2] Affymetrix Technical Note (2007), "Array design for the GeneChip human genome U133 set," *www.affymetrix.com/support/technical/technotes/hgu133 design technote.pdf*, May 2013.

[3] Akaike H. (1973), "Information theory and an extension of the maximum likelihood princip," *principle. 2nd Int. Symp. Inf. Theory (B. N. Petrov and F. CzAki, eds)*, 267-281.

[4] Alon U., Barkai N., Notterman D., Gish K., Mack S. and Levine J. (1999), "Broad Patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," *PNAS*, 96, 6745-6750.

[5] An H., Huang D., Yao Q., and Zhang C. (2009), "Stepwise Searching for Feature Variables in Hig-Dimensional Linear Regression," *stats.lse.ac.uk/q.yao/qyao.links/paper/ahyz08.pdf*, 2012.

[6] Bakin S. (1999), "Adaptive regression and model selection in data mining problems," *PhD Thesis. Australian National University, Canberra.*

[7] Belkin M. and Niyogi P. (2003), "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," *Neutral Science*, Vol. 15, No. 6, 1373-1396.

[8] Bellman R. (1961), "Adaptive control processes: A guided tour," *Princeton University Press, Princeton, N.J.*

[9] Bondell H. D. and Li L. (2009), "Shrinkage inverse regression estimation for model-free variable selection," *Journal of the Royal Statistical Society, Ser. B*, 71, 287299.

[10] Bura E. and Cook R. D. (2001), "Estimating the structural dimension of regressions via parametric inverse regression," *Journal of the Royal Statistical Society, Ser, B*, 63, 393-410.

[11] Bura E. and Cook R. D. (2001), "Extending Sliced Inverse Regression: The Weighted Chi-Squared Test," *Journal of American Statistical Association*, 96, 996-1003.

[12] Burges C. J. C. (2009), "Dimension Reduction: A Guided Tour," *Foundations and Trends in Machine Learning*, Vol 2, Issue 4, 2009, 275-365.

[13] Chand S. (2012), "On Tuning Parameter Selection of Lasso-Type Methods-A Monte Carlo Study," *The 9th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, Islamabad, Pakistan.

[14] Chen C. H. and Li K. C. (1998), "Can SIR be as popular as multiple linear regression?" *Statistica Sinica*, 8, 289-316.

[15] Chen J. (2008), "Extended Bayesian information criteria for model selection with large model spaces," *Biometrika*, 95, 3, 759-771.

[16] Chong I. G. and Jun C. H. (2005), "Performance of some variable selection methods when multicollinearity is present," *Chemometrics and Intelligent Laboratory Systems*, 78, 103-112.

[17] Cook R. D. (1996), "Graphics for regressions with a binary response," *Journal of the American Statistical Association*, Vol. 91, No. 435, 983-992.

[18] Cook R. D. (1998), "Regression Graphics," *Wiley*, New York.

[19] Cook R. D. (1998), "Principal Hessian directions revisited (with discussion)," *Journal of the American Statistical Association*, 93, 84-94.

[20] Cook R. D. (2004), "Testing Predictor Contributions in Sufficient Dimension Reduction," *The Annals of Statistics*, Vol. 32, No. 3, 1062-1092.

[21] Cook R. D. (2007), "Fisher Lecture: Dimension Reduction in Regression," *Statistical Science*, Vol. 22, No. 1, 1-26.

[22] Cook R. D. and Li B. (2002), "Dimension Reduction for Conditional Mean in Regression," *The Annals of Statistics*, Vol. 30, No. 2, 455-474.

[23] Cook R. D. and Nachtsheim (1994), "Reweighting to Achieve Elliptically Contoured Covariates in Regression," *Journal of the American Statistical Association*, Vol. 89, Issue 426, 592-599.

[24] Cook R. D. and Ni L. (2005), "Sufficient Dimension Reduction via Inverse Regression: A Minimum Discrepancy Approach," *Journal of the American Statistical Association*, Vol. 100, No. 470, 410-428.

[25] Cook R. D. and Ni L. (2006), "Using intraslice covariances for improved estimation of the central subspace in regression," *Biometrika*, Vol. 93, No. 1, 65-74.

[26] Cook R. D. and Setodji C. M. (2003), "A Model-Free Test for Reduced Rank in Multivariate," *Journal of the American Statistical Association*, Vol. 98, No. 462, 340-351.

[27] Cook R. D. and Weisberg S. (1991), "Sliced Inverse Regression for Dimension Reduction: Comment," *Journal of the American Statistical Association*, Vol. 86, No. 414, 328332.

[28] Cook R. D. and Yin X. (2001), "Special Invited Paper: Dimension Reduction and Visualization in Discriminant Analysis," *Australian and New Zealand Journal of Statistics*, 43(2), 147-199.

[29] Diamantaras K. and Kung S. (1996), "Principal component neural networks: Theory and Applications," *NY John Wiely Sons.*

[30] Duan N. and Li K. C. (1991), "Slicing Regression: A Link-Free Regression Method," *The Annals of Statistics*, Vol. 19, No. 2, 505-530.

[31] Dudoit S., Fridyland J. F. and Speed T. P. (2002), "Comparison of discrimination methods for tumor classification based on microarray data," *Journal of the American Statistical Association*, 97, 77-87.

[32] Eatson M. (1986), "A characterization of spherical distributions," *Journal of Multivariate Analysis*, Vol. 20, Issue 2, 272276.

[33] Efron B., Hastie T., Johnstone I, and Tibshirani R. (2004), "Least Angle Regression," *The Annals of Statistics*, Vol. 32, No. 2, 407-451.

[34] Efron B. and Tibshirani R. (1993), "An introduction to the Bootstrap," *Wiley.*

[35] Fan J. and Lv J. (2009), "Invited Review Article: A Selective Overview of Variable Selection in High Dimensional Feature Space," *Statistica Sinica*, 20, 101-148.

[36] Fan J. and Li R. (2001), "Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties," *Journal of the American Statistical Association*, Vol. 96, No. 456, 1348-1360.

[37] Fang K. and Zhu L.-X. (1996), "Asymptotics for kernel estimate of sliced inverse regression," *Journal of the American Statistical Association*, Vol. 96, No. 456, 1348-1360.

[38] Ferre L. (1998), "Determining the Dimension in Sliced Inverse Regression and Related Methods," *Annals of Statistics*, Vol. 24, No. 3, 1053-1068.

[39] Fisher R. A. (1924), "The influence of rainfall on the yield of wheat at Rothamsted," *Philos. Trans. Roy. Soc. London Ser. B*, 213 89142.

[40] Flom P. and Cassell D. L. (2007), "Stopping stepwise: Why stepwise and similar selection methods are bad, and what you should use," *Statistics and Datat Analysis*, NESUG 2007.

[41] Friedman J., Hastie T. and Tibshirani R. (2010), "A note on the group lasso and a sparse group lasso," *Technical Report*, Statistics Department, Stanford University.

[42] Friedman J. and Stuetzle W. (1981), "Projection Pursuit Regression," *Journal of the American Statistical Association*, Vol. 76, Issue 376, 817-823.

[43] Fukumizu K., Bach F. R. and Jordan M. (2009), "Kernel dimension reduction in regression," *The Annals of Statistics*, Vol. 37, No. 4, 1871-1905.

[44] Griffths A. J., Miller J. H., Suzuki D. T., Lewontin R. C., and Gelbart W. M. (2000), "Introduction to Genetic Analysis, 7th edition" *W.H. Freeman and Company.*

[45] Griffths A. J., S. R. Wessler, R. C. Lewontin, and S. B. Carroll (2008), "Introduction to Genetic Analysis," *W.H. Freeman and Company.*

[46] Hall P. and Li K. C. (1993), "On almost linearity of low dimensional projections from high dimensional data," *The Annals of Statistics*, Vol. 21, No. 2, 867-889.

[47] Hall P. and Li K. C. (1993), "On almost linearity of low dimensional projections from high dimensional data," *The Annals of Statistics*, Vol. 21, No. 2, 867-889.

[48] Hardle W., Hall P. and Marron J. S. (1988), "How Far are Automatically Chosen Regression Smoothing Parameters from their Optimum," *Journal of the American Statistical Association*, Volume 83, Issue 401, 86-95.

[49] Hardle W. and Stoker T. (1989), "Investigating Smooth Multiple Regression by the Method of Average Derivatives," *Journal of the American Statistical Association*, Volume 84, Issue 408, 986-995.

[50] Horvath S., et al. (2006), "Analysis of oncogenic signaling networks in glioblastoma identifies ASPM as a molecular target," *Proc. Natl. Acad. Sci*, 103, 17402-17407.

[51] Hotelling H. (1936), "Relations Between Two Sets of Variates," *Biometrika*, 28, 321-377.

[52] Hsing T. and Carroll J. (1992), "An Asymptotic Theory for Sliced Inverse Regression," *The Annals of Statistics*, Vol. 20, No. 2, 1040-1061.

[53] Huang J., Ma S., Xie H. and Zhang C. H. (2009), "A group bridge approach for variable selection," *Biometrika*, 96 (2), 339-355.

[54] Irizarry R., et al. (2003), "Exploration, normalization, and summaries of high density oligonucleotide array probe level data," *Biostatistics*, 4, 249-264.

[55] Karp P. D., Ouzounis C. A., et al. (2005), "Expansion of the BioCyc collection of pathway/genome databases to 160 genomes," *Nucleic Acids Research*, 19, 6083-6089.

[56] Li B., Artemiou A. and Li L. (2011), "Principal support vector machines for linear and nonlinear sufficient dimension reduction," *The Anals of Statistics*, Vol. 39, No. 6, 3182-3210.

[57] Li B., Cook R. D. and Chiaromonte F. (2003), "Dimension Reduction for the Conditional Mean in Regression with Categorical Predictors," *The Anals of Statistics*, Vol. 31, No. 5, 1636-1668.

[58] Li B. and Dong Y. (2009), "Dimension Reduction for Nonelliptically Distributed Predictors," *The Anals of Statistics*, Vol. 37, No. 3, 1272-1298.

[59] Li B., Kim M. K. and Altman N. (2010), "On dimension folding of matrix- or array-valued statistical objects," *The Anals of Statistics*, Vol. 38, No. 2, 1094-1121.

[60] Li B. and Wang s. (2007), "On Directional Regression for Dimension Reduction," *Journal of the American Statistical Association*, 102:479, 997-1008.

[61] Li B., Wen S. and Zhu L. (2008), "On a Projective Resampling Method for Dimension Reduction With Multivariate Responses," *Journal of the American Statistical Association*, 103:483, 1177-1186.

[62] Li B. and Yin X. (2007), "On Surrogate Dimension Reduction for Measurement Error Regression: An Invariance Law," *The Anals of Statistics*, Vol. 35, No. 5, 2143-2172.

[63] Li B., Zhua H., and Chiaromonte F. (2005), "Contour Regression: A General Approach to Dimension Reduction," *The Anals of Statistics*, Vol. 33, No. 4, 1580-1616.

[64] Li C. and Li H. (2008), "Network-constrained regularization oand variable selection for analysis of genomic data," *Bioinformatics*, Vol. 24, No. 9, 1175-1182.

[65] Li C. and Duan N. (1989), "Regression Analysis Under Link Violation," *The Anals of Statistics*, Vol. 17, No. 3, 1009-1052.

[66] Li K. C. (1991), "Sliced Inverse Regression for Dimension Reduction," *Journal of the American Statistical Association*, 86, 316-327.

[67] Li K. C. (1992), "On Principal Hessian Directions for Data Visualization and Dimension Reduction: Another Application of Stein's Lemma," *Journal of the American Statistical Association*, Vol. 87, Issue 420, 1025-1039.

[68] Li L. (2007), "Sparse sufficient dimension reduction," *Biometrika*, 94,3, 603-613.

[69] Li L. (2007), "A Selective Review of Sufficient Dimension Reduction," *www.bios.unc.edu/research/bias/documents/unc.pdf*, North Carolina State University.

[70] Li L. (2008), "Model Free Variable Selection via Sufficient Dimension Reduction," *Workshop at Isaac Newton Institute*, Cambridge, UK.

[71] Li L., Cook R. D. and Nachtsheim C. (2004), "Cluster-based estimation for sufficient dimension reduction," *Computational Statistics and Data Analysis*, Vol. 47, Issue 1, 175193.

[72] Li L., Cook R. D. and Nachtsheim C. (2005), "Model-Free Variable Selection," *Journal of the Royal Statistical Society, Ser. B*, 67, 285-299.

[73] Li L., Li B. and Zhu L. (2010), "Groupwise Dimension Reduction," *Journal of American Statistical Association*, Vol. 105, No. 491, 1188-1201.

[74] Li L. and Nachtsheim C. (2006), "Sparse Sliced Inverse Regression," *Technometrics*, 48:4, 503-510.

[75] Li L. and Nachtsheim C. (2007), "Comment: Fisher Lecture: Dimension Reduction in Regression," *Statistical Science*, 22, 36-39.

[76] Li L. and Yin X. (2008), "Sliced inverse regression with regularizations," *Biometrics*, 64, 124-131.

[77] Li Y. and Zhu L.-X. (2007), "Asymptotics for sliced average variance estimation," *Annals of Statistics*, Vol. 35, No. 1, 41-69.

[78] Lippman, Z., A.-V. Gendrel M. Black, M. W. Vaughn, N. Dedhia, W. R. McCombie, K. Lavine, V. Mittal, B. May, K. D. Kasschau, J. C. Carrington, R. W. Doerge, V. Colot, and R. Martienssen (2004), "Role of transposable elements in heterochromatin and epigenetic control," *Nature*, 430, 471-476.

[79] Liu J., Chen J. and Ye J. (2009), "Large-Scale Sparse Logistic Regression," *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 547-556.

[80] Liu J. and Ye J. (2009), “SLEP: Sparse learning with efficient projections,” $http://www.public.asu.edu/_jye02/Software/SLEP, ArizonaStateUniversity, May2013.$

[81] Luo L., Peng G., Zhu Y., Dong H., Amos C. and Xiong M. (2010), “Genome-wide gene and pathway analysis,” *European Journal of Human Genetics*, 18, 1045-1053.

[82] Ma S. and Kosorok M. R. (2009), “Identification of differential gene pathways with principal component analysis,” *Bioinformatics*, 25, 882-889.

[83] Mallow C. L. (1973), “Some Comments on C_p,” *Technometrics*, Vol. 15, Issue 4, 661-675.

[84] Manoli T., Gretz N. and others. (2006), “Group testing for pathway analysis improves comparability of different microarray datasets,” *Bioinformatics*, 22, 2500-2506.

[85] Matthews L., Gopinath G., Gillesphie M. and others. (2008), “Reactome knowledgebases of biological pathways and processes,” *Nucleic Acids Research*, 37, 619-622.

[86] Mehmood T., Liland K. H., Snipen, L. and Saebo S. (2012), “A review of variable selection methods in Partial Least Squares Regression,” *Chemometrics and Intelligent Laboratory Systems*, 118, 62-69.

[87] Meier L., van de Geer S. and Bühlmann P. (2008), “The group lasso for logistic regression,” *Journal of the Royal Statistical Society, Ser. B*, 70, 53-71.

[88] Murtaugh P. (2009), “Performance of several variable-selection methods applied to real exological data,” *Ecology Letters*, 12, 1061-1068.

[89] Nguyen D. and Rocke D. M. (2002), “Partial least squares proportional hazard regression for application to DNA microarray data. B,” *Bioinformatics*, 18, 1625-1632.

[90] Ni L., Cook D. and Tsai C. (2005), “A note on shrinkage sliced inverse regression,” *Biometrika*, 92, 242-247.

[91] Nishii R. (1984), “Asymptotic Properties of Criteria for Selection of Variables in Multiple Regression,” *The Annals of Statistics*, Vol. 12, No. 2, 758-765.

[92] Ogata H., Goto S., Sato K., Fujibuchi W., Bono H. and Kanehisa M. (1999), “KEGG: Kyoto encyclopedia of genes and genomes,” *Nucleic Acids Research*, 27, 29-34.

[93] Olbricht G. (2010), “Incorporating genome annotation in the statistical analysis of genomic and epigenomic tiling array data,” *Ph.D Thesis, Purde University, West Lafayette, Indiana.*

[94] Pan W., Xie B. and Shen X. (2010), “Incorporating predictor network in penalized regression with application to microarray data,” *Biometrics*, 66, 474-484.

[95] Pearson K. (1901), “On lines and planes of closest fit to systems of points in space,” *Lond. Edinb. Dub. Phil. Mag. J. Sci.*, 6th ser., 2, 559-573.

[96] Pfeiffer R. M., Forzani L. and Bura E. (2011), “Sufficient dimension reduction for longitudinallly measured predictors,” *Statistics in Medicine*, DOI: 10.1002/sim. 4437.

[97] Rosenwald A, Wright G and others. (2003), "The proliferation gene expression signature is a quantitative integrator of oncogenic events that predicts survival in mantle cell lymphoma," Cancer Cell,3, 185-197.

[98] Roweis S. T. and Saul L. K. (2000), "Nonlinear Dimensionality Reduction by Locally Linear Embedding," Science,Vol. 290, No. 5500, 2323-2326.

[99] Scholkoph B., Smola A. and Muller K. R. (1998), "Nonlinear Component Analysis as a Kernel Eigenvalue Problem," *Neural Computation*, Vol. 10, No. 5, 1299-1319.

[100] Schott J. R. (1994), "Determining the dimensionality in sliced inverse regression," *Journal of American Statistical Association*, Vol. 89, No. 425, 141-148.

[101] Schwarz G. (1978), "Estimating the dimension of a model," *The Annals of Statistics*, Vol. 6, No. 2, 461-464.

[102] Serfling R. J. (1980), " Approximation Theorems of Mathematical Statistics," *Wiley Series in Probability and Mathematical Statistics, John Wiley and Sons, Inc.*

[103] Shahbaba B., Shachaf C. M. and Yu Z. (2011), "A pathway analysis method for genome-wide association studies," *Statistics in Medicine*, DOI: 10.1002/sim. 4477.

[104] Shi J. and Malik J. (2000), "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence*, Vol. 22, Issue 8, 888 - 905.

[105] Shi P. and TSAI C. L.(2002), "Regression model selectiona residual likelihood approach," *J. R. Statist. Soc. B*, 64, 237252.

[106] Silva V. and Tenenbaum J. B. (2002), "Unsupervised learning of curved manifolds," *http://web.mit.edu/cocosci/Papers/conformal4.pdf*, May 2013.

[107] Simon N., Friedman J., Hastie T., and Tibshirani R. (2012), "The sparse group lasso," *Journal of Computational and Graphical Statistics*, in press.

[108] Simon N. and Tibshirani R. (2011), "Standardization and the group lasso penalty," *Statistica Sinica.*

[109] Sun X., Zhang L., and others. (2012), "Multi-scale agent-based brain cancer modeling and prediction of TKI treatment response: Incorporating EGFR signaling pathway and angiogenesis," *BMC Bioinformatics*, 13, 218.

[110] Tibshirani R. (1996), "Regression shrinkage and selection via the lasso," *Journal of Royal Statistics. Society B*, 58, 267-288.

[111] Tipping M. and Bishop C. M. (1999), "Probabilistic Principal Component Analysis," *Journal of Royal Statistics. Society B*, Vol. 61, Issue 3, 611-622.

[112] Tikhonov AN (1943), "On the stability of inverse problems," *Doklady Akademii Nauk SSSR*, 39(5), 195198.

[113] Tong T. (2010), "Variable Selection and Model Building," *amath.colorado.edu/courses/7400/2010Spr/lecture15.pdf*, May 2013.

[114] Wang H., Li B. and Leng C. (2009), "Shrinkage tuning parameter selection with a diverging number of parameters," *Journal of Royal Statistics. Society B*, 71, 671-683.

[115] Wang H. and Xia Y. (2008), "Sliced Regression for Dimension Reduction," *hansheng.gsm.pku.edu.cn/pdf/2008/SR-Main.pdf*, May 2013.

[116] Wang K., Li M. and Bucan M. (2007), "Pathway-Based Approaches for Analysis of Genomewide Association Studies," *American Journal of Human Genetics*, 81, 1278-1283.

[117] Wang L., "High Dimensional Data Analysis," *http://lilywang.myweb.uga.edu/Research/highdimension.pdf*, Department of Statistics and Probability, Michigan State University, May 2013.

[118] Wang Q., (2009), "Sufficient Dimension Reduction and Sufficient Variable Selection," *Ph.D Thesis: University of Georgia, Athens, Georgia.*

[119] Watson, J. D. and F. H. C. Crick (1953), "A structure for deoxyribose nucleic acid," *Nature* 171, 737-738.

[120] Wang T., Xu P. and Zhu L.-X. (2012), "Non-convex penalized estimation in high-dimensional models with single-index structure," *Journal of Multivariate Analysis*, 109, 221-235.

[121] Wei P. and Pan W. (2008), "Incorporating gene networks into statistical tests for genomic data via a spatially correlated mixture model," *Bioinformatics*, 24, 404-411.

[122] Weisberg S. (2011), "The dr Package," *cran.r-project.org/package=dr*, May 2013.

[123] Wen X. (2007), "A note on sufficient dimension reducion," *Statistics and Probability Letters*, 77, 817-821.

[124] Wen X. and Cook R. D. (2007), "Optimal sufficient dimension reduction in regressions with categorical predictors," *Journal of Statistical Planning and Inference*, 137, 1961-1978.

[125] Wen X. and Cook R. D. (2009), "New approaches to model-free dimension reduction for bivariate regression," *Journal of Statistical Planning and Inference*, 139, 734-748.

[126] Werner T. (2008), "Bioinformatics applications for pathway analysis of microarray data," *Current Opinion in Biotechnology*, 19, 50-54.

[127] Wong W. H. and Li B. (1992), "Laplace expansion for posterior densities of nonlinear functions of parameters," *Biometrika*, 79 (2), 393-398.

[128] Wu Q. and Liang F. and Mukherjee S. (2008), "Localized Sliced Inverse Regression," *Journal of Computational and Graphical Statistics*, Vol. 19, Issue 4, 843-860.

[129] Xia Y. (2007), "A constructive approach to the estimation of dimension reduction directions," *Annals of Statistics*, Vol. 35, No. 6, 2654-2690.

[130] Xia Y., Tong H., Li W. K., and Zhu L.-X. (2002), "An adaptive estimation of dimension reduction space," *Journal of Royal Statistical Society*, Vol. 64, No. 3, 363-410.

[131] Yang Y., (2005), "Can the strengths of AIC and BIC be shared?" *Biometrika*, 92, 937-950.

[132] Yin X. and Cook R. D (2002), "Dimension reduction for the conditional kth moment in regression" *Journal of the Royal Statistical Society: Series B*, Vol. 64, Issue 2, 159-175.

[133] Yin X. and Cook R. D (2003), "Estimating central subspaces via inverse third moments" *Biometrika*, 90 (1), 113-125.

[134] Yin X. and Li B. (2011), "Sufficient dimension reduction based on a nensemble of minimum average variance estimators" *The Annals of Statistics*, Vol. 39, No. 6, 3392-3416.

[135] Yin X., Li B., and Cook R. D. (2008), "Successive direction extraction for estimating the central subspace in a multiple-index regression," *Journal of Multivariate Analysis*, 99, 1733-1757.

[136] Yu Z., Zhu L.-X., and Wen X. (2012), "On model-free conditional coordinate tests for regressions," *Journal of Multivariate Analysis*, 109, 161-72.

[137] Yuan M. and Lin Y. (2006), "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society, Ser. B*, 68, 49-67.

[138] Zeng P. and Zhu Y. (2010), "An integral transform method for estimating the central mean and central subspaces," *Journal of Multivariate Analysis*, 101, 271-290.

[139] Zhang C. H. (2010), "Nearly unbiased variable selection under minimax concave penalty," *Annals of Statistics*, Vol 38, No. 2, 894-942.

[140] Zhu H. (2011), "Pharmacometabolomics Data Analysis and Nonlinear Sucient Dimension Reduction for Genome-Scale Studies," *Ph.d Thesis: North Carolina State University, Raleigh, North Carolina.*

[141] Zhu H. and Li L. (2011), "Biological pathway selection through nonlinear dimension reduction," *Biostatistics*, 12, 429-444.

[142] Zhu L., Wang T., Zhu L.-X. and Ferre L. (2010), "Sufficient dimension reduction through discretization-expectation estimation," *Biometrika*, 97, 295-304.

[143] Zhu L. and Zhu L.-X. (2009), "On distribution-weighted partial least squares with diverging number of highly correlated predictors," *Journal of the Royal Statistical Society, Ser. B*, 71, 525-548.

[144] Zhu L., Zhu L.-X. and Feng Z. (2010), "Dimension reduction in regression through cumulative slicing estimation," *Journal of American Statistical Association*, 105:492, 1455-1466.

[145] Zhu L., Zhu L.-X. and Wen S. (2010), “On dimension reduction in regressions with multivariate responses,” *Statistica Sinica*, 20, 1291-1307.

[146] Zhu L.-X., Miao B. and Peng H. (2006), “On sliced inverse regression with high-dimensional covariates,” *Journal of American Statistical Association*, Vol. 101, No. 474, 630-643.

[147] Zhu L.-X. and Ng K. W. (1995), “ASYMPTOTICS OF SLICED INVERSE REGRESSION,” *Statistica Sinica*, 5, 727-736.

[148] Zhu Y. and Zeng P. (2006), “Fourier methods for estimating the central subspace and the central mean subspace in regression,” *Journal of American Statistical Association*, Vol. 101, No. 476, 1638-1651.

[149] Zou H. and Hastie T. (2005), “Regularization and variable selection via the elastic net,” *Journal of the Royal Statistical Society, Ser. B*, 67, 301-320.

[150] 23 and Me, “What are genes?” *https://www.23andme.com/gen101/genes/*, May 2013.

VITA

Bilin Zeng was born in Xiamen (Amoy), a beautiful coastal city in the southeast of People's Republic of China. She attended Shanghai University of Finance and Economics from 2004 to 2008 and completed her Bachelor's degree in Statistics in May 2008. She joined University of Missouri-Rolla in August, 2008 and obtained her master degree in Mathematics with emphasis in Statistics from Missouri University of Science and Technology (formerly University of Missouri-Rolla) in 2010. She continued working towards her Doctor of Philosophy in Mathematics with emphasis in Statistics in the Department of Mathematics and Statistics at Missouri University of Science and Technology and received her doctoral degree in August 2013.